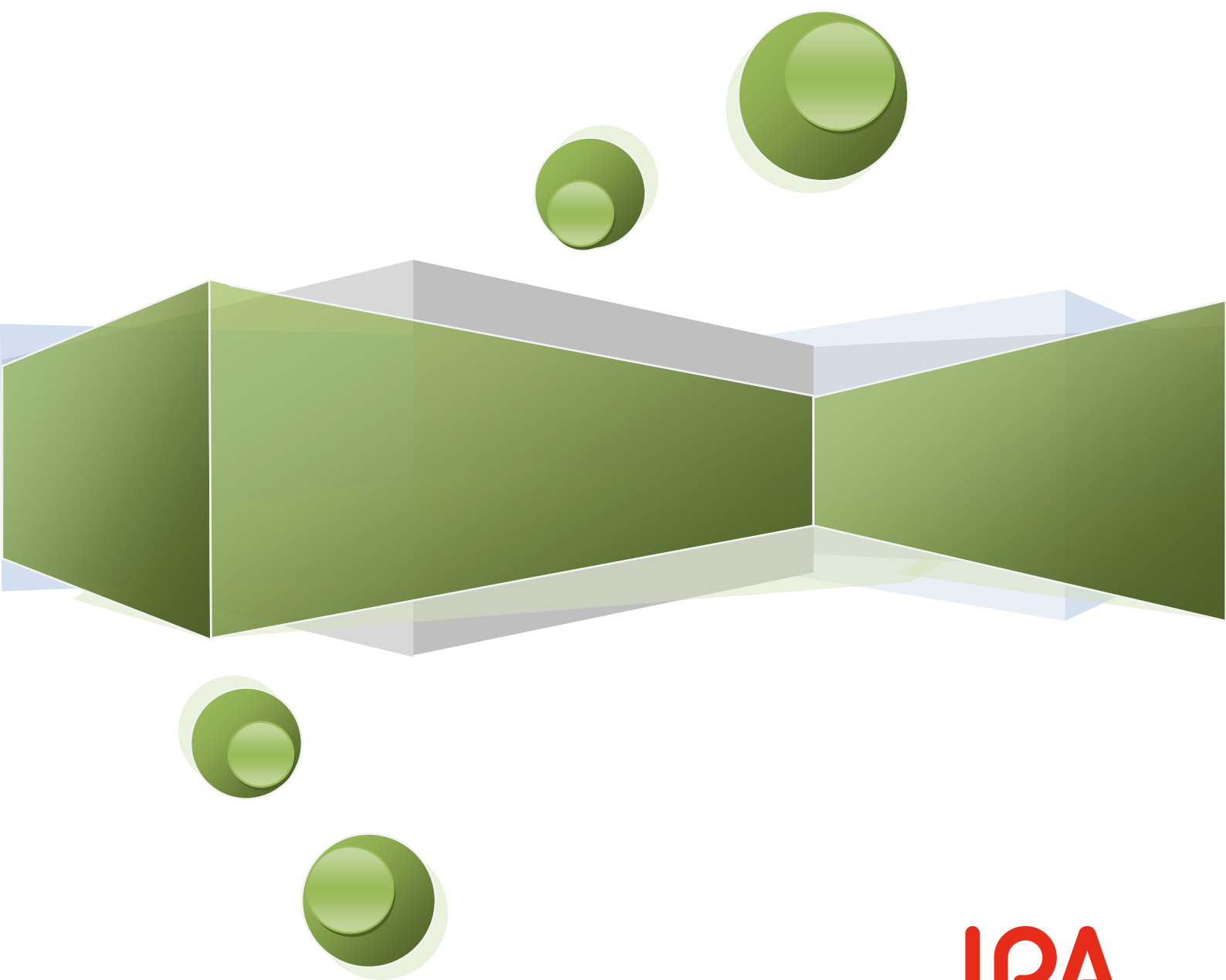


New FE Textbook Vol.1

IT Fundamentals



IPA

INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN

IT Fundamentals

Introduction

1 Computers and Information Society	2
2 Computers Within Society	4
3 The World Surrounding Computers, and the Structure of this Textbook.....	7
4 The Relationship Between the ITEE and this Textbook.....	9

Chapter 1 Hardware

1 Basic Configuration of Computers	13
1 – 1 History of Computers	13
1 – 2 Five Major Units of Computers	16
2 Data Representation in Computers	20
2 – 1 Data Representation	20
2 – 2 Radix and Radix Conversion	23
2 – 3 Representation Form of Data	28
3 Central Processing Unit and Main Memory Unit.....	46
3 – 1 Configuration of CPU	46
3 – 2 Main Memory Configuration	49
3 – 3 Instruction and Addressing	52
3 – 4 Circuit Configuration of ALU	61
3 – 5 High Speed Technologies	70
4 Auxiliary Storage.....	78
4 – 1 Magnetic Disk	78
4 – 2 Optical Disc	86
4 – 3 Semiconductor Memory	89
4 – 4 Other Auxiliary Storage Media and Drives	90
5 Input/Output Unit.....	92
5 – 1 Input Unit	92
5 – 2 Output Unit	97
5 – 3 Other Input/Output Units	103
5 – 4 Input/Output Control Methods	104
5 – 5 Input/Output Interfaces	105
Exercises.....	112

Chapter 2 Information Processing System

1	Processing Type of Information Processing System	121
1 – 1	Non-interactive Processing System and Interactive Processing System	121
1 – 2	Batch Processing System and Real-time Processing System	122
1 – 3	Centralized Processing System and Distributed Processing System	124
2	Configuration of High-reliability System	131
2 – 1	Series System	131
2 – 2	Parallel System	132
2 – 3	Multiplexing System	133
3	Evaluation of Information Processing System	137
3 – 1	Evaluation of the Processing Power	137
3 – 2	Evaluation of Reliability	143
3 – 3	Evaluation of Cost Efficiency	151
4	Human Interface	153
4 – 1	Human Interface Technology	153
4 – 2	Interface Design	157
5	Multimedia	169
5 – 1	Multimedia Technology	169
5 – 2	Multimedia Application	176
	Exercises.....	179

Chapter 3 Software

1	Classification of Software	184
1 – 1	Systematic Classification of Software	184
1 – 2	Classification by Software License	192
2	OS (Operating System)	196
2 – 1	Functions and Configurations of OS	196
2 – 2	Management Functions of OS	197
3	Programming Languages and Language Processors	211
3 – 1	Classification of Programming Languages	211
3 – 2	Language Processor	217
3 – 3	Program Attributes	226
4	Files	227
4 – 1	Files and Records	227
4 – 2	File Access Methods	230
4 – 3	File Organization Formats	231
4 – 4	File Management in Small Computers	235
4 – 5	Backup	238
	Exercises.....	240

Chapter 4 Database

1	Outline of Database	247
1 – 1	Difference Between Database and File	247
1 – 2	Database Design	248
1 – 3	DBMS (DataBase Management System)	254
2	SQL.....	262
2 – 1	Data Definition	262
2 – 2	Data Manipulation	266
3	Various Databases	279
3 – 1	Distributed Database	279
3 – 2	Data Warehouse	280
3 – 3	Other Related Techniques	281
	Exercises.....	283

Chapter 5 Network

1	Network Mechanism	289
1 – 1	Types and Characteristics of Networks	289
1 – 2	Basic Configuration of a Network	291
1 – 3	Basic Techniques of a Network	293
1 – 4	Transmission Control Procedures	305
1 – 5	Communication Services	310
2	Network Architecture.....	314
2 – 1	What is Network Architecture?	314
2 – 2	OSI (Open Systems Interconnection)	314
2 – 3	TCP/IP	316
3	LAN.....	319
3 – 1	Basic Techniques of a LAN	319
3 – 2	Other LAN Techniques	326
4	The Internet.....	328
4 – 1	TCP/IP Protocol	328
4 – 2	Basic Configuration of the Internet	336
4 – 3	Internet Services	337
5	Network Management	341
5 – 1	Network Operations Management	341
5 – 2	Network Management Techniques	342
	Exercises.....	344

Section 6 Security

1 Overview of Information Security	352
1 – 1 Concept of Information Security	352
1 – 2 Information Security Technology	361
1 – 3 Information Security Management	374
1 – 4 Information Security Agencies and Evaluation Criteria	381
2 Information Security Measures	383
2 – 1 Human Security Measures	383
2 – 2 Technical Security Measures	384
2 – 3 Physical Security Measures	387
2 – 4 Security Implementation Technology	388
Exercises	396

Chapter 7 Data Structure and Algorithm

1 Data Structure	403
1 – 1 Array	403
1 – 2 List	405
1 – 3 Stack and Queue	407
1 – 4 Tree Structure	409
2 Basic Algorithm	416
2 – 1 Flowchart	416
2 – 2 Data Search Process	419
2 – 3 Data Sorting Process	426
2 – 4 Other Algorithms	438
2 – 5 Algorithm Design	443
Exercises	445

Index	450
--------------	------------



Introduction



1 Computers and Information Society

The society in which we live today is sometimes called the “Information Society” or the “Information Processing Society.” This “Information Society” is one in which people need to create or locate, from among a flood of data, the information that is beneficial to themselves. Such an explanation may prompt the question of what is the difference between data and information. Data and information are, in fact, entirely separate things. Looking up each in the dictionary reveals the following descriptions.

Data	= “Material that forms the basis for debate or research”
Information	= “Reports of circumstances or conditions that are beneficial to some person”

An example may help make the difference clear.

The numeric string “20080607” is considered here. This numeric string is simply a meaningless listing of numerals as it is. (This is an example of data.) However, when “20080607” is interpreted as the “07”-th day of the “06”-th month of the year “2008”, the string comes to have a meaning, such as a “date of birth.” In this manner, data that holds meaning is called “information.” In other words, “data” becomes “information” when it comes to hold meaning.

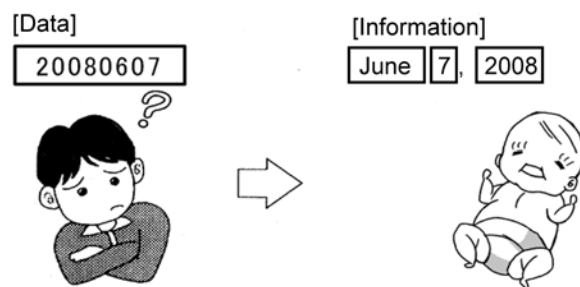


Figure 1 The difference between data and information

What is important here is correctly interpreting “data” as “information.”

For example, the numeric string that is interpreted above as a birthday could actually be the “phone number” “2008” - “0607”, or could be a “product sales record” that tells us that “product number” “200”, at “80” dollars per unit, sold “607” units. In other words, “data” can become entirely different “information,” depending on how it is interpreted. Correctly and quickly processing data is important. The tool necessary for doing so is the “computer.”

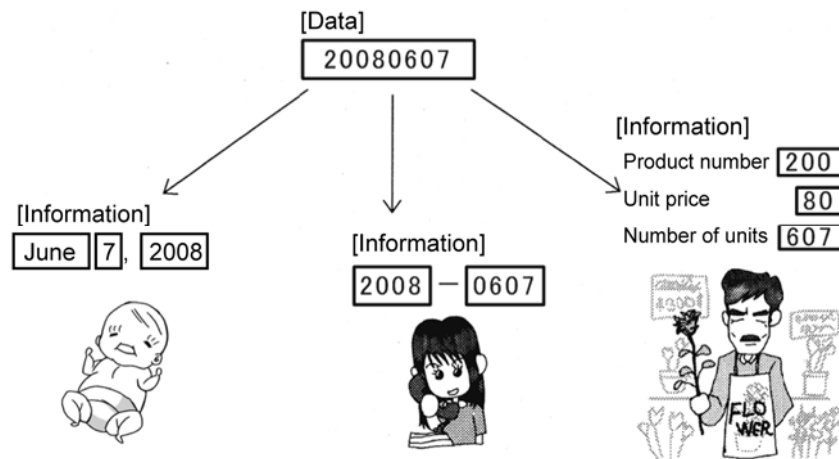


Figure II Data can be interpreted in many ways

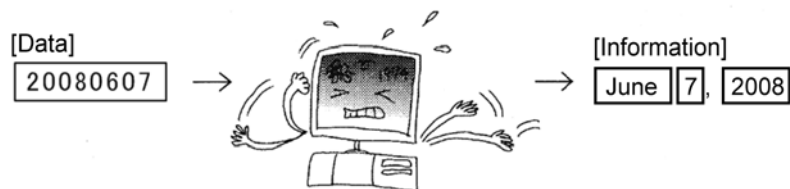


Figure III The computer is a tool for quickly obtaining correct information

2 Computers Within Society

The following examples describe how computers are actually used in society.

(1) POS system

POS (Point Of Sales) system is a system for managing information at the point of sales. While the idea may seem complex, the system has become familiar to most people.

When one buys something at a super store or a convenience store, a machine beeps when the product **bar code** is scanned and read. This device is the “**barcode reader**” that enters product data into a POS system. A barcode, which appears as a group of lines affixed to a product, represents a variety of information through the thickness and spacing of the lines. The role of “POS system” is to read this information, analyze factors such as what products are selling, and determine the number of products that are purchased.



4988 615 018428

Bar code



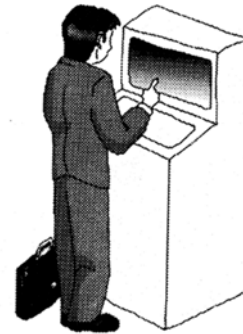
POS system terminal



Barcode reader

(2) ATM system

In a bank, there is a machine that allows the use of a card or a passbook to withdraw cash or check account balances. Users insert a card or a passbook into the machines, and enter instructions for making deposits or withdrawals along with the monetary value, or issue instructions to check account balances. The processes (e.g., the cash withdrawal) are then carried out on the basis of the personal information (e.g., the account balance) that is registered with the bank. If money is withdrawn, the amount withdrawn must be subtracted from the account balance at the same time. “ATM system” allows this process to take place without inconsistencies. This system greatly enhances convenience, by letting customers withdraw money from banks other than the one holding the deposit, and by eliminating the need to queue at a bank counter.



ATM system

(3) Seat reservation system

A seat reservation system is used to secure a variety of reservations, ranging from a seat for a high-speed train or an airplane to a ticket for a concert or an event. Before this kind of system is created, workers had to perform reservation tasks manually. Reservations had to be performed at specified locations, which caused queues from early hours and troubles such as duplicate reservations for the same seats. The description may sound unbelievable to people who have only used modern reservation systems.



Seat reservation system

(4) The Internet

It would be difficult to imagine anyone who does not know about “the Internet” today. Computer is used as a terminal for connecting to “the Internet.” Computer is also used in the administration and operation of “the Internet.” The Internet could truly be called a system representative of today’s information society.

“Actions that people can perform through use of the Internet” can be broadly divided into “information search/transmission” and “online shopping.”

[Information search/transmission]

This refers to the search for information that people want to know (i.e., search), and the release of information that people want to be learned by others (i.e., transmission). Today there are numerous search engines (i.e., tools to aid in investigating information), and countless websites that range from individuals to companies.



An example of a website

[Online shopping]

This refers to the use of the Internet for shopping. At present, many companies sell their products via their own websites. There are also many virtual stores and shopping malls that sell products only on the Internet, without brick-and-mortar stores. Payments through credit cards and online banking have become increasingly common payment methods, as an environment has been prepared for convenient shopping from home.



An example of online shopping

3 The World Surrounding Computers, and the Structure of this Textbook

Today, since computers perform a variety of jobs, the world is becoming increasingly connected around computers. The world surrounding computers today is composed of three broadly divided structural components as shown in Figure IV.

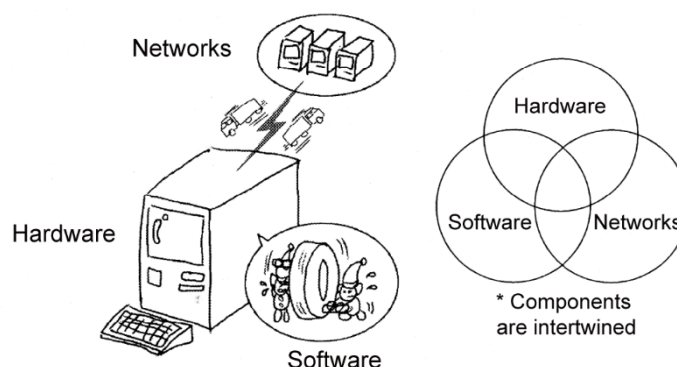


Figure IV Structure of the computing world

The following detailed discussion uses POS system that is introduced in the previous section “2 Computers Within Society” as an example.

(1) Hardware

This is the mechanical component that makes up computers. Hardware technology has seen dramatic advances in recent years, which results in creating a succession of achievements thought impossible until now. The decrease in the cost of hardware is also a major factor behind today’s proliferation of computers.

In a POS system, the “barcode reader” that is used to read the bar codes on products is one component of hardware. The bar codes are read optically by lasers. You will be able to learn about such hardware mechanisms in “[Chapter 1 Hardware](#).”

The read information is not processed by the POS system terminal in the store. Instead, it is sent to and processed by a large scale computer in the head office. This form of processing is called “online transaction processing.” Today, various types of data are generated, and each type of data is processed in a variety of form. When computers break down, the resulting impact is often large and widespread. For that reason, the most appropriate form of processing must be considered, with factors such as reliability in mind. You will be able to learn the knowledge required to do so in “[Chapter 2 Information Processing System](#).”

(2) Software

This is the component of a computer that instructs the computer to perform activity. Computers cannot act under their own discretion, without instruction by humans. The creation of documents, the exchange of e-mails, and such other function that are made possible by using computers is due to software that performs the processing.

A POS system carries out processes that include “Read bar code,” “Record data,” and “Calculate monetary amount.” In order to perform the processing, commands are issued to the hardware. (For example, the command “Read in the product bar code” is issued to a barcode reader.) Issuing these commands is the role of software. You will be able to learn about the mechanisms by which software issues orders, and ways by which resources are used efficiently, in “[Chapter 3 Software](#).” Also, you can learn about “databases,” a means of managing and using data, in “[Chapter 4 Database](#).”

In actual processing, it is possible to check the names or unit prices of products on the basis of the data contained in product codes, as an example. In order to do so, it is necessary to think of efficient procedures for finding the target product codes from among many codes. There is a close relationship between these procedures and the structure used to record data. You will be able to learn about data structures and processing procedures in “[Chapter 7 Data Structure and Algorithm](#).”

(3) Network

Network is technology for exchanging information among computers. The rapid growth of computers in recent years can be seen as a result of the rapid growth in networks. This is because jobs that once could not be performed on computers now can be, because of the exchange of information among many computers.

In a POS system, data read at the store is transmitted for processing by large scale computers in the head office. Accurate and rapid transmission of the data requires knowledge of what methods and procedures to use in transmission. You will be able to learn basic knowledge about such communications in “[Chapter 5 Network](#).”

In order to combine the above (1) through (3) and perform a variety of activities accurately, rapidly, and securely, a variety of “systems” are created (with the POS system being one such example). However, we cannot use such systems, nor use data that is used on networks such as the Internet, with assurance if these are not properly and securely managed. In response, a variety of network technologies are employed to enable the secure use of systems and networks. You will be able to learn about the concepts of information security management that are used in these security technologies in “[Chapter 6 Security](#).”

4 The Relationship Between the ITEE and this Textbook

The ITEE (Information Technology Engineers Examination) is a national examination that is implemented by the IPA (Information Technology Promotion Agency, Japan) as one index for the training of human resources in the information processing industry. Under a 2008 revision to the examination system, the ITEE has been mapped to Levels 1 through 3 of the Common Career/Skills Framework, with level determined upon success in the examination.

Level	Name of examination	Targeted human resources
Level 1	IT Passport Examination	Persons with basic knowledge of the information technology that should be common to professionals
Level 2	Fundamental Information Technology Engineer Examination	Persons who possess the necessary basic knowledge and skills for becoming advanced IT professionals, and also have acquired practical capabilities for using them
Level 3	Applied Information Technology Engineer Examination	Persons who possess the necessary applied knowledge and skills, and also have established a direction, for becoming advanced IT professionals

This textbook is structured to allow learning in technology related fields within the scope of the morning questions in the Fundamental Information Technology Engineer Examination of the ITEE. The following table shows the correspondence relationship between the chapters of this textbook and the question areas of the Fundamental Information Technology Engineer Examination.

IT Fundamentals		Question areas of the Fundamental Information Technology Engineer Examination
Chapter 1	Hardware	1 Basic Theory (Basic Theory [Discrete Mathematics, etc.]
		2 Computer System (Computer Component)
		2 Computer System (Hardware)
Chapter 2	Information Processing System	2 Computer System (System Component)
		3 Technology Element (Human Interface)
		3 Technology Element (Multimedia)
Chapter 3	Software	2 Computer System (Software)
Chapter 4	Database	3 Technology Element (Database)

Chapter 5	Network	3 Technology Element (Network)
Chapter 6	Security	3 Technology Element (Security)
Chapter 7	Data Structure and Algorithm	1 Basic Theory (Algorithm and Programming)

Areas that are within the scope of morning questions but that are not covered in this textbook can be studied in another textbook “IT Strategy and Management.”

The following table shows the correspondence relationship between the chapters of “IT Strategy and Management” and the question areas of the Fundamental Information Technology Engineer Examination.

IT Strategy and Management		Question areas of the Fundamental Information Technology Engineer Examination
Chapter 1	Corporate and Legal Affairs	1 Basic Theory (Basic Theory [Applied Mathematics, etc.])
		9 Corporate and Legal Affairs (Corporate Activities)
		9 Corporate and Legal Affairs (Legal Affairs)
Chapter 2	Business Strategy	8 Business Strategy (Business Strategy Management)
		8 Business Strategy (Technological Strategy Management)
		8 Business Strategy (Business Industry)
Chapter 3	Information Systems Strategy	7 System Strategy (System Strategy)
		7 System Strategy (System Planning)
Chapter 4	Development Technology	4 Development Technology (System Development Technology)
		4 Development Technology (Software Development Management Techniques)
Chapter 5	Project Management	5 Project Management (Project Management)
Chapter 6	Service Management	6 Service Management (Service Management)
Chapter 7	System Audit and Internal Control	6 Service Management (System Audit)

This textbook is organized, as noted above, so as to allow mastery of the morning questions of the Fundamental Information Technology Engineer Examination, when used together with “IT Strategy and Management.”

Moreover, since this textbook corresponds to Level 2 of the Common Career/Skills Framework, its study contents include the question areas of the Level 1 IT Passport Examination. The IT Passport Examination is an examination for all who have acquired the fundamental knowledge required of persons using IT.

With the use of this textbook and “IT Strategy and Management,” it is possible to take the IT Passport Examination and the Fundamental Information Technology Engineer Examination in order. We hope that this textbook will be of use in improving the reader’s skills, with the aim of acquiring the desired qualifications.



Chapter 1

Hardware



1 Basic Configuration of Computers

Various types of computers are extensively used around us. However, it has not even passed 100 years since the first computer was launched. Moreover, the basic configuration of computers has hardly changed since their initial launch. This chapter touches upon the history of computers from each generation and describes the five major units that form the basic configuration of a computer.

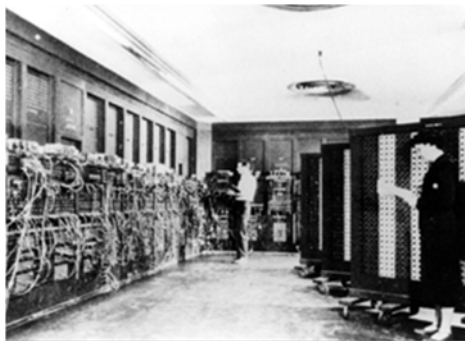
1 - 1 History of Computers

Generations of computers are divided according to the **logic gates** they used. A computer consists of components known as logic gates, which correspond to the brain of the computer that is used in arithmetic operations.

(1) 1st generation (1940s)

The world's first computer **ENIAC** was developed by J.W. Mauchly and J.P. Eckert in 1946. ENIAC used more than 18,000 **vacuum tubes** for its logic gates. Therefore, an enormous amount of heat was generated and the system consumed too much electricity for cooling, which even caused power outages. In those days, the computer was mainly used in ballistic calculations. However, it was necessary to replace circuit wiring according to the contents to be processed, and therefore, some people do not recognize it as a computer.

It was **EDSAC**, developed by M.V. Wilkes in 1949, which eliminated the replacement of circuit wiring required in ENIAC. EDSAC was a computer that used a **stored-program system**. The stored-program system stores the contents to be processed as a program inside the computer and then runs it, and computers based on this system are also known as **Neumann computers** from the name of its inventor (J. Von Neumann).



World's first computer: ENIAC

(2) 2nd generation (1950s)

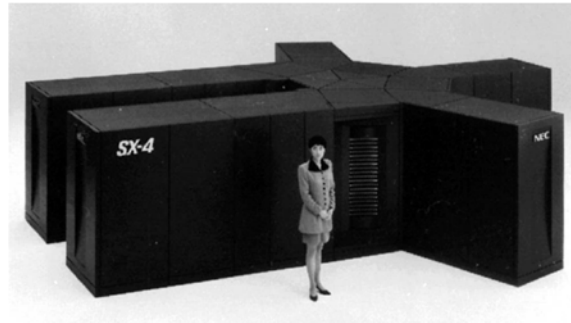
While 1st generation computers were mainly used in military and research and development, **UNIVAC I** was launched as a commercial computer in 1951. In this generation of computers, semiconductors like **transistors** were used as logic gates. Semiconductors have intermediate electrical properties between conductors that pass electricity and insulators that do not pass electricity. Besides transistors, semiconductor devices include the **diode** (rectifying device), which only passes the electric current in one direction, and **LED (Light Emitting Diode)**, which emits light when voltage is applied in the forward direction. Semiconductors are smaller than vacuum tubes, and their failure occurrence rate is low. Therefore, computers became compact, and their reliability increased.

(3) 3rd generation (1960s)

In this generation of computers, **IC (Integrated Circuit)** was used as a logic gate. IC implements the processing capacity of several-hundred transistors in a square silicon chip of a few millimeters, thanks to the advancement of semiconductor device technology. Computers became extremely compact and fast thanks to IC technology, and several manufacturers announced various types of computers. A typical example is **IBM/360** developed by the IBM Corporation. This computer was a **general-purpose computer** that was capable of handling any kind of processing without any restrictions on the usage purposes.

(4) 3.5th generation (1970s)

In this generation, IC technology made further progress, and manufacturers began using **LSI (Large Scale Integration)** as logic gates. LSI has higher integration density than IC, which enabled computers to become even more compact and faster. Since computers became compact, this led to the development of **control computers** that are used in industrial devices and **microprocessors** that are used in home appliance products. Moreover, their high-speed technology led to the development of **super computers** that are very useful for high-speed operations in the area of scientific and engineering computations. Moreover, **microcomputers** were also launched as small computers for individual use.



Super computer

(5) 4th generation (1980s)

Logic gate technology progressed further, and manufacturers began using **VLSI (Very Large Scale Integration)**. Such exponential progress of hardware technology transformed the era of computers from “one computer in a company or facility” to “one computer per person.” Manufacturers started developing and selling **PC (Personal Computers)** for individuals, which formed the foundation of the present-day information society.



Desktop computer



Laptop computer

In this era, the network environment was also developed, and it became common to use computers (**servers**) that provide services and terminals by connecting them through networks. In this usage, besides PCs, **work stations** that have higher performance than PCs were also used as terminals connected to networks. Moreover, the sizes of devices rapidly decreased with the development of the palm-sized **PDA (Personal Digital Assistant)** and **smartphone**; other **personal digital assistants** such as portable **tablet terminals**; and **SoC (System on a Chip)** and **one chip microcomputers (single chip microcomputers)** where functionality of computer is packed in one chip (LSI).

(6) 5th generation (?)

The progress of computer technology continues even today. For example, there is **FPGA (Field Programmable Gate Array)**, which can be programmed after manufacture through

simulation of a design diagram. FPGA is slower than a dedicated LSI, and it is also expensive. However, as compared to implementing similar functions in software, FPGA devices are high-speed and low-cost devices, which is why they are used in home appliance products, etc. With the advancement of various technologies, computers increasingly became compact and high performance, and concentrated efforts were made so that even beginners could use them easily. Moreover, as the next generation computers, computers (e.g., computers with inference function) closer to people and more closely linked to society are being developed. Furthermore, as part of a recent trend, computers with a focus on energy conservation are also being developed. **Power consumption** of computers is expressed in units of Watts (W) just like general electrical appliances. Each of the components used in computers consumes power. Therefore, researchers are working on design technologies combining components that have low power consumption.

1 - 2 Five Major Units of Computers

Computers are used in various fields, and there are various types of computers. However, the basic configuration elements are the same. This subsection describes the units that are used in computers.

In order to explain the configuration elements of computers, a list is created for the functions that are required for computers. The easiest way to understand this is to think in terms of human action and behavior. If we think about human behavior in solving the problem “ $3+6= \square$ ”, it becomes as follows:

- 1) Look at the problem and memorize it.
↓
- 2) Think about the meaning of “+”.
...We know this is a sign that adds two values.
↓
- 3) Memorize “9” as a result of adding “3” and “6”.
↓
- 4) Write “9” in the answer sheet.



Figure1-1
Action and behavior of humans

The five units of a computer perform this same action and behavior of humans. These units are called the **five major units** of a computer.

- 1) **Input unit**
This is a unit that reads data to be processed by computer.
- 2) **Output unit**
This is a unit that writes the processing results in a form that can be understood by humans.
- 3) **Storage unit**
This is a unit that records data. There are main memory and auxiliary storage.
- 4) **Arithmetic and logical unit**
This is a unit that performs arithmetic operations on data that is stored in the storage unit or makes a decision as per the instructions of the control unit.
- 5) **Control unit**
This is a unit that interprets a command and gives instructions to the remaining four units.

Below is an explanation of these units by using a specific example. When the arithmetic operation “ $3+6= \square$ ” is performed by a computer, a summary of the operation of each unit is as follows:

- 1) Enter “ $3+6$ ” from the input unit, and store it in the storage unit.
↓
- 2) Control unit decodes the meaning of “ $+$ ”.
As a result of decoding, it knows that “ $+$ ” is a command for making addition. Therefore, it gives an instruction to the arithmetic and logical unit.
↓
- 3) The arithmetic and logical unit fetches “ 3 ” and “ 6 ” from the storage unit, and performs the calculation. The result “ 9 ” is stored in the storage unit.
↓
- 4) “ 9 ” is written from the storage unit to the output unit.

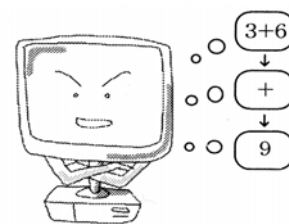


Figure 1-2
Functioning of a computer

Both input and output also operate on the basis of the instructions that are given by the control unit, which is not covered here. Therefore, the flow of data and control (instructions) between each unit is as shown in Figure 1-3 below.

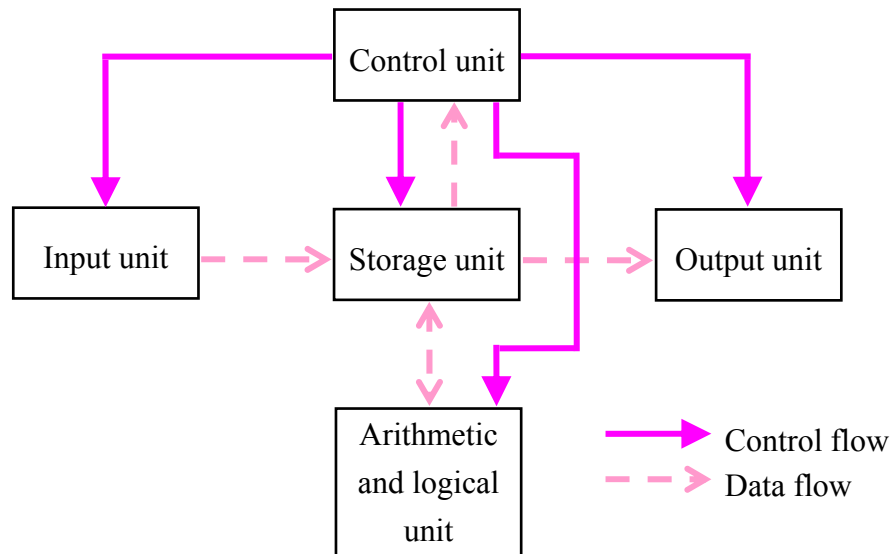


Figure1-3 Flow of control and data in five major units

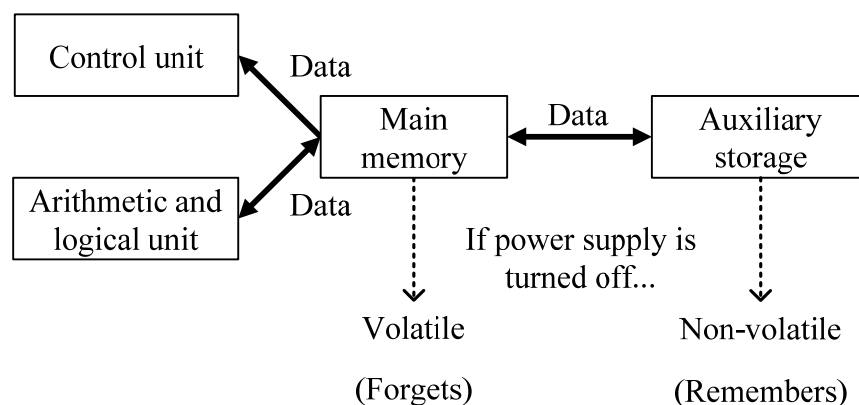
In the storage unit, there is main memory and auxiliary storage. (In Figure 1-3, what is mentioned as the storage unit is main memory.) The differences between these two are described below.

- **Main memory**

This unit can directly exchange data with the control unit and the arithmetic and logical unit, and it has a **volatility** characteristic (contents are lost if power supply is turned off).

- **Auxiliary storage**

This unit stores data that cannot be accommodated in main memory, and it has a **non-volatility** characteristic (contents are not lost even if power supply is turned off).



Moreover, the control unit and the arithmetic and logical unit are collectively referred to as the **CPU (Central Processing Unit)** or the **processor**. The input unit, the output unit, and

the auxiliary storage outside the processor are also referred to as **peripheral devices**.

Present-day computers use **microprocessors** (or **MPU (Micro Processing Unit)**), where CPU (processor) functions are consolidated into one LSI. Microprocessors are also used in home electrical appliances other than PCs.

On the other hand, a semiconductor chip (LSI) where all required functions (system) including memory are integrated is referred to as **SoC (System on a Chip)**. SoC has advantages such as high speed and low power consumption. However, it suffers from the disadvantage of high development risk. Therefore, it is also used in combination with **SiP (System in a Package)**, which integrates multiple semiconductor chips into one package. Moreover, in a **one chip microcomputer (single chip microcomputer)** used in home electrical appliances, not only CPU and memory functions but also input and output functions are integrated into one semiconductor chip.

In addition, there are **co-processors** that perform only specific processes for assisting the processor, and **dedicated processors** that are focused only on specific processes, unlike **general purpose processors** that perform various processes like in PCs.

2 Data Representation in Computers

In order to make a computer work, it is necessary to store a program with a processing sequence and data to be processed inside the computer in advance. This section describes the methods of recording (representing) information inside computers.

2 - 1 Data Representation

(1) Unit of representation

Inside computers, data is recorded as electrical signals. Electric signals can basically represent only two states.

- Is electric current flowing? \longleftrightarrow Is it not flowing?
- Is voltage high? \longleftrightarrow Is it low?

0 and 1 are linked to these two states, and they are recorded and saved as data inside computers. In other words, data recorded by computers is represented with 0 or 1. The minimum unit that represents this 0 or 1 is called a **bit**. Various meanings are formed by combining 0 and 1 represented by this bit, and they are recorded as data. At this time, one unit formed by collecting 8 bits is called a **byte**.

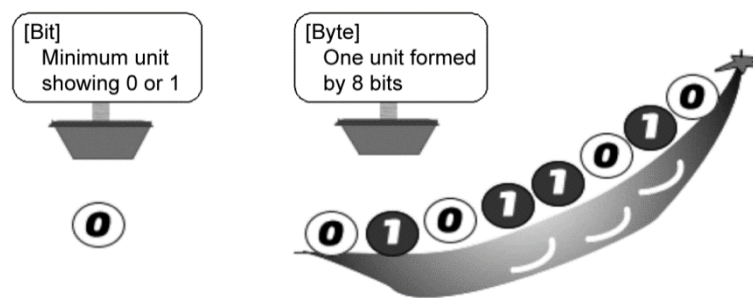


Figure 1-4 Bit and byte

Moreover, there is a unit called **word**, which is formed by collecting more bits than in a byte. Word is the unit of processing inside computers, and there are 16 bits, 32 bits, 64 bits, etc. according to the model of computer used. Needless to say that if more bits can be processed at a time, then more information can be processed in a certain time. Therefore, the higher the number of bits in one word, the higher the processing speed of the computer. In most present-day computers, 1 word is 32 bits or 64 bits.

(2) Information amount

There are two types (0, 1) of information that can be represented with 1 bit, and the number of bit combinations is referred to as the **information amount**. The information amount that can be represented with 2 bits is of 4 types (00, 01, 10, 11), the information amount that can be represented with 3 bits is of 8 types (000, 001, 010, 011, 100, 101, 110, 111), ... and so on.

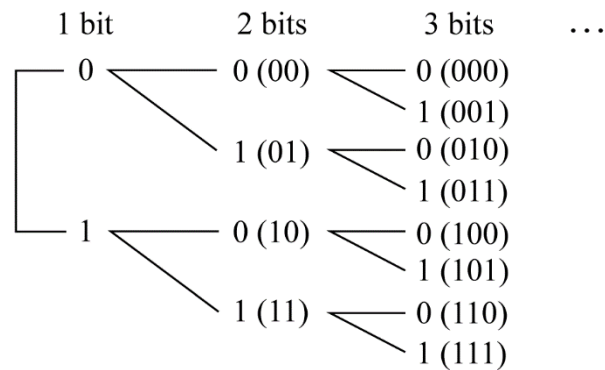


Figure1-5 Number of bits and information amount

As shown in Figure 1-5, branches increase with the increase of every one bit used, and the information amount that can be represented keeps on doubling. In other words, the information amount that can be represented with n bits is of **2^n types** (2^n shows the power of 2, and means multiplying 2 by itself n times). Using this concept, a summary of the information amount that can be represented with byte and word (16 bits) is given below.

*** Information amount of n bits = 2^n types**

- Information amount that can be represented with 1 byte (=8 bits)
= 2^8 types = 256 types (00000000 through 11111111)
- Information amount that can be represented with 1 word (=16 bits)
= 2^{16} types = 65,536 types (0000000000000000 through 1111111111111111)

[Information amount in information theory]

In the field of information theory, when event x occurs with **occurrence probability** (probability that a certain event occurs) $P(x)$, the information amount that can be obtained is defined as $I(x)$. According to this definition, the smaller $P(x)$ is, the larger $I(x)$ is; and the larger $P(x)$ is, the smaller $I(x)$ is. (This is easy to understand if the information amount is considered as the degree of surprise when an event occurs.)

To explain the relation between bit and information amount, 1 bit represents the information of 0 or 1. In other words, the occurrence probability $P(x)$ of event x , where either 0 or 1 appears, is $1/2$, and so the information amount of occurrence probability $1/2$ is 1 bit. Therefore, information amount n bit is an event with occurrence probability $1/2^n$; in other words, it means information of 2^n types.

(3) Prefix

When data is handled, rather large numbers are handled as they are. However, it is very difficult to handle very large numbers or very small numbers without modification. Therefore, they are represented in combination with a **prefix** (auxiliary unit) representing a certain value.

For example, even if a nurse holding a newborn baby says “He is a healthy baby boy of 3,000g,” no one would say “I lately got a little fat and became 75,000g.” Normally, one would say “75 kg.” This “k” is the prefix. “k” is a prefix that means the value “ 10^3 ”, and it can represent values like “75kg = 75×10^3 g”.

The readings and values of major prefixes are summarized in the table below.

[Prefixes used for representing large numbers]

Symbol	Reading	Decimal value	Binary value
k	Kilo	10^3	2^{10}
M	Mega	10^6	2^{20}
G	Giga	10^9	2^{30}
T	Tera	10^{12}	2^{40}
P	Peta	10^{15}	2^{50}

[Prefixes used for representing small numbers]

Symbol	Reading	Decimal value
m	Milli	10^{-3}
M	Micro	10^{-6}
n	Nano	10^{-9}
p	Pico	10^{-12}

Here, for prefixes that represent large numbers, binary values are also shown in the table. This is because inside computers, for representing information in binary form by using two numbers of 0 and 1, prefixes are also represented in binary form when numbers related to computers are represented (details of binary numbers are explained later).

For examples, data like “1k calorie” is not related to computers. Therefore, it means

$$1\text{k calorie} = 1 \times 10^3 \text{ calories} = 1,000 \text{ calories.}$$

On other hand, data like “1k byte” is related to computers, and therefore it means

$$1\text{k byte} = 1 \times 2^{10} \text{ bytes} = 1,024 \text{ bytes.}$$

However, it is almost equally related to “ $10^3 \approx 2^{10}$, $10^6 \approx 2^{20}$, ...”, and therefore it is often treated as $1\text{k byte} = 1,000 \text{ bytes}$ these days. Memorizing this relation helps when arithmetic calculation problems are solved.

2 - 2 Radix and Radix Conversion

Decimal numbers that we normally use are numbers that are obtained by raising each digit to the power of 10 by using ten numbers from 0 through 9.

$$\begin{aligned} (362.9)_{10} &= 3 \times \underline{10^2} + 6 \times \underline{10^1} + 2 \times \underline{10^0} + 9 \times \underline{10^{-1}} \\ &= (362.9)_{10} \end{aligned}$$

*10 in ()₁₀ shows that it is a decimal number.

Here, the weight (10) of the decimal digit is called **radix**, and numbers represented with radix n are referred to as **n -adic numbers**. n -adic numbers are represented by using n numbers from 0 through $(n-1)$ that are carried over when n is reached.

(1) Binary numbers

Binary numbers are represented by using two numbers 0 and 1 that are carried over when 2 is reached.

The smallest unit of information handled inside computers is a bit, and only two numbers of 0 and 1 can be used. Therefore, computers use binary numbers that represent numerical values

with two numbers, 0 and 1.

$$(101.1)_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} \\ = (5.5)_{10} \quad * 2 \text{ in } ()_2 \text{ shows that it is a binary number.}$$

Decimal numbers and their corresponding binary numbers are as shown below. Binary numbers can use only two numbers of 0 and 1 for one digit. Therefore, the number of digits quickly becomes large, compared with decimal numbers.

Decimal numbers	0	1	2	3	4	5	6	7	8	9	10	...
Binary numbers	0	1	10	11	100	101	110	111	1000	1001	1010	...

Figure 1-6 Decimal numbers and corresponding binary numbers

(2) Octal numbers

Octal numbers are numerical values that are represented by using eight numbers from 0 through 7 and that are carried over when 8 is reached.

$$(317.5)_8 = 3 \times 8^2 + 1 \times 8^1 + 7 \times 8^0 + 5 \times 8^{-1} \\ = (207.625)_{10} \quad * 8 \text{ in } ()_8 \text{ shows that it is an octal number}$$

Decimal numbers	0	1	2	3	4	5	6	7	8	9	10	...
Octal numbers	0	1	2	3	4	5	6	7	10	11	12	...

Figure 1-7 Decimal numbers and corresponding octal numbers

(3) Hexadecimal numbers

Hexadecimal numbers are numerical values that are represented by using sixteen numbers from 0 through 9, and A(10), B(11), C(12), D(13), E(14), and F(15) and that are carried over when 16 is reached.

$$(1A6.E)_{16} = 1 \times 16^2 + 10 \times 16^1 + 6 \times 16^0 + 14 \times 16^{-1} \\ = (422.875)_{10} \quad * 16 \text{ in } ()_{16} \text{ shows that it is a hexadecimal number.}$$

Decimal numbers	0	1	2	3	4	5	6	7	8	9	10
Hexadecimal numbers	0	1	2	3	4	5	6	7	8	9	A

11	12	13	14	15	16	17	18	19	20	...
B	C	D	E	F	10	11	12	13	14	...

Figure 1-8 Decimal numbers and corresponding hexadecimal numbers

(4) Relation between binary numbers, octal numbers, and hexadecimal numbers

Binary numbers have an extremely large number of digits compared with decimal numbers. Although computers can handle binary numbers without problems, it is very difficult for a human being to understand when the internal status of computers is concerned. Therefore, octal numbers and hexadecimal numbers are used so that people can understand more easily.

Octal numbers use the power of 8 ($= 2^3$) as the weight of each digit, and therefore the information of 3 digits of binary numbers corresponds to 1 digit of octal numbers.

Binary number: (1011100.11101)₂

↓

(001 011 100 . 111 010)₂ ... Separate after every 3 digits with radix point as reference.

↓

(0 is supplemented in the parts that fall short of 3 digits.)

Octal numbers: (1 3 4 . 7 2)₈ ... Convert every 3 digits

Hexadecimal numbers use the power of 16 ($= 2^4$) as the weight of each digit, and therefore the information of 4 digits of binary numbers corresponds to 1 digit of hexadecimal numbers.

Binary number: (1011100.11101)₂

↓

(0101 1100 . 11100 1000)₂ ... Separate after every 4 digits with radix point as reference.

↓

(0 is supplemented in the parts that fall short of 4 digits.)

Hexadecimal number: (5 C . E 8)₁₆ ... Convert every 4 digits

(5) Radix conversion

Radix conversion means changing the radix of a numerical value. It is also radix conversion to convert a binary number (radix 2) into an octal number (radix 8) or a hexadecimal number (radix 16).

When radix has a constant (power) relation as in the case of binary numbers, octal numbers, and hexadecimal numbers, radix conversion can be done by consolidating multiple digits. However, when radix does not have a constant (power) relation as in the case of binary numbers and decimal numbers, it is necessary to convert through computation.

1) Radix conversion from a binary number into a decimal number

Radix conversion from a binary number to a decimal number can be performed by adding the weight of each digit that is 1 in the binary number. (This is the same as adding the results that are obtained by multiplying the weight of each digit with the number (0 or 1) of each digit.)

Example: Perform radix conversion of the binary number $(1001101.101)_2$ into a decimal number.

$$\begin{array}{ccccccccccc} (& 1 & & 0 & & 0 & & 1 & & 1 & & 0 & & 1 & . & 1 & & 0 & & 1 &)_2 \\ & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & & \vdots & \\ & 2^6 & & 2^5 & & 2^4 & & 2^3 & & 2^2 & & 2^1 & & 2^0 & & 2^{-1} & & 2^{-2} & & 2^{-3} & \\ & \downarrow & & & & \downarrow & & \downarrow & & & & \downarrow & & \downarrow & & \downarrow & & & & \downarrow & \\ & 64 & & & & + 8 & + 4 & & + 1 & + 0.5 & + 0.125 & = (77.625)_{10} \end{array}$$

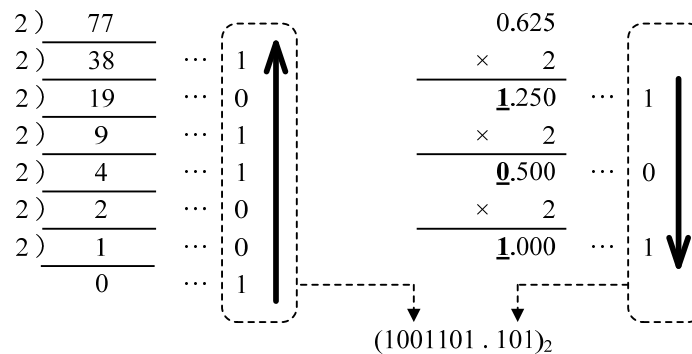
2) Radix conversion from a decimal number into a binary number

Radix conversion from a decimal number into a binary number is performed separately for the integer part and the fractional part.

The integer part can be converted by repeatedly dividing by 2 until the quotient becomes 0, and then arranging the remainder of calculation results in sequence from the back.

A fractional part can be converted by continuously multiplying by 2 until the fractional part in calculation results becomes 0, and then arranging the integer part of each calculation result in sequence from the front.

Example: Perform radix conversion of the decimal number $(77.625)_{10}$ into a binary number.



A summary of radix conversion between decimal numbers and n -adic numbers is provided below.

[Procedure of radix conversion from an n -adic number into a decimal number]

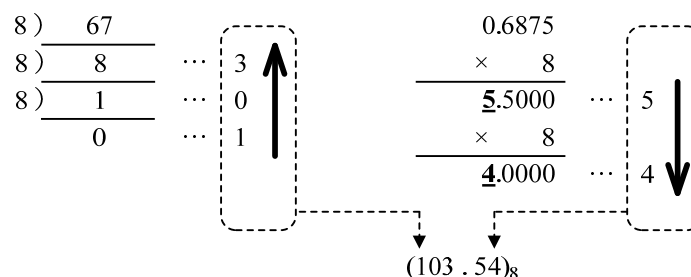
Add all results obtained by multiplying the weight (power of n) of each digit by the number {from 0 through $(n-1)$ } of each digit.

[Procedure of radix conversion from a decimal number into an n -adic number]

Integer part: Repeatedly divide by n until the quotient becomes 0, and then arrange the remainder of calculation results in sequence from the back.

Radix part: Continuously multiply the radix part by n until the radix part in calculation results becomes 0, and then arrange the integer part of each calculation result in sequence from the front.

For example, below is an example of radix conversion of the decimal number $(67.6875)_{10}$ into an octal number.



However, when the radix conversion from decimal to octal or hexadecimal is performed, it may be easier to convert from decimal to binary, and then convert the result into octal or hexadecimal.

$$\begin{aligned}
 \text{Decimal number: } (67.6875)_{10} &= \text{Binary number: } (1000011.1011)_2 \\
 &= (001\ 000\ 011 . 101\ 100)_2 \\
 &= \text{Octal number: } (1\ 0\ 3 . 5\ 4)_8
 \end{aligned}$$

[Fractional number that cannot be represented by the finite number of digits]

When the fractional part of a decimal number is converted into an n -adic number, it gets into a loop as the fractional part of calculation results does not become 0, and it may not be converted into a **finite fraction**. For example, converting the decimal number $(0.2)_{10}$ into a binary number gives the following.

$$\begin{array}{ccccccc}
 \rightarrow & 0.2 & \rightarrow & 0.4 & \rightarrow & 0.8 & \rightarrow & 0.6 \\
 \times 2 & & \times 2 & & \times 2 & & \times 2 & \\
 \hline
 0.4 & & 0.8 & & 1.6 & & 1.2 & \\
 \hline
 \end{array}$$

$(0.2)_{10} = (0.0011001100110011\cdots)_2$

In this manner, when the results of radix conversion become an **infinite fraction** (or **recurring fraction**), numerical values are handled as approximate values inside the computer.

2 - 3 Representation Form of Data

All data is represented with 0 or 1 inside computers. This representation form can be classified as follows:

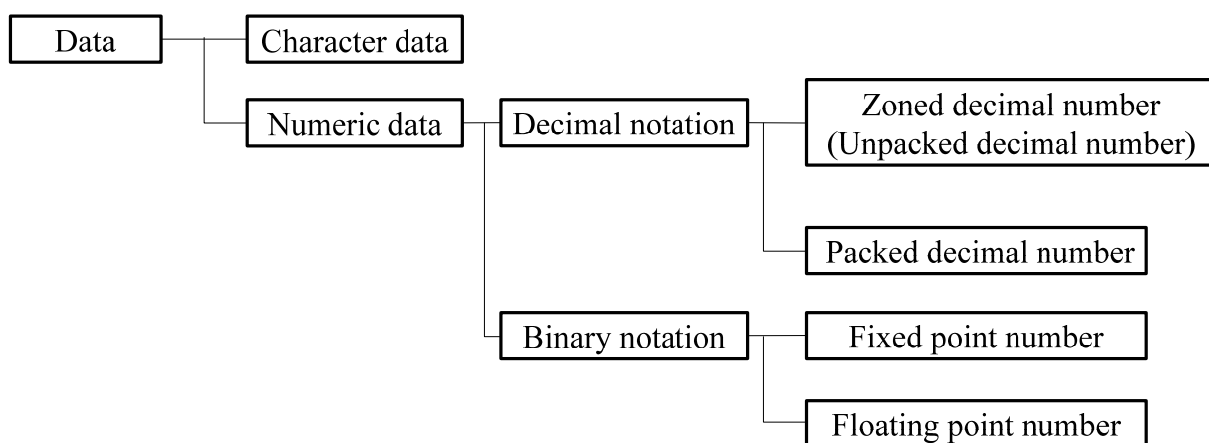


Figure 1-9 Representation Form of data

2-3-1 Character Data

Inside computers, character data is represented with combinations of 0 and 1. In computers during the early stage of their development, 1 character was linked to a bit pattern of 8 bits (1 byte). Moreover, bit patterns linked to characters were referred to as **character codes**, with the six main character codes as described below. When data between computers is exchanged by using different character codes, **character corruption** (garbled characters) may occur where characters that are different from the original data are displayed.

(1) ASCII code

ASCII code is the character code defined by **ANSI (American National Standards Institute)** in 1962. It is composed of 8 bits, which include code bits (7 bits) that represent an alphabetic character or a number, and a parity bit (1 bit) that detects an error. Although it is used in PCs, it does not have any definitions concerning Japanese characters (e.g., kanji, kana).

(2) ISO code

ISO code is a 7-bit character code defined by the standardization agency **ISO (International Organization for Standardization)** on the basis of ASCII code in 1967. This character code forms the basis of character codes used in different countries all over the world.

(3) JIS code

JIS code is a character code that is deliberated by **JISC (Japanese Industrial Standards Committee)** on the basis of the Industrial Standardization Act for representing Japanese-specific characters based on the ISO code and is defined as **JIS (Japanese Industrial Standards)**.

1) **JIS 7-bit codes / JIS 8-bit codes** (JIS X 0201)

This character code can represent half-width katakana characters. Figure 1-10 shows the table of JIS 8-bit codes for reference. Here is how to use the table. Firstly, search in the table for the character to be represented with the code. Secondly, arrange columns (higher 4 bits) and rows (lower 4 bits) from top to bottom and left to right respectively. The arranged numbers form the character code of that character.

Example: Convert “JIS” into character code.

Character	Position in table	Character code	(Hexadecimal notation)
J	... 4th column 10th row	: 0100 1010	(4A)
I	... 4th column 9th row	: 0100 1001	(49)
S	... 5th column 3rd row	: 0101 0011	(53)

Bit number	b8	b7	b6	b5	b4	b3	b2	b1		0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	
	b8	b7	b6	b5	b4	b3	b2	b1		0	0	0	0	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	
	b8	b7	b6	b5	b4	b3	b2	b1		0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	
	b8	b7	b6	b5	b4	b3	b2	b1		0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	
	b8	b7	b6	b5	b4	b3	b2	b1		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15				
Lower bits	0	0	0	0	0					NUL	TC7(DLE)	SP	0	@	P	`	p	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	0	0	1	1					TC1(SOH)	DC1	!	1	A	Q	a	q	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	0	1	0	2					TC2(STX)	DC2	~	2	B	R	b	r	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	0	1	1	3					TC3(ETX)	DC3	#	3	C	S	c	s	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	1	0	0	4					TC4(EOT)	DC4	\$	4	D	T	d	t	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	1	0	1	5					TC5(ENQ)	TC8(NAK)	%	5	E	U	e	u	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	1	1	0	6					TC6(ACK)	TC9(SYN)	&	6	F	V	f	v	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	0	1	1	1	7					BEL	TC10(ETB)	'	7	G	W	g	w	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	
	1	0	0	0	8					FE0(BS)	CAN	(8	H	X	h	x	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	0	0	1	9					FE1(HT)	EM)	9	I	Y	i	y	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	0	1	0	10					FE2(LF)	SUB	*	:	J	Z	j	z	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	0	1	1	11					FE3(VT)	ESC	+	:	K	[k	{	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	1	0	0	12					FE4(FF)	IS4(FS)	.	<	L	¥	l		↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	1	0	1	13					FE5(OR)	IS3(GS)	-	=	M]	m	}	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	1	1	0	14					SO	IS2(RS)	.	>	N	^	n	~	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
	1	1	1	1	15					SI	IS1(US)	/	?	O	_	o	DEL	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑
										Higher bits																			

Figure 1-10 JIS 8-bit codes (JIS X 0201)

2) JIS kanji code (JIS X 0208)

This is a character code for representing 1 hiragana or kanji character with 2 bytes (16 bits).

3) Shift JIS code

This code system is formed by expanding JIS kanji code, and it is a character code that enables mixing of 1-byte characters and 2-byte characters without using a special switching code.

(4) EBCDIC (Extended Binary Coded Decimal Interchange Code)

EBCDIC is an 8-bit character code developed by the IBM Corporation of the United States. This code was originally developed by IBM for its own computers. However, 3rd generation (1960s) computers were mostly IBM computers, and therefore this code became industry standard (**de facto standard**) in the area of large computers.

(5) Unicode

Unicode is the character code that was developed and proposed by U.S.-based companies like Apple, IBM, and Microsoft as a 2-byte universal uniform code for smooth exchange of computer data. It supports characters of several countries such as alphabets, kanji and hiragana/katakana, Hangul characters, and Arabic letters. This was standardized by ISO as an international standard, and at present, **UCS-2** (2 bytes) and **UCS-4** (4 bytes) are defined.

(6) EUC (Extended Unix Code)

EUC (Extended Unix Code) is the character code that was defined by the AT&T Corporation for internationalization support of **UNIX** (an OS (Operating System), which is a software program for controlling computers). An alphanumeric character is represented with 1 byte, and a kanji or kana character is represented with 2 bytes. In this case, kanji characters are codes where the hexadecimal number “80” is added to JIS kanji code. Therefore, in the kanji part, the value of the most significant bit is “1”, and it is possible to differentiate single byte alphanumeric characters and kanji characters.

Moreover, although it is not a character code, **zoned decimal numbers** are also classified as a code that represents character data (Details of zoned decimal numbers are explained in the next sub-subsection entitled “Numeric data”).

2-3-2 Numeric Data

In broad terms, there are two methods for numeric representation inside computers. The first method is to represent numerical values with binary numbers, which is suitable for performing calculations inside computers. However, binary numbers are not easy for humans to use, and therefore there exists another method where decimal numbers are adopted to facilitate understanding for humans.

(1) Decimal notation

This is a numeric representation method of introducing the way of thinking about decimal numbers that are easy-to-understand for humans. In specific terms, **BCD code (Binary-Coded Decimal code)** where each digit of a decimal number is converted into a 4-bit binary number is used.

Example: Represent $(2741)_{10}$ with binary-coded decimal code.

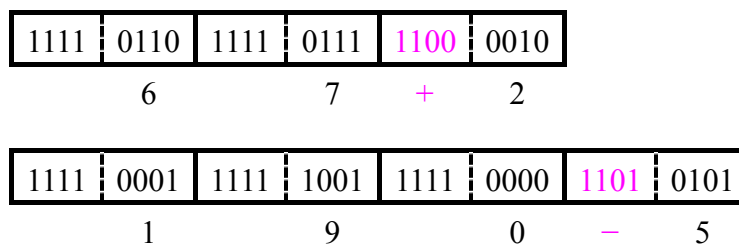
Decimal numbers	:	2	7	4	1
		↓	↓	↓	↓
Binary-coded decimal code	:	0010	0111	0100	0001

1) Zoned decimal number

This is a format that represents 1 digit of a decimal number with 1 byte. A binary-coded decimal code is stored in the lower 4 bits of 1 byte corresponding to each digit, while zoned bits are stored in the higher 4 bits. However, a sign bit indicating a sign is stored in the higher 4 bits of the lowest digit.

Zoned bits and sign bits differ depending on the character code used in computers. When EBCDIC is used, “1111 (F)” is stored in the zoned bit, while “1100 (C)” is stored as a sign bit if it is positive (+) and “1101 (D)” is stored as a sign bit if it is negative (−). On the other hand, when ASCII code or JIS code is used, “0011 (3)” is stored in the zoned bit. In most of the cases, the sign bit uses the same value as in the case of EBCDIC (it differs by manufacturer because there are no uniform definitions).

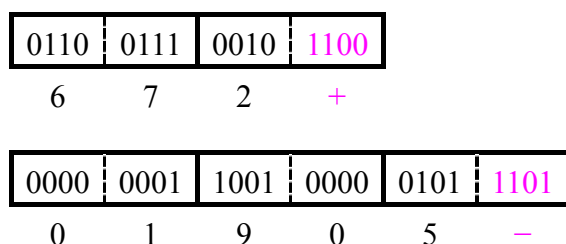
Example: Represent $(+672)_{10}$ and $(-1905)_{10}$ with zoned decimal numbers (EBCDIC).



2) Packed decimal number

This is a format that represents 2 digits of a decimal number with 1 byte. Each digit of a decimal number is represented with a binary-coded decimal code and this is the same as the zoned decimal number, however, it differs because the sign is shown with the lowest 4 bits, and when it falls short of a byte unit, 0 is inserted to make it a byte unit.

Example: Represent $(+672)_{10}$ and $(-1905)_{10}$ with packed decimal numbers.



If we compare zoned decimal numbers and packed decimal numbers, by packing after the omission of the zoned bits (higher 4 bits) of zoned decimal numbers, the packed decimal numbers can represent more information (digits) with fewer bytes. Moreover, it is called **unpacking** to represent packed decimal numbers by using zoned decimal numbers, and therefore the zoned decimal numbers are also referred to as **unpacked decimal numbers**.

While zoned decimal numbers cannot be used in calculations, packed decimal numbers can be used in calculations. However, when decimal numbers from an input unit are entered, or when decimal numbers are sent to an output unit, it is preferable to use zoned decimal numbers because the unit (1-digit numerical value) of input/output and the unit (1 byte) of information correspond to each other. Therefore, conversion between zoned decimal numbers and packed decimal numbers is performed inside computers.

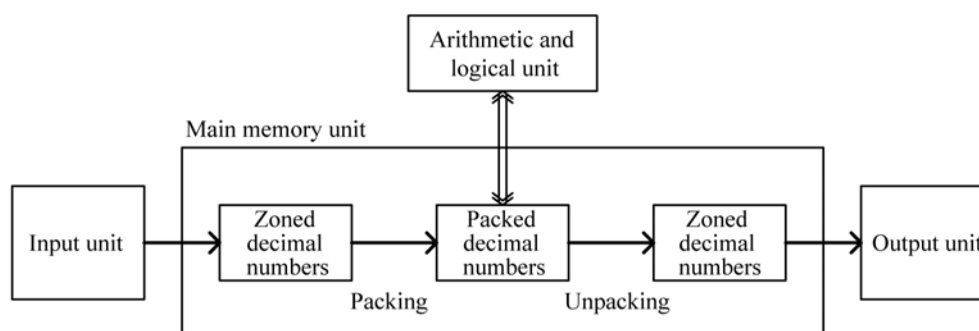


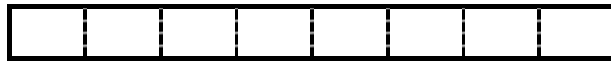
Figure1-11 Conversion between zoned decimal numbers and packed decimal numbers inside computers

(2) Binary notation

Forms for representing numerical values as binary numbers that are easy to handle inside computers include **fixed point numbers** where the number of digits of integer part and fractional part are decided in advance, and **floating point numbers** where the position of radix point is changed according to the numerical value to be represented.

1) Fixed point numbers

In this form of representation, numerical values are handled by fixing the radix point at a specific position. Generally, it is mostly used for handling integer numbers by fixing the position of the radix point in the least significant bit.



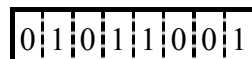
▲ (Position of radix point)

In this form of representation, after numerical values are converted into binary numbers, they are used without modification according to the position of the radix point.

Example: Represent $(89)_{10}$ as an 8 bit fixed point number.

$$(89)_{10} = (1011001)_2$$

↓



In fixed point numbers, methods of representing negative numbers (**representation forms of negative numbers**) are important. The two main methods are described below.

A: Signed absolute value notation

In this method, the most significant 1 bit is used as a sign bit, and 0 is stored in it if the numerical value is positive, while 1 is stored in it if the numerical value is negative. In the remaining bits, the binary bit string is stored as it is.

Example: Represent $(+89)_{10}$ and $(-89)_{10}$ with an 8-bit fixed point number (signed absolute value).

$$(89)_{10} = (1011001)_2$$

$$(+89)_{10}: \quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1$$

$$(-89)_{10}: \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1$$

B: Complement notation

In this method, when a negative number is represented, the **complement** of the positive number is used.

[Complement]

By using a certain number as the reference value, the shortfall is shown with respect to this reference value.

In n -adic number, there is a complement of n and a complement of $n-1$.

- Complement of $n-1$... Reference value is the maximum value that has the same number of digits.
- Complement of n ... Reference value is the minimum value that has one more additional digit.

The complement can be determined by subtracting from the reference value the number for which the complement is to be determined.

Example 1: Determine the complement of decimal number $(614)_{10}$

• 9's complement

$$\begin{array}{r} 999 \\ - 614 \\ \hline 385 \end{array}$$

• 10's complement

$$\begin{array}{r} 1000 \\ - 614 \\ \hline 386 \end{array}$$

Example 2: Determine the complement of binary number $(01101101)_2$

• 1's complement

$$\begin{array}{r} 1111111 \\ - 01101101 \\ \hline 10010010 \end{array}$$

• 2's complement

$$\begin{array}{r} 10000000 \\ - 01101101 \\ \hline 10010011 \end{array}$$

Example: Represent $(+89)_{10}$ and $(-89)_{10}$ with an 8-bit fixed point number (2's complement).

$$\begin{array}{r} (89)_{10} = (1011001)_2 \rightarrow 100000000 \\ - 01011001 \\ \hline 10100111 \end{array}$$

$$(+89)_{10}: \boxed{0} \boxed{1} \boxed{0} \boxed{1} \boxed{1} \boxed{0} \boxed{0} \boxed{1}$$

$$(-89)_{10}: \boxed{1} \boxed{0} \boxed{1} \boxed{0} \boxed{0} \boxed{1} \boxed{1} \boxed{1}$$

Generally, 2's complement notation is often used in fixed point numbers for the following reasons.

1. Subtraction can be substituted for addition.

When signed absolute values are used, addition or subtraction must be appropriately selected according to the first bit of each numerical value to be calculated. However, when 2's complement is used, calculation can be performed without any selection. By using this, subtraction can be represented with addition.

Example: Calculate “ $(15)_{10} - (10)_{10}$ ” as “ $(15)_{10} + (-10)_{10}$ ”.

- 1) Represent $(-10)_{10}$ with 2's complement.

$$\begin{array}{r} (10)_{10} = (1010)_2 \rightarrow 100000000 \\ - 00001010 \\ \hline 11110110 \end{array}$$

$$(-10)_{10}: \boxed{1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0}$$

- 2) Add $(15)_{10}$ to $(-10)_{10}$ represented as 2's complement.

$$\begin{array}{r} (15)_{10} = (1111)_2 \\ (+15)_{10} \quad \boxed{0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1} \end{array}$$

$$\begin{array}{r} 00001111 \quad \dots (+15)_{10} \\ + 11110110 \quad \dots (-10)_{10} \\ \hline 1 \ 00000101 \quad \dots (00000101)_2 = (+5)_{10} \end{array}$$

↑
* Discard the carry-over bit.

By using this concept, a feature for subtraction is not required if computer has the feature for addition and the feature for calculating complement. Moreover, if multiplication is done with iterations of addition, and division is done with iterations of subtractions, four basic arithmetic operations (**addition**, **subtraction**, **multiplication**, **division**) can be done with addition only.

2. A wide range of numerical values can be represented.

When signed absolute values are used, two types of 0s, namely $(+0)$ and (-0) , occur. However, there is only one type of 0 when 2's complement is used. Since the information amount of identical numbers of bits is the same, 2's complement enables the representation of a wider range of numerical values.

Signed absolute value		2's complement	
01111111	+127	01111111	+127
01111110	+126	01111110	+126
:	:	:	:
00000000	+0	00000000	0
10000000	-0	11111111	-1
:	:	:	:
11111110	-126	10000001	-127
11111111	-127	10000000	-128

[Range of numerical values that can be represented with n bits (2's complement)]

From -2^{n-1} through $+2^{n-1}-1$

2) Floating point numbers

If the number of bits that can be used for representing numerical value is decided, the range of numerical values that can be represented with fixed point numbers is limited. Therefore, when a very large numerical value is handled, the number of bits should also be increased accordingly. However, in reality, the number of bits used for recording data cannot be increased infinitely. Therefore, floating point numbers are used to represent extremely large numbers or extremely small numbers below the radix point.

Before the explanation of the representation of numbers inside computers, the mechanism of floating point numbers is explained by using decimal numbers. The case that there is the numerical value below is considered here.

+456,000,000,000

For the ease of explanation, when 1 unit is used for recording a sign or one number, 13 units are required for recording this numerical value as it is.

+	4	5	6	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---

Here, if we convert the numerical value and then record only required information in the following manner, we can represent this numerical value with a total of 6 units.

+456,000,000,000
 $= +0.456 \times 10^{12}$

$$= (-1)^0 \times 0.456 \times 10^{12} \quad * (-1)^0 = +1, \quad (-1)^1 = -1$$

Sign: Sign to show whether the numerical value is positive or negative

0

Fraction (mantissa): Numerical value to be placed after the radix point

4	5	6
---	---	---

Exponent: Power of 10 to be multiplied by fraction (mantissa)

1	2
---	---

This is the concept of floating point numbers. Using this concept, from extremely large numerical values to extremely small numerical values can be represented with the same number of digits.

“10” in power of 10 used in representing exponent is called **radix**. Since a decimal number was used in this explanation, “10” was used as it is easy to understand in terms of moving the radix point. However, since binary numbers are used in computers, either “2” or “16” is used.

[Basic form of floating point numbers]

$$(-1)^{\text{Sign}} \times \text{Fraction (mantissa)} \times \text{Radix}^{\text{Exponent}}$$

In reality, the form of representation of floating point numbers used inside computers differs depending on the model. Here, we explain the **single-precision floating point number** (32-bit format) and **double-precision floating point number** (64-bit format) standardized as IEEE 754 format. A double-precision floating point number can handle a wider range of numerical values with higher precision.

[Single-precision floating point number (32-bit format)]

Sign (1 bit)	Exponent (8 bits)	Fraction (mantissa) (23 bits)
------------------------	-----------------------------	---

[Double-precision floating point number (64-bit format)]

Sign (1 bit)	Exponent (11 bits)	Fraction (mantissa) (52 bits)
------------------------	------------------------------	---

There are several forms of representation of information in each part of the floating point number format. Main forms of representation of each part are as follows:

A: **Sign** (S)

This is used for representing the sign of a numerical value. It is 0 for positive values, and 1 for negative values.

B: Exponent (E)

This is used for representing exponent with respect to radix. The two main representation methods below are used.

- **2's complement:**

In this method, exponent is recorded as a binary number, and 2's complement is used if it is negative (-).

- **Excess method:**

This method records a value (i.e., biased value) that is obtained by adding a certain value to the exponent. The value (i.e., **bias value**) to be added is decided according to the number of bits of exponent. Below is an example of the excess method used with single-precision floating point numbers.

Method name	Intended format (Typical example)	Format of exponent		Bias value
		Number of bits	Information amount	
Excess 127	IEEE 754 format	8 bits	256 types	127
Excess 64	IBM format	7 bits	128 types	64

C: Fraction (mantissa) (M)

This is used for representing a numerical value after the radix point. In order to represent the part after the radix point, it is common to perform **normalization**. In this case, the integer part is either set to 0 or 1 (storing after the omission of 1 from the integer part).

- When the integer part is set to 0: 0.101101 → Store “101101” in fraction (mantissa)
- When the integer part is set to 1: 1.011010 → Store “011010” in fraction (mantissa)

[Normalization]

This operation is for maintaining the precision of numerical values by increasing digits that can be used in fraction (mantissa). In most cases, this is done by reducing extra 0s after the radix point.

For example, in the case of the decimal number $(0.000123456789)_{10}$ represented by using a 7-digit fraction (mantissa), if the 7-digit fraction is registered in fraction (mantissa) as is, the following result is obtained.

0	0	0	1	2	3	4
---	---	---	---	---	---	---

In contrast, if the 7-digit fraction is registered in fraction (mantissa) after normalization, the following result is obtained.

Normalized form: $0.123456789 \times 10^{-3}$

1	2	3	4	5	6	7
---	---	---	---	---	---	---

In other words, more digits can be represented if normalization is performed (**the number of significant digits** increases). Therefore, the precision of numerical values becomes high.

Example: Express the decimal number $(1234.625)_{10}$ with the single-precision floating point number (32 bits) of IEEE 754 format shown below.

Sign (1 bit): Show a positive value with 0, and a negative value with 1.

Exponent (8 bits): Show with excess 127 where radix is 2.

Fraction (23 bits): Show with normalized expression where integer part is 1.

- 1) Convert the decimal number $(1234.625)_{10}$ into a binary number.

$$(1234.625)_{10} = (10011010010.101)_2$$

- 2) Normalize the binary number determined in 1).

$$\text{Normalized form: } +(1.0011010010101)_2 \times 2^{10}$$

- 3) Show exponent as a binary number (8 bits) of excess 127.

$$\text{Power of 10} \rightarrow 10 + 127 = 137 \rightarrow (10001001)_2$$

- 4) Describe according to the form of representation.

0	10001001	001101001010100000000000
---	----------	--------------------------

Sign (1 bit): 0

Exponent (8 bits): 10001001

Fraction (mantissa) (23 bits): 001101001010100000000000

2-3-3 Error

Error refers to the difference between the actual value and the value represented inside the computer. When numerical values inside the computer are handled, we must pay attention to error.

For example, when $(0.1)_{10}$ is converted into a binary number, it is represented as follows:

$$(0.1)_{10} = (0.00011001100110011...)_{2}$$

The conversion result becomes a **recurring fraction** where “0011” is repeated an infinite number of times. However, since there is a limit for the number of bits that can be used for representing numerical values inside computers, the only option for storing in fraction (mantissa) is to store after the loop is cut in the middle. If it is 8 bits, only up to $(0.00011001)_2$ can be stored. When this binary number is converted into a decimal number, $(0.09765625)_{10}$ is obtained. In other words, this is different from the original numerical value $(0.1)_{10}$ by only $(0.00234375)_{10}$. This is error.

(1) Rounding error

Rounding error occurs when the part smaller than the least digit is rounded off, rounded up, or rounded down in order to represent the real number with the effective number of digits in the computer. When a decimal number is converted into a binary number, this kind of error occurs to represent the resulting binary number with the effective number of digits. As a measure against rounding error, there are methods such as minimizing the value of error as much as possible by changing **single precision** (32 bits) into **double precision** (64 bits).

(2) Loss of trailing digits

Loss of trailing digits is the error that occurs when an extremely small value is ignored at the time of computing two values: one absolute value is extremely large and the other is extremely small.

Example: Calculate $(0.10110011)_2 \times 2^{10} + (0.11010001)_2 \times 2^{-10}$.

$$\begin{array}{r}
 1011001100 \\
 + \quad \quad \quad 0.000000000011010001 \\
 \hline
 1011001100.000000000011010001 \\
 = (0.10110011)_2 \times 2^{10} \quad \quad \quad \downarrow \\
 \quad \quad \quad \text{This value will be ignored.}
 \end{array}$$

In order to minimize loss of trailing digits, when multiple numerical values are added with floating point radix numbers, it is necessary to take steps such as arranging all data in the ascending order of absolute value, and adding them in sequence from the top (smaller value).

(3) Cancellation of significant digit

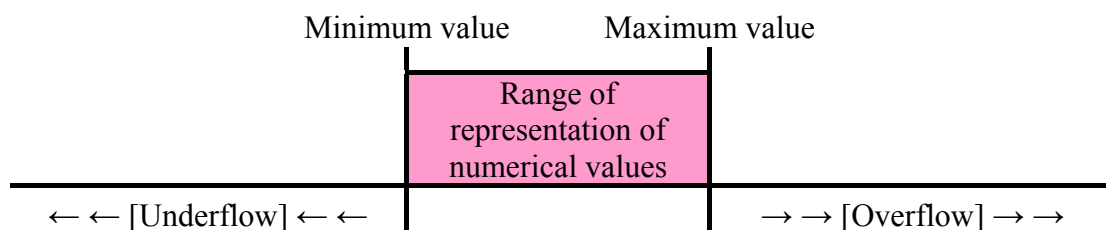
Cancellation of significant digit is the error that occurs because of decline in the **number of significant digits** that can be trusted as numerical values when calculation is performed between almost equal numerical values.

Example: Calculate $(0.10110011)_2 \times 2^0 - (0.10110010)_2 \times 2^0$.

$$\begin{array}{r}
 0.10110011 \\
 - 0.10110010 \\
 \hline
 0.00000001 \\
 = (0.\underline{10000000})_2 \times 2^{-7} \\
 \downarrow \\
 \text{These 0s cannot be trusted as a numerical value.}
 \end{array}$$

(4) Overflow, underflow

Inside a computer, the range of numerical values that can be represented is already decided, because the limited number of bits is used for representing them. Flow is the error that occurs when the calculation results exceed this range of representation. When the calculation results exceed the maximum value of range of representation, it is referred to as **overflow**, and when it exceeds the minimum value, it is referred to as **underflow**. When it is simply referred to as “**flow**,” it mostly means overflow.



Moreover, the following two indexes are used as the concept of evaluating these errors (whether precision is high or low).

- **Absolute error:** This is an index that evaluates on the basis of how large the actual error is.

$$\text{Absolute error} = | \text{True value} - \text{Computed value (including error)} |$$

- **Relative error:** This is an index that evaluates on the basis of the proportion (ratio) of error with respect to true value.

$$\text{Relative error} = \frac{|\text{True value} - \text{Computed value (including error)}|}{|\text{True value}|}$$

2-3-4 Shift Operation

Shift operation is the operation of shifting the position of bit to left or right. Shift operation is used in computation of numerical values, and in changing the position of bits.

(1) Arithmetic shift

Arithmetic shift is the shift operation used when numerical values are computed. It is mainly used in fixed point numbers that represent negative values in 2's complement.

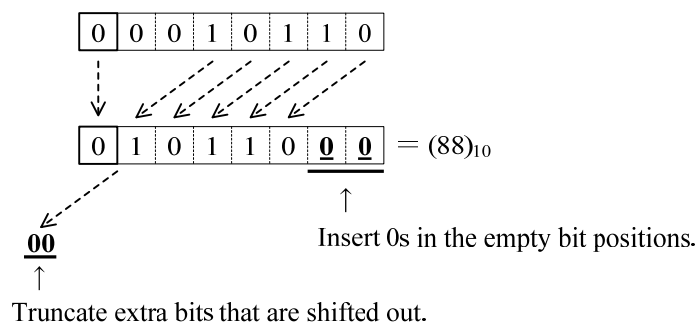
[Rules of arithmetic shift]

- Do not shift the sign bit.
- Truncate extra bits that are shifted out as a result of the shift.
- Store the following bit in the empty bit position that is created as a result of the shift.

In the case of left shift: 0

In the case of right shift: Same as the sign bit

Example: Shift $(22)_{10} = (00010110)_2$ arithmetically left by 2 bits.



Binary numbers have weight of power of 2 in each digit. Therefore, even for the same numeral 1, 1 as the second digit and 1 as the third digit have different meanings.

1 as second digit: $(10)_2 \rightarrow 2^1 = 2$

1 as third digit: $(100)_2 \rightarrow 2^2 = 4$

Because of this, computation rules of arithmetic shift can be summarized as follows:

[Computation rules of arithmetic shift]

- With an arithmetic shift to left by n bits, the value becomes 2^n times of the original number.
- With an arithmetic shift to right by n bits, the value becomes 2^{-n} times of the original number. (It is the value that is obtained by dividing the original number by 2^n)

(2) Logical shift

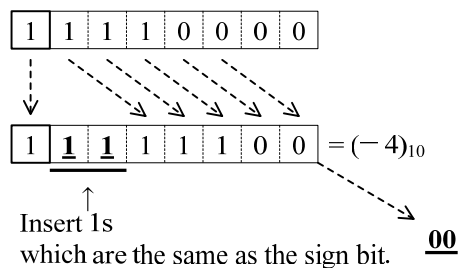
Logical shift is the shift operation used when the position of bits are changed. Its main difference from the arithmetic shift is that it does not treat the sign bit in a special manner.

[Rules of logical shift]

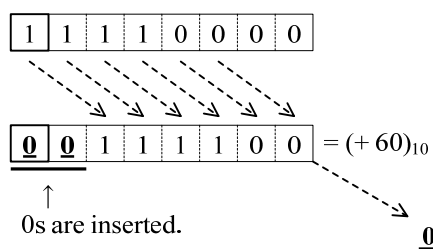
- Shift (move) the sign bit as well.
- Truncate extra bits that are shifted out as a result of the shift.
- Store 0 in the empty bit position that is created as a result of shift.

Example: Compare the results of 2-bit arithmetic shift right and 2-bit logical shift right for $(-16)_{10} = (11110000)_2$.

1) Perform 2-bit arithmetic shift right.

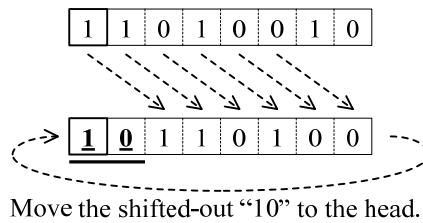


2) Perform 2-bit logical shift right.



Rotation shift (circular shift) is a type of logical shift. In rotation shift, the bits shifted out are circulated to the empty positions.

Example: Perform 2-bit rotation shift right on $(11010010)_2$.



3 Central Processing Unit and Main Memory Unit

Data processing in computers takes place in the steps of “Input → Processing → Output.” Section 1 explained the CPU (Central Processing Unit) that handles “Processing” in this series of steps is composed of a control unit and an arithmetic and logical unit. This section describes more detailed operating principles of the CPU and the main memory unit that is closely related to the CPU.

3 - 1 Configuration of CPU

In computers based on stored-program, programs (instructions) recorded in the main memory unit are read in the CPU one by one, and the control unit issues directions to each device on the basis of the contents of the instruction for processing. In order to carry out this operation, the CPU is composed of various devices and components.

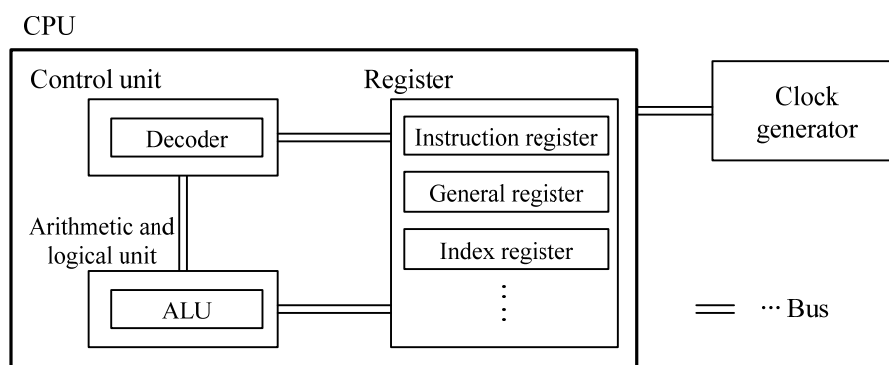


Figure 1-12 Components of CPU

(1) Control unit

Control unit is a unit that decodes the instruction to be executed and gives directions to each device. It is composed of a **decoder** (**instruction decoder** that interprets the instruction to be executed), etc.

(2) Arithmetic and logical unit

Arithmetic and logical unit is a unit that performs computations inside computers. It is composed of an **ALU (Arithmetic and Logical Unit)** that uses an **adder** for addition, a **complementer** for calculating complements, and such other components.

(3) Register

Register is a device used for temporarily storing various data inside the CPU. Different types of registers are available according to the type of data to be stored. The main registers (details of each register are explained later) are as follows:

- **Instruction register**

It stores the instructions to be executed. It is composed of an instruction part and an address part.

- **Instruction address register (program counter, program register)**

It stores the address (storing position in the main memory) of the instruction to be executed next.

- **General register**

It is used for various purposes such as storing the data to be processed.

- **Accumulator**

It stores the data for performing computations. It is sometimes also substituted with a general register.

- **Base address register**

It stores the beginning address of a program.

- **Index register**

It stores the index for address modification.

- **Flag register**

It stores the information of a certain status (whether the results of computations are positive or negative, etc.). Depending on the status of this register, the next action, such as branch destination of the condition branch instruction, is decided.

- **PSW (Program Status Word)**

It stores the running status of a program (value of program counter, value of flag register, etc.)

(4) Clock generator

Clock generator is a device for generating signals (clock signals) in order to synchronize and control the timing of operations between various devices inside computers. The speed of generating clock signals is shown with **clock frequency**, and **MHz (Mega Hertz)** (1 Million times in 1 second) is used as the unit. Usually, either the rising or falling edge of a signal is used as synchronization timing, and there is a technique called **DDR (Double Data Rate)** that uses both.

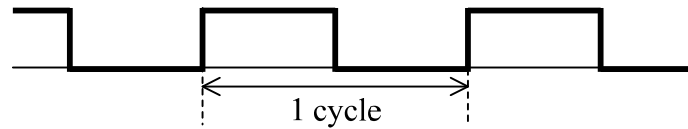


Figure 1-13 1 cycle of clock frequency

(5) Bus

Bus is the signal path for connecting various devices and registers, and transmitting data and control signals. Concerning the bus, there is a **serial bus** that sends data in the sequence of 1 bit each, and there is a **parallel bus** that simultaneously sends multiple bits. Performance of bus is decided by the **access mode**, such as the number of bits that can be sent in one clock signal (**bus width**) and clock signal, which is defined for each bus.

[Classification according to connection point]

1) **Internal bus (CPU internal bus)**

This is a bus used inside the CPU.

2) **External bus (CPU external bus)**

This is a bus that connects the CPU and the external devices. The clock frequency of the external bus is generally different from that of the CPU.

- **System bus**: Collective term for the buses directly connected from CPU to outside
- **Memory bus**: Bus that is mainly connected to the main memory unit
- **Input/output bus**: Bus that is mainly connected to the input/output devices

3) **Expansion bus**

This is a bus that connects the PC and the expansion cards.

[Classification according to usage]

1) **Address bus**: Bus for specifying the reference address to the main memory unit

2) **Control bus**: Bus for giving directions to each device from the control unit

3) **Data bus**: Bus for exchanging data

In conventional buses (**Neumann architecture**), the same bus was used for loading instructions and data. However, in modern-day computers, for loading instructions and data, **Harvard architecture** is used where independent buses are provided for loading instructions and data.

The main buses presently used in PCs are shown below. Other than in PCs, various buses are used (there are also manufacturers' specific buses).

- **PCI (Peripheral Component Interconnect)**

This is an external bus specification defined by Intel Corporation, USA. PCI specification of bus width 32 bits, bus clock 33 MHz, and transfer rate of 133 megabytes/second is common.

- **PCI Express (PCIe)**

This is an external bus specification defined by PCI-SIG to replace PCI. The transfer rate is 500 megabytes/second in full-duplex mode. The specification of "PCI Express x 16" where 16 transmitting lanes are contained is used as a replacement of the specification of AGP (Accelerated Graphics Port) that is used to connect graphics boards.

3 - 2 Main Memory Configuration

3-2-1 Memory Devices

Memory devices are devices that configure main memory and registers. In particular, IC-based memory devices are referred to as **semiconductor memory** or **IC memory**.

[Classification of memory devices]

- 1) **MOS (Metal Oxide Semiconductor) type**

Although it has a high degree of integration and low power consumption, it is a semiconductor device with somewhat low operating speed. These days, **CMOS (Complementary MOS)** are most commonly used after improvements were made so that operating speed could be increased by transporting electrical charges by the use of free electrons and holes.

- 2) **Bipolar type**

Although operating speed is high, this semiconductor device has a low degree of integration and power consumption is also large. A typical example of bipolar memory devices is a logic IC **TTL (Transistor-Transistor Logic)** composed of only bipolar transistors.

(1) RAM (Random Access Memory)

RAM is IC memory where reading and writing of data can be done freely. It is not suitable for storing data for a long time because it has a property (**volatility**) where data is cleared when the power is off.

- **SRAM (Static RAM)**

While its operating speed is high, it is expensive and it also has high power consumption. As for its memory mechanism, it uses a **flip-flop circuit** that continues to retain the preceding status, and electrical charge that records information can be retained as long as power is supplied. However, its disadvantages are a low degree of integration because of complex configuration of the circuit, and storage capacity that is smaller compared with DRAM. It is mainly used in registers or other memory devices.

- **DRAM (Dynamic RAM)**

While its operating speed is somewhat slow, it uses a simple circuit where electrical charge is retained by **condenser** or **capacitor**. Therefore, the degree of integration is high, and large capacity memory can be easily created at low cost. However, electrical charge that records information is lost over time, and therefore it is necessary to rewrite (**refresh**) the information. Examples include **SDRAM (Synchronous DRAM)** or **DDR SDRAM (Double Data Rate SDRAM)** used in main memory, and **RDRAM (Rambus DRAM)** that uses Rambus technology in the external bus interface.

(2) ROM (Read Only Memory)

ROM is IC memory that can be used only for reading data. It has a property (**non-volatility**) where data is not lost even when the power is off.

- **Mask ROM**

This is a type of ROM where users cannot write data. This memory is used to store programs or data in factories, and the information is used only for the purpose of reference.

- **User programmable ROM**

This is a type of ROM where users can write data. On the bases of the writing methods and restrictions on the number of rewritable times, this is classified as follows:

- **PROM (Programmable ROM):**

This is a type of ROM where user can write information only once.

- **UV-EPROM (UltraViolet-Erasable PROM):**

This is a type of ROM where data can be rewritten after information is erased by irradiating ultraviolet rays.

- **EEPROM (Electrically EPROM):**

This is a type of ROM where data can be rewritten after all or a part of information is electrically erased. It has limited life because of deterioration, and the number of rewritable times is restricted to a few tens of thousands of times to a few million times.

- Flash memory:

This is semiconductor memory where data can be rewritten after data is erased in units of blocks through electrical operations. Flash memory is a type of EEPROM. Therefore, the number of rewritable times is limited. However, it is used for various applications as a portable and convenient storage medium.

3-2-2 Components of Main Memory

In broad terms, the main memory unit is composed of three components.

- Memory unit

This contains memory cells (storage devices) that record data.

- Read/write feature

This reads and writes data in the recording area (collection of memory cells).

- Address selection feature

This interprets the specified address and selects the recording area of data.

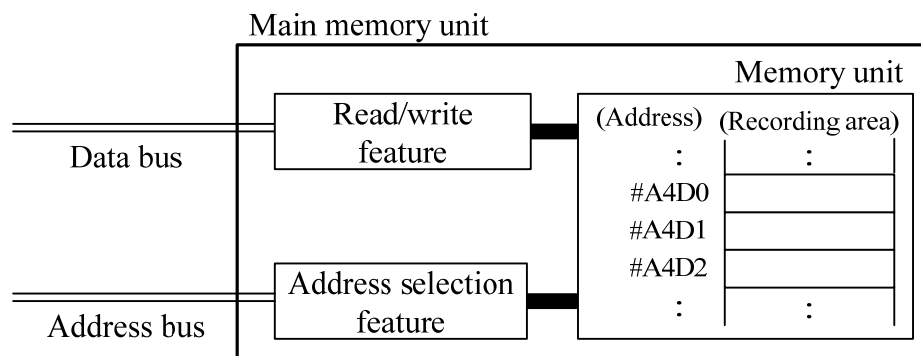


Figure 1-14 Components of main memory

Access operation to main memory is done as follows:

[Access operation to main memory]

- 1) Through address bus, the specified address is handed over to the address selection feature.
- 2) The address selection feature decodes the specified address by using the address decoder, and selects the recording area to be accessed. (address selection operation)
- 3) The read/write feature reads and writes data in the selected recording area. When data is read, data that is read from the recording area is passed on to the CPU via data bus. When data is written, data that is transferred from the CPU via data bus is written in the recording area.

3-2-3 Capacity Expansion of Main Memory

In the commercially available PCs, the capacity of main memory is fixed beforehand. Below are two methods of further expanding this capacity.

- **Extended memory (additional memory)**

This is a type of memory added in the expansion slots provided for in desktop PCs so that devices and electronic boards (components) can be added. Examples include **SIMM (Single In-line Memory Module)** where DRAM memory chips are consolidated and mounted on a small board, and **DIMM (Dual In-line Memory Module)**.

- **Memory card**

This IC memory is used in capacity expansion of a notebook PC, and a typical example of this is a flash memory-based memory card. It is standardized by **JEIDA (Japan Electronic Industry Development Association)** and **PCMCIA (Personal Computer Memory Card International Association)**.

3 - 3 Instruction and Addressing

3-3-1 Types and Configuration of Instructions

The CPU reads the instructions that are stored in main memory one by one, and interprets and executes the instructions. Instructions the CPU can interpret are **machine language instructions**, which are represented with combinations of 0 and 1, and a programming language (i.e., language used to write the programs) that is used to write such instructions is called the **machine language**.

Below are the main types of machine language instructions.

- **Arithmetic operation instruction**

This is an instruction for performing arithmetic operations such as addition and subtraction.

- **Logical operation instruction**

This is an instruction that performs logical operations such as logical product and logical sum operations.

- **Transfer instruction**

This is an instruction that transfers data such as load and store.

- **Comparison instruction**

This is an instruction that compares the magnitude relation between two values.

- **Branch instruction**

This is an instruction that branches (jumps) the control on the basis of the value of a flag register.

- **Shift instruction**

This is an instruction that performs shift operations such as arithmetic shift and logical shift.

- **Input/output instruction**

This is an instruction that reads data from or write data to I/O devices.

Moreover, the configuration of machine language instructions includes instructions recorded in 1 word (recording area corresponding to 1 address) of main memory (**1-word instructions**) and instructions recorded consecutively in multiple words (when recorded in 2 words, it is referred to as **2-word instructions**). However, even if the length of instructions is different, there is a similarity in the sense that both of them are composed of an **instruction part** where the instruction code specifying the process to be executed is recorded, and an **address part** (or **operand part**) for specifying the address to be processed.



Figure 1-15 Configuration of machine language instructions

Based on the number of address parts (operand parts), they are also separately referred to as **0-address instruction**, **1-address instruction**, **2-address instruction**, and **3-address instruction**.

3-3-2 Execution Sequence of Instructions

Figure 1-16 shows the general execution sequence of instructions. Here, the process until fetching the instruction is called the **instruction fetching stage (fetch cycle)**, and the process until decoding and executing the instruction is called the **instruction execution stage (execution cycle)**.

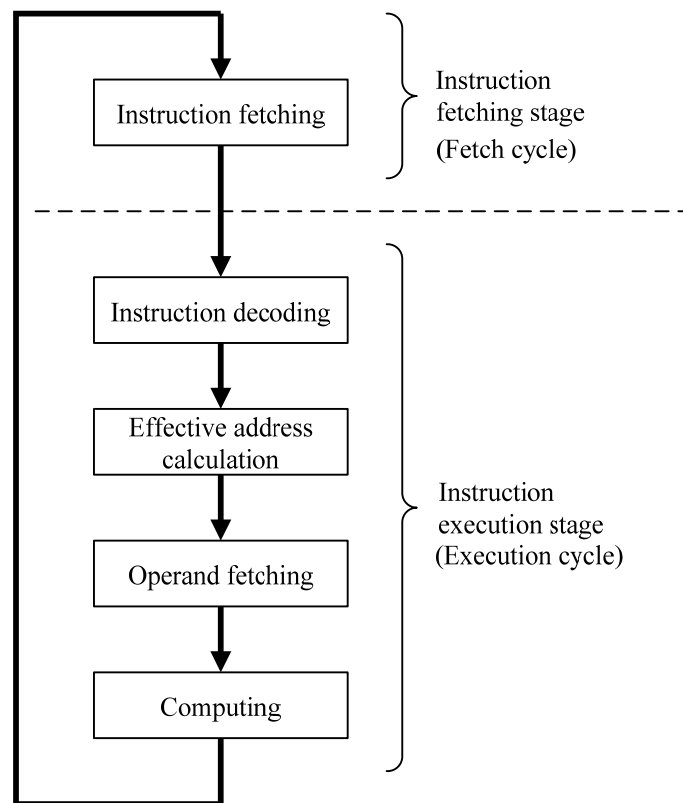


Figure 1-16 Execution sequence of instructions

Details of the processing sequence from “instruction fetching” until “computing” are explained in order by using the following figure.

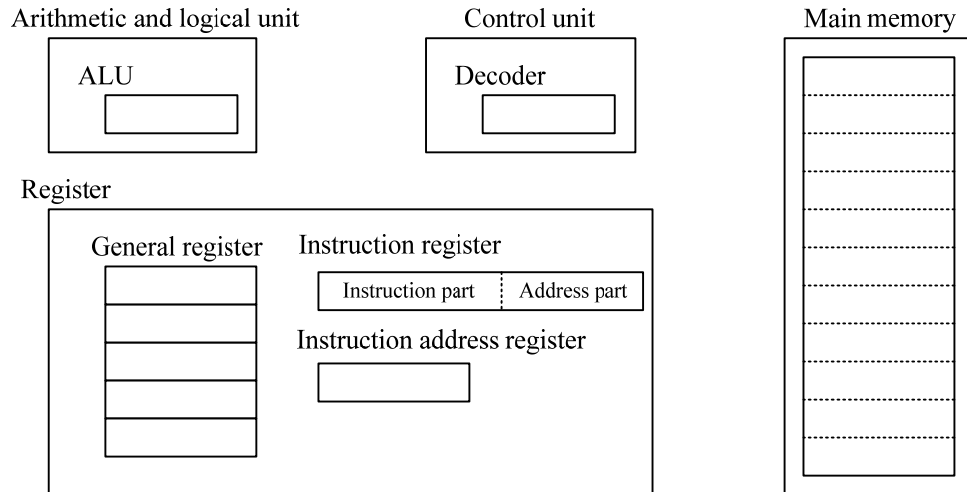


Figure 1-17 CPU and main memory

(1) Instruction fetching

In **instruction fetching**, on the basis of the directions given by the control unit, an instruction stored in the address shown by the instruction address register is fetched from main memory, and it is stored in the instruction register. After the instruction is fetched, instruction word length L (number of words where 1 instruction fetched is stored) is added to the instruction address register for fetching the next instruction.

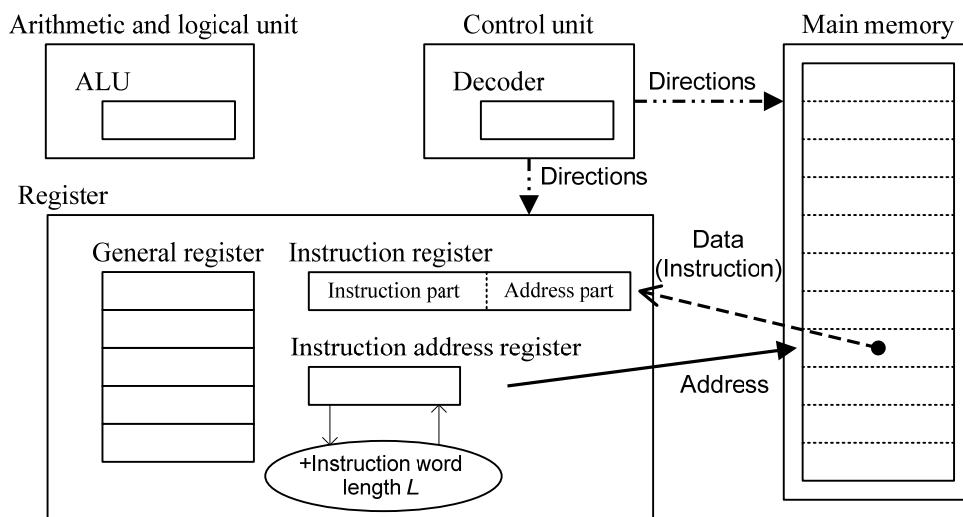


Figure 1-18 Instruction fetching

(2) Instruction decoding

In **instruction decoding**, the instruction part of the instruction fetched in the instruction register is decoded by the decoder (instruction decoder) of the control unit.

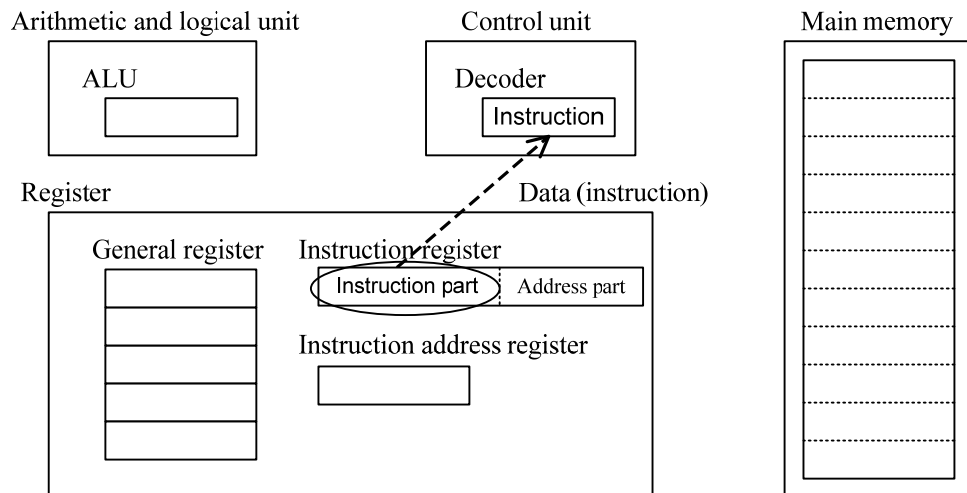


Figure 1-19 Instruction decoding

(3) Effective address calculation/operand fetching

In **effective address calculation**, the storage position (**effective address**) of data stored in main memory is determined from the address part of the instruction. (This is referred to as **address modification**)

In **operand fetching**, the effective address calculated is sent to main memory, and the value or variable (**operand**) to be computed such as arithmetic operation instructions is read into the general register. There may not be any operand in some cases, depending on the type of instruction, and this process may be omitted in such cases.

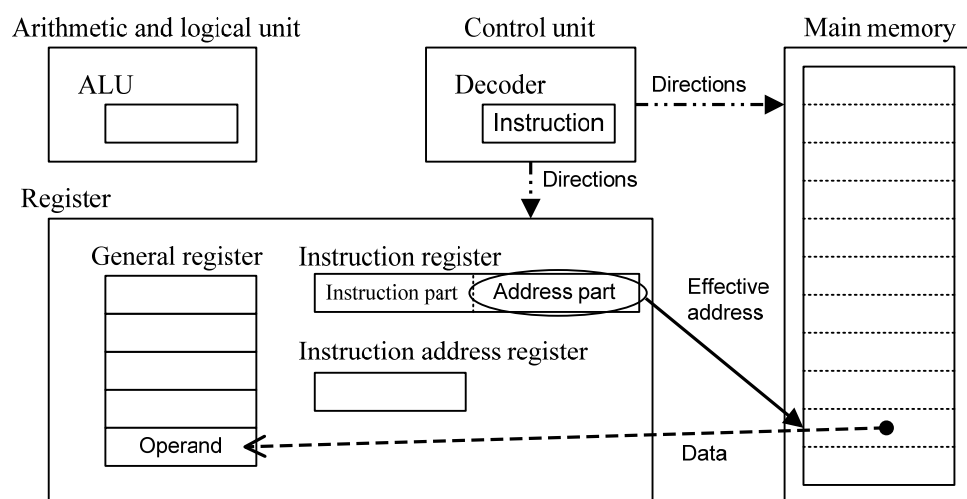


Figure 1-20 Effective address calculation and operand fetching

(4) Computation (instruction execution)

In computation (instruction execution), the arithmetic and logical unit executes the

computation based on the decoded instruction. The operation result is recorded in the general register and written in main memory.

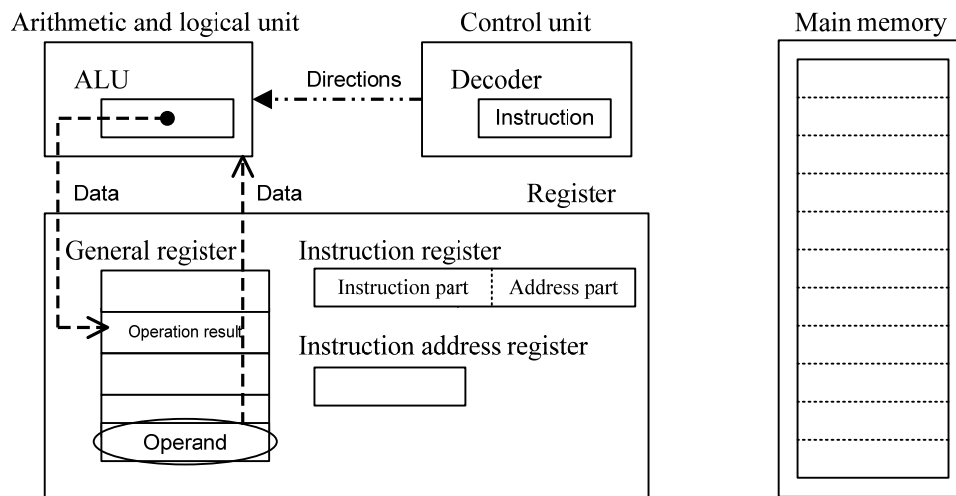


Figure 1-21 Arithmetic operation (instruction execution)

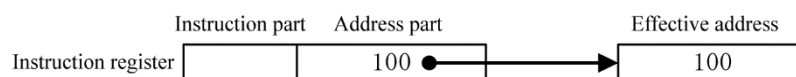
3-3-3 Addressing Mode

Addressing mode is the method of determining the effective address from the value recorded in the address part of instruction, and then fetching the operand. The addressing mode is classified into two methods: one is the method (**address modification**) of determining the effective address from the value of the address part of instruction and the other is the method of determining the operand from the effective address.

(1) Method of determining effective address from the value of the address part (address modification)

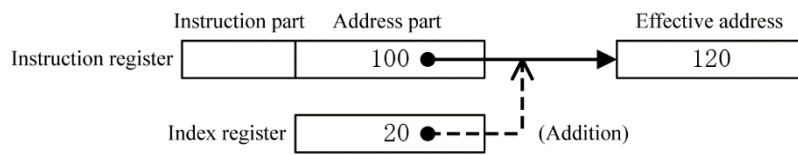
1) **Absolute addressing**

In this method, the value of the address part is used as the effective address as it is.



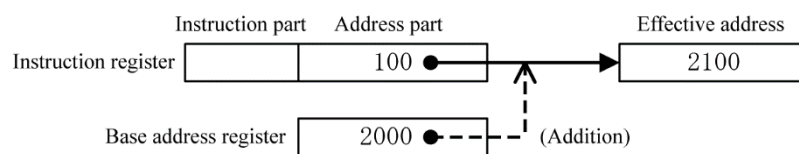
2) **Index addressing**

In this method, the effective address is determined by adding the value of the index register to the value of the address part. A general register is also used as the index register, and it is necessary to specify which general register to use in such cases.



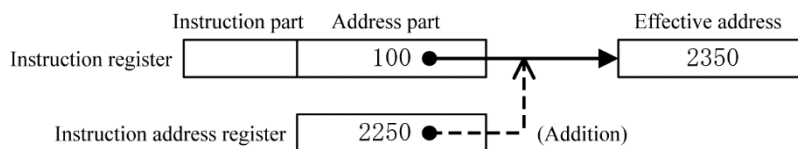
3) Base addressing

In this method, the effective address is determined by adding the value of the base address register to the value of the address part. This means the relative position from the beginning of the program.



4) Relative addressing

In this method, the effective address is determined by adding the value of the instruction address register (program counter) to the value of the address part. This means the relative position from the instruction being executed.



(2) Method of determining operand from the effective address

1) Immediate addressing

In this method, the effective address is used as it is as operand. Since it does not reference main memory, execution speed is somewhat faster than other addressing.

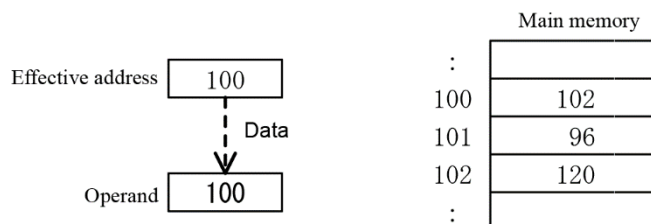


Figure 1-22 Immediate addressing

2) Direct addressing

In this method, the content (value) of main memory referenced by the effective address is used as operand. Generally, direct addressing where there is no description concerning address modification can be considered as absolute addressing, and it can be considered as direct addressing when only address modification is specified.

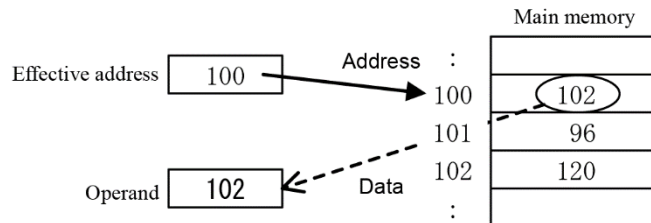


Figure 1-23 Direct addressing

3) Indirect addressing

In this method, the content (value) of main memory referenced by the effective address is used as the address of operand. Double (in the example shown in Figure 1-24, data stored in address 120 is fetched as operand), and triple indirect addressing is also possible.

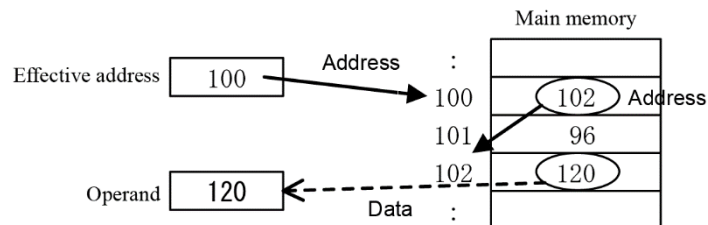


Figure 1-24 Indirect addressing

3-3-4 Interrupt

Interrupt means executing a separate process (instruction) while a series of processes (instructions) are being executed. Instead of **user mode** (mode where there are restrictions on the use of CPU), which is normally used, the interrupt process is executed in **privilege mode** (mode where there are no restrictions on the use of CPU).

[Processing sequence of processor when interrupt has occurred]

- 1) Switch from user mode to privilege mode.
- 2) Save the values of various registers (program counter, etc.).
- 3) Decide the starting address of the interrupt process routine (interrupt program).
- 4) Execute the interrupt process routine.
- 5) After the execution of the interrupt process routine is complete, restore the values of various registers that are saved in step 2.
- 6) Switch from privilege mode to user mode, and restart the interrupted process.

According to conditions that an interrupt occurs, this can be classified described below. Here, it is assumed that multiple interrupts may occur simultaneously, and therefore priority ranking is assigned to each interrupt.

- 1) **External interrupt:** This is an interrupt that occurs because of a reason not related to the process being executed.

- **Timer interrupt**

This is an interrupt that occurs when the time measured in the timer, such as interval timer and watchdog timer, has exceeded the specified time (i.e., the timer has been timed-out).

- **Input/output interrupt** (input/output completion interrupt)

This is an interrupt that occurs when an input/output operation has been completed.

- **Machine check interrupt**

This is an interrupt that occurs because of a hardware malfunction, a power failure, or such other factor.

- **Restart interrupt**

This is an interrupt that occurs when the user has pressed the external restart switch.

- 2) **Internal interrupt:** This is an interrupt that occurs because of the process being executed. This is also referred to as a trap.

- **SVC (SuperVisor Call) interrupt**

This is an interrupt that occurs when **supervisor** (program that offers basic functions) is requested to invoke a process, such as when input/output instruction is used or storage protection exception happens.

- **Program interrupt**

This is an interrupt that occurs by the error of a running program (e.g., divide-by-zero or overflow)

3 - 4 Circuit Configuration of ALU

ALU (Arithmetic and Logical Unit) is a unit that performs arithmetic operations and logical operations. This unit is composed of various circuits.

3-4-1 Logic Circuit

Logic circuit is a circuit that performs **logical operations** for logical processing of instructions by computers. Logical operations refer to arithmetic operations that have only two truth values of True (1) and False (0).

[Main logical operations]

1) **Logical product operation (AND)**

The output becomes True (1) when both input values are True (1), or else the output is False (0) (when either one is False (0)).

2) **Logical sum operation (OR)**

The output becomes False (0) when both input values are False (0), or else the output is True (1) (when either one is True (1)).

3) **Negation operation (NOT)**

The output becomes False (0) when the input value is True (1), and the output is True (1) when the input value is False (0).

4) **Exclusive logical sum (Exclusive OR) operation (XOR or EOR)**

The output becomes False (0) when both input values are the same, and the output is True (1) when both input values are different.

The table below summarizes the relations between input and output of the respective logical operation. This kind of table is referred to as a **truth table**.

Input		AND	OR	NOT	XOR
X	Y	(X AND Y)	(X OR Y)	(NOT X)	(X XOR Y)
True (1)	True (1)	True (1)	True (1)	False (0)	False (0)
True (1)	False (0)	False (0)	True (1)	False (0)	True (1)
False (0)	True (1)	False (0)	True (1)	True (1)	True (1)
False (0)	False (0)	False (0)	False (0)	True (1)	False (0)

Besides “X AND Y”, AND is also expressed as “ $X \cdot Y$ ” and “ $X \wedge Y$ ”; besides “X OR Y”, OR is also expressed as “ $X + Y$ ” and “ $X \vee Y$ ”; besides “NOT X”, NOT is also expressed as “ \bar{X} ” and “ $\neg X$ ”; and besides “X XOR Y”, XOR is also expressed as “ $X \oplus Y$ ” and “ $X \nabla Y$ ”.

(1) Sequential circuit

Sequential circuit is a logic circuit where the output is decided on the basis of the input at that time and earlier status. **Flip-flop circuit**, which is one of the main sequential circuits, has two stable states, and it is used in storage cells of SRAM. Figure 1-25 shows the circuit diagram of a flip-flop for reference and the truth table. Circuit symbols used in Figure 1-25 are one of the methods of representing electronic circuits, and it is referred to as **MIL (Military Specifications and Standards) symbols**.

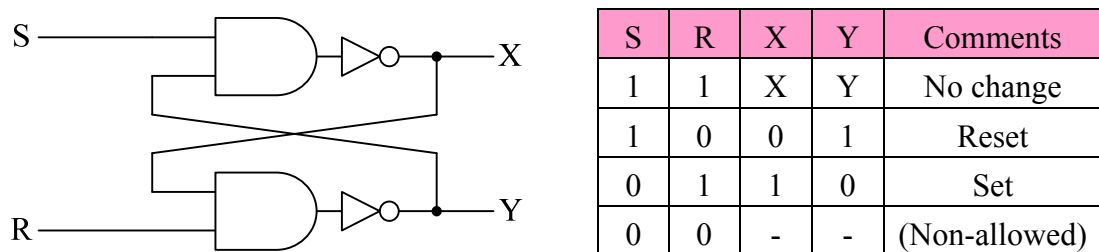


Figure 1-25 Circuit diagram of flip-flop and truth table

(2) Combination circuit

Combination circuit is a logic circuit where the output is decided on the basis of the input at that time. There are various types of combination circuits, from basic circuits for implementing basic logical operations (logical product, logical sum, negation) to circuits formed by combining these basic circuits.

1) AND circuit

This circuit performs logical product operation (AND). The truth table and MIL symbol of AND circuit are as shown below.

X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

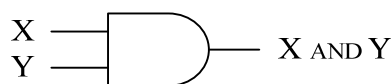


Figure 1-26 AND circuit

2) OR circuit

This circuit performs logical sum operation (OR). The truth table and MIL symbol of OR

circuit are as shown below.

X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

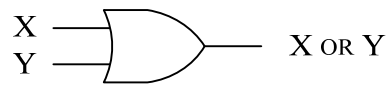


Figure 1-27 OR circuit

3) NOT circuit

This circuit performs negation operation (NOT). The truth table and MIL symbol of NOT circuit are as shown below.

X	NOT X
0	1
1	0

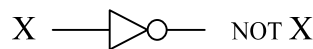


Figure 1-28 NOT circuit

4) NAND circuit

This circuit performs **negative logical product (negative AND) operation (NAND)**. NAND operation is the arithmetic operation that negates the results of AND operation. The truth table and MIL symbol of NAND circuit are as shown below. NAND circuit is formed by combining AND circuit and NOT circuit.

X	Y	X NAND Y
0	0	1
0	1	1
1	0	1
1	1	0

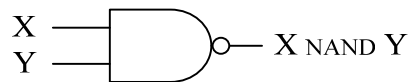


Figure 1-29 NAND circuit

5) NOR circuit

This circuit performs **negative logical sum operation (NOR)**. NOR operation is the arithmetic operation that negates the results of OR operation. The truth table and MIL symbol of NOR circuit are as shown below. NOR circuit is formed by combining OR circuit and NOT circuit.

X	Y	X NOR Y
0	0	1
0	1	0
1	0	0
1	1	0

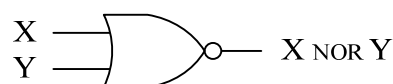


Figure 1-30 NOR circuit

6) XOR circuit

This circuit performs exclusive logical sum operation (XOR). The truth table and MIL symbol of XOR circuit are as shown below.

X	Y	X XOR Y
0	0	0
0	1	1
1	0	1
1	1	0

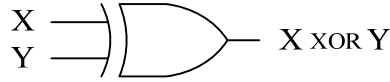


Figure 1-31 XOR circuit

XOR circuit is also formed by combining the basic circuits. However, it cannot be formed with a combination of simple circuits as in the case of NAND circuits and NOR circuits. In such cases, it is called **logical design** (or **circuit design**) to think about the combination of basic circuits for achieving the targeted results (truth values).

In logical design, **logical expressions** (**logical functions**) obtained from the truth tables are simplified using the rules of logical operations (**logical laws**), and the optimum combination of basic circuits is determined. (“Optimum” means that the design is performed in consideration of performance, efficiency, and cost.

[Main logical laws]

* AND operation is represented with \wedge , OR operation is represented with \vee , and NOT operation is represented with \neg .

- **Idempotent laws**

$$A \vee A = A$$

$$A \wedge A = A$$

- **Commutative laws**

$$A \vee B = B \vee A$$

$$A \wedge B = B \wedge A$$

- **Associative laws**

$$A \vee (B \vee C) = (A \vee B) \vee C$$

$$A \wedge (B \wedge C) = (A \wedge B) \wedge C$$

- **Distributive laws**

$$A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$$

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

- **Absorption laws**

$$A \vee (A \wedge B) = A$$

$$A \wedge (A \vee B) = A$$

- **De Morgan's laws**

$$\neg(A \vee B) = (\neg A) \wedge (\neg B)$$

$$\neg(A \wedge B) = (\neg A) \vee (\neg B)$$

- **Others**

$$A \vee 0 = A$$

$$A \wedge 0 = 0$$

$$A \vee 1 = 1$$

$$A \wedge 1 = A$$

$$A \vee (\neg A) = 1$$

$$A \wedge (\neg A) = 0$$

$$\neg(\neg A) = A \quad (\text{Returns to original by double negation})$$

[Logical design of XOR circuit]

1) Focus on the parts where the output of the truth table is 1 and determine AND operation where the result is 1 on the basis of the input at that time.

- When $(X=0, Y=1)$: $\neg X \wedge Y$

- When $(X=1, Y=0)$: $X \wedge \neg Y$

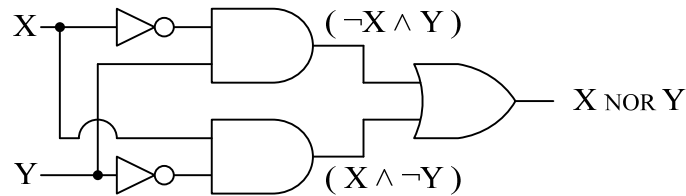
2) Create a circuit where the output will be 1 when either of the AND operations you determined is 1. In order to design such a circuit, determine logical expression F

where two AND operations are combined with OR operation.

Logical expression $F = (\neg X \wedge Y) \vee (X \wedge \neg Y)$

3) Determine the circuit diagram by combining the basic circuits such that it represents logical expression F.

[Circuit diagram]



By expanding the logical expression F determined in [Logical design of XOR circuit] using logical laws, we can determine different logical circuits.

Logical expression $F = (\neg X \wedge Y) \vee (X \wedge \neg Y)$

$= ((\neg X \wedge Y) \vee X) \wedge ((\neg X \wedge Y) \vee \neg Y)$...Distributive law

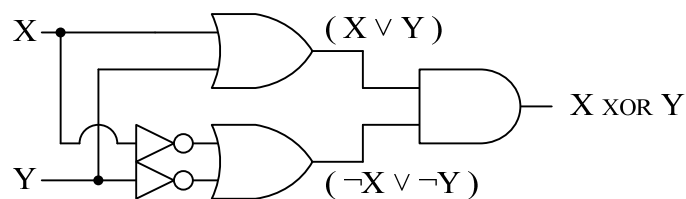
$= ((\neg X \vee X) \wedge (Y \vee X))$

$\wedge ((\neg X \vee \neg Y) \wedge (Y \vee \neg Y))$...Distributive law

$= (1 \wedge (Y \vee X)) \wedge ((\neg X \vee \neg Y) \wedge 1)$... $A \vee (\neg A) = 1$

$= (X \vee Y) \wedge (\neg X \vee \neg Y)$... $A \wedge 1 = A$, Commutative law

[Circuit diagram]



In the actual logical design, it is common to use NAND circuit, NOR circuit, or XOR circuit as one circuit (basic circuit) as it is.

3-4-2 Arithmetic Operation Circuit

In a computer, subtraction can be performed with addition using complement notation. Similarly, the four basic arithmetic operations can be implemented with addition only. This is because multiplication can be performed with repeated addition, and division can be performed with repeated subtraction (i.e. repeated addition). Therefore, in order to implement the four basic arithmetic operations in a computer, only an adder circuit and a complement

circuit are required.

First, the adder circuit is considered here. The basis of addition in a computer is 2-bit addition that adds one bit to one bit. Here, then, a circuit to perform 2-bit addition is considered. As 2-bit addition, the following four types are possible.

$$\begin{array}{r}
 X: \quad \quad 0 \quad \quad 0 \quad \quad 1 \quad \quad 1 \\
 Y: \quad + \quad 0 \quad + \quad 1 \quad + \quad 0 \quad + \quad 1 \\
 \hline
 \quad 0 \ 0 \quad 0 \ 1 \quad 0 \ 1 \quad 1 \ 0
 \end{array}$$

The results of the addition of two 1-bit binary numbers are shown below.

[Results of the addition of two 1-bit binary numbers]

Values to be added		Operation result	
X	Y	Carry (c)	Sum (s)
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

From this table, it is clear that carry is 1 only when two entered values are both 1, this is a logical product operation (AND). It is also clear that sum is 0 when two entered values are the same and 1 when they are different, and therefore, this is an exclusive logical sum operation (XOR). Therefore, a circuit diagram with the configuration below is derived. This circuit is called a **half adder**.

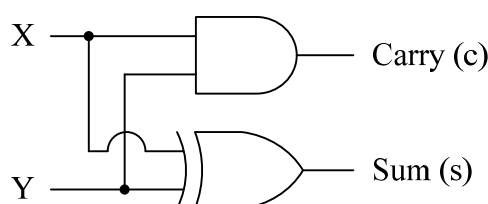


Figure 1-32 Circuit diagram of half adder

Adder circuit of a computer is configured with this half adder as the basis. However, in half adder, carry from low order is not taken into account. Therefore, arithmetic operation of numerical values represented with multiple bits cannot be performed. Because of that, another adder circuit of 3 bits including carry from low order becomes necessary.

The table below summarizes the results of arithmetic operation of 3-bit addition considering carry (c₀) from low order.

[Results of arithmetic operation of 3-bit addition]

Values to be added			Operation result	
X	Y	c0	Carry (c1)	Sum (s)
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
1	0	0	0	1
0	1	1	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

In order to derive circuit configuration from this table, we will get the following if we think about logical expression F1 for determining carry (c1) and logical expression F2 for determining sum (s).

[Logical expression F1 for determining carry]

- 1) With the input for which the output of truth table will become 1, determine AND operations where the result will be 1.
 - When (X=0, Y=1, c0=1): $\neg X \wedge Y \wedge c0$
 - When (X=1, Y=0, c0=1): $X \wedge \neg Y \wedge c0$
 - When (X=1, Y=1, c0=0): $X \wedge Y \wedge \neg c0$
 - When (X=1, Y=1, c0=1): $X \wedge Y \wedge c0$
- 2) Determine logical expression F1 by combining AND operations we determined with OR operations.

$$\text{Logical expression } F1 = (\neg X \wedge Y \wedge c0) \vee (X \wedge \neg Y \wedge c0) \vee (X \wedge Y \wedge \neg c0) \vee (X \wedge Y \wedge c0)$$

[Logical expression F2 for determining sum]

- 1) With the input for which the output of truth table will become 1, determine AND operations where the result will be 1.
 - When (X=0, Y=0, c0=1): $\neg X \wedge \neg Y \wedge c0$
 - When (X=0, Y=1, c0=0): $\neg X \wedge Y \wedge \neg c0$
 - When (X=1, Y=0, c0=0): $X \wedge \neg Y \wedge \neg c0$
 - When (X=1, Y=1, c0=1): $X \wedge Y \wedge c0$
- 2) Determine logical expression F2 by combining the AND operations we determined with OR operations.

$$\text{Logical expression } F2 = (\neg X \wedge \neg Y \wedge c0) \vee (\neg X \wedge Y \wedge \neg c0) \vee (X \wedge \neg Y \wedge \neg c0) \vee (X \wedge Y \wedge c0)$$

$$\vee (X \wedge \neg Y \wedge \neg c_0) \vee (X \wedge Y \wedge c_0)$$

By expanding (simplifying) these logical expressions F1 and F2 by using logical laws, we can derive the configuration of a 3-bit adder circuit.

However, even expansion of logical expression with this method is difficult, and it can be anticipated that configuration of the circuit will become complicated. Therefore, we will use half adders after 3-bit addition is divided into 2-bit addition.

[Dividing 3-bit addition into 2-bit addition]

- 1) Add X and Y, and determine carry (c') and sum (s').
- 2) Add s' and c_0 , and determine carry (c'') and sum (s''). s'' determined here will be sum (s) in overall arithmetic operation.
- 3) If there is any carry in this addition, then carry will occur for overall arithmetic operation. Therefore, carry (c_1) in overall arithmetic operation is determined with OR operation of c' and c'' . (Carry will never be 1 in both the operations)

$$\begin{array}{r}
 X \\
 + Y \\
 \hline
 c' \quad s' \\
 + c_0 \\
 \hline
 c'' \quad s''
 \end{array}$$

Carry (c_1) = $c' \text{ OR } c''$ Sum (s) = s''

Full adder shown in Figure 1-33 is the adder circuit configured on the basis of this concept. In a computer, the number of full adders equal to the number of bits of arithmetic operation is arranged to form the arithmetic operation circuit.

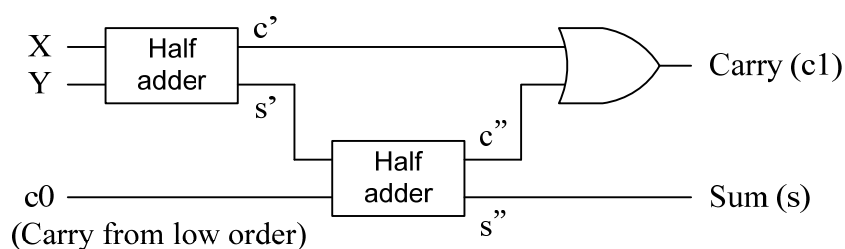


Figure 1-33 Circuit diagram of full adder

Next, let us think about the complement circuit. In order to perform arithmetic operations (four basic arithmetic operations) with only adders (half adder/full adder), we will need a circuit for determining complement (**complementer**) in order to perform subtraction by using addition. Inside a computer, binary arithmetic operations are performed. Therefore, it is common to use two types of complementers, namely, **1's complementer** that determines 1's

complement, and **2's complementer** that determines 2's complement. 1's complement in binary numbers is determined by inverting the original bit sequence (0 is inverted to 1, and 1 is inverted to 0). Therefore, 1's complementer can be formed only with NOT circuit. On the other hand, there are various methods of forming 2's complementer. However, the concept of using an adder to add 1 to 1's complement determined using 1's complementer is a relatively easy method.

3 - 5 High Speed Technologies

High speed technologies are technologies that increase the operating speed (processing capacity) of computers by creating new mechanisms using each device.

3-5-1 High Speed Memory Access

Memory is the term that means a device or a medium that records data, and when simply referred to as “memory,” it mostly means the main memory unit (**main memory**).

High speed memory access is a technology that resolves bottlenecks caused by difference in access speeds (access gap) by even slightly accelerating access to main memory (DRAM), which has slower access speed compared with the register (SRAM) of the processor.

(1) Buffer memory

Buffer memory is the medium-speed storage installed between high-speed storage and low-speed storage. **Cache memory**, which is a buffer memory, is the medium-speed storage composed of SRAM and installed between high-speed register and low-speed main memory.

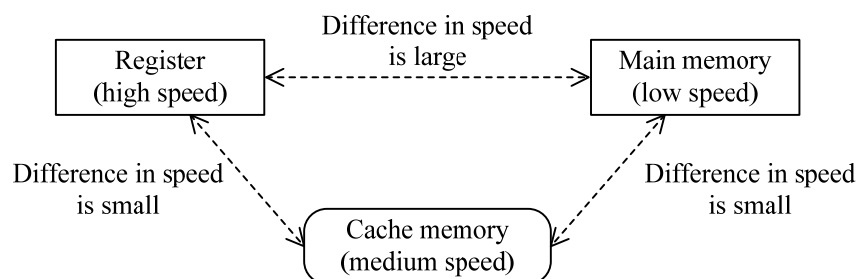


Figure 1-34 Working of cache memory

In a computer system where cache memory is used, if the data required by the processor is available (when it is hit) in cache memory, then cache memory is accessed. On the other hand, when the data is not there (when it is mishit) in cache memory, main memory is accessed (in precise terms, on the basis of the directions of hardware or OS (software for controlling the computer), the data is loaded from main memory into the corresponding block of cache

memory). In other words, **average access time (effective access time)** as seen from the processor is decided on the basis of the extent to which the required data is found in cache memory. Therefore, it is determined by **weighted average** of access time of cache memory and access time of main memory. Here, the access time is determined by considering both the probability (**hit ratio**) that the required data is available in cache memory and the probability (**NFP (Not Found Probability)**) that the required data is not available in cache memory. “Hit ratio + NFP = 1” always holds true.

Example: Calculate the average access time of the following processor. NFP (probability that the required data is not available in cache memory) is 0.1.

Access destination	Access time (nanosecond)
Cache memory	50
Main memory	500

Average access time of processor

$$\begin{aligned}
 &= \text{Access time of cache memory} \times \text{hit ratio} \\
 &\quad + \text{Access time of main memory} \times \text{NFP} \\
 &= 50 \text{ nanoseconds} \times (1 - 0.1) + 500 \text{ nanoseconds} \times 0.1 \\
 &= 95 \text{ nanoseconds}
 \end{aligned}$$

From this example, it is clear that the higher the hit ratio is, the shorter the average access time is because of increased access to cache memory. When **Harvard architecture**, which independently loads instructions and data, is used, cache memory is also separated into **instruction cache** and **data cache**. In this case, creating a program consolidating process (instruction) sections that are frequently executed will increase the hit ratio of instruction cache, which may shorten the average access time (access speed will improve).

Moreover, when cache memory is used, data inside cache memory is updated (changed). However, original data is stored in main memory, so it is necessary to reflect the data updated inside cache memory (write it in main memory). Below are the two writing methods according to the timing of reflecting data update.

[Writing methods from cache memory]

1) Write-through method

This method reflects the update in main memory whenever data is updated. While certainty is high, main memory needs to be accessed for every process, so operating speed is somewhat slow.

2) Write-back method

In this method, data is not updated in main memory while data is updated in cache memory. Data is updated in main memory only when data from cache memory is removed. While its operating speed is fast, inconsistency in the data of main memory may occur at some point in time.

In modern-day computers, there is a large difference in the speed of processor and the speed of main memory. Therefore, it is common to use multi-level cache configuration where cache memory is divided into 2 levels, namely, **primary cache** and **secondary cache**. In this case, the processor will access in the sequence of primary cache → secondary cache → main memory.

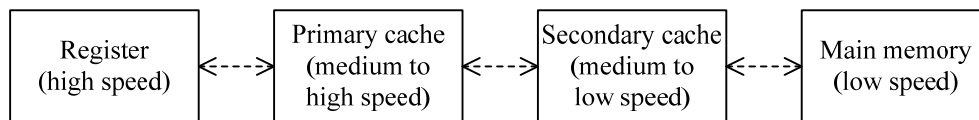


Figure 1-35 Multi-level cache configuration

Meanwhile, there is **disk cache**, which is another buffer memory. This buffer memory is placed between main memory and auxiliary storage (hard disk), and it may use a part of main memory, or semiconductor memory may be installed separately.

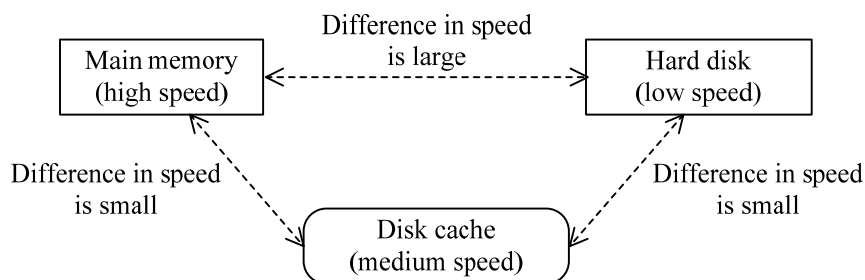


Figure 1-36 Working of disk cache

Basically, the higher the access speed of storage, the smaller the storage capacity. Therefore, by combining high-speed low-capacity storage and low-speed high-capacity storage as shown in Figure 1-37, we can configure high-speed high-capacity storage on an overall basis. This concept (concept where each storage is tiered on the basis of access speed and capacity) is

referred to as **memory hierarchy**.

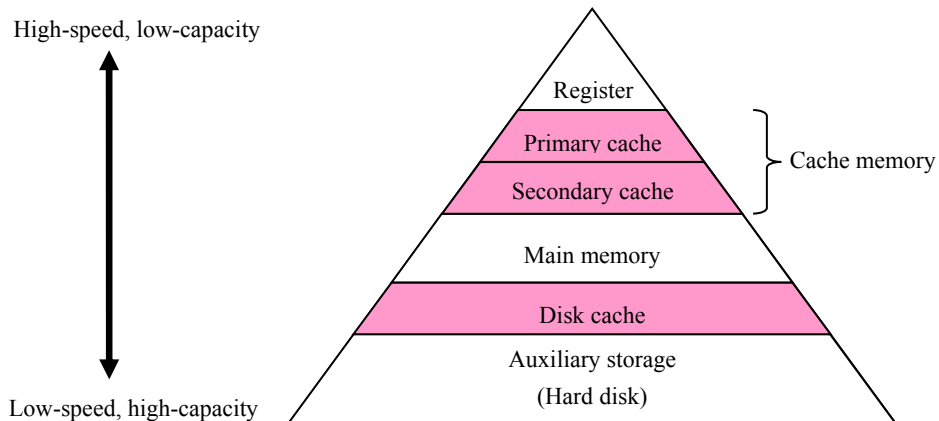
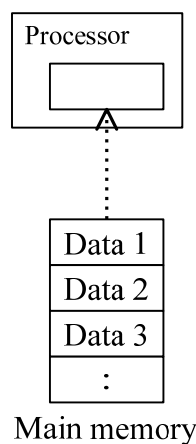


Figure 1-37 Memory hierarchy

(2) Memory interleave

Memory interleave is a method where main memory is divided into several sections (**bank**) that are simultaneously and independently accessed in order to improve the average access time of main memory.

Conventional access scheme



Memory interleave scheme

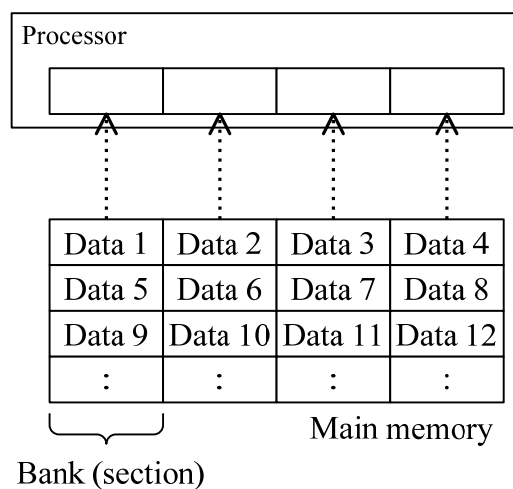


Figure 1-38 Memory interleave method

Memory interleave is a method of reading the data in advance, and it is useful for the processing of data that is located in the contiguous area of main memory. However, data may not be referenced in the sequence of storage area at all times, and the average access time may not be always proportionate to the number of banks (even if there are 4 banks, the average

access time may not be 1/4).

3-5-2 Speeding-up of Processor

The speeding-up of processors is a technology for increasing the operating speed of processors. It includes various high-speed techniques starting such as a technique for enhancing the performance of the processor itself and a technique for enhancing the apparent operating speed by managing the execution procedure of instructions.

(1) Multi-core processor

In a **multi-core processor**, two or more processors cores are integrated on one chip (LSI). A processor with only one processor core on one chip (LSI) is referred to as a **single-core processor**. The multi-core processor enhances processing performance by allocating processes to multiple processor cores (if n -core processors are integrated, the processing performance will become approximately n times greater). Moreover, in comparison with mounting multiple single-core processors, the processing performance of the processor itself can be increased while power consumption is minimized. However, when multiple processor cores operate simultaneously, contention for shared resources may occur.

(2) Pipeline control

Pipeline control (or the pipeline processing) divides the execution cycle of each instruction into multiple stages, and by shifting each stage little by little and executing multiple stages separately, multiple instructions are executed simultaneously and in parallel.

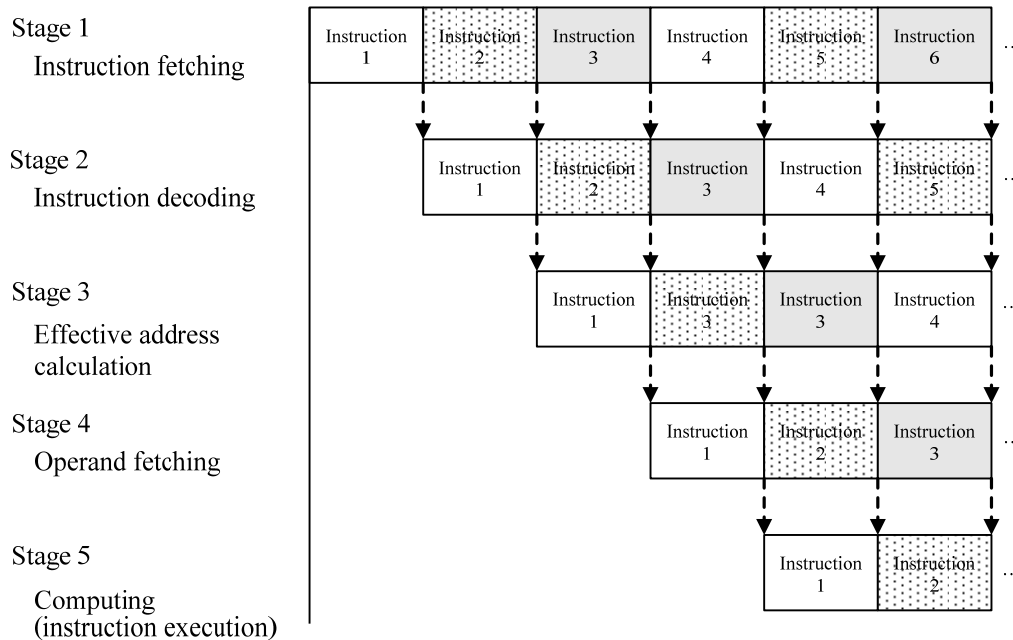


Figure 1-39 Pipeline control

Pipeline control is a method of advanced control of instructions. Therefore, efficiency will decline if there are instructions (jump instructions) that change the execution sequence of instruction. Moreover, if the execution time of each instruction is different, the waiting time may be required before starting the execution of a stage, and efficiency will decline.

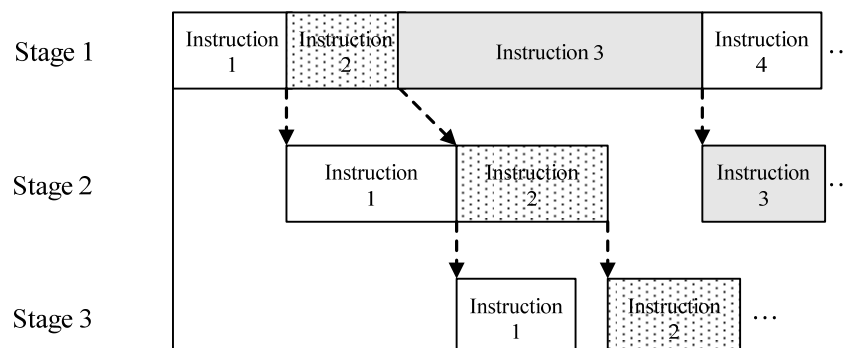


Figure 1-40 Pipeline control with poor efficiency

For efficient pipeline control, it is necessary that the execution time of all instructions be almost the same. Among the different architectures of processors (computers), **RISC** architecture is suitable for this concept, while **CISC** architecture is not very suitable.

[Processor architecture]

1) **RISC (Reduced Instruction Set Computer)**

Reduced instruction set computer - this is a computer where, by reducing the instruction set (collection of instructions) to the basic instructions with high frequency of use in order to fix the instruction word length, the execution time for a

single instruction is shortened and unified as much as possible. Instructions are executed by hardware (wired logic).

- **Wired logic control:**

Instruction is executed by passing signals through a logic circuit having a certain function. Since logic is formed by connecting signal lines for transmitting signals, it is also called **hard-wired logic control**.

2) **CISC (Complex Instruction Set Computer)**

Complex instruction set computer - this computer implements complex instructions by using software (microprograms), and therefore it can handle multifunction instructions. However, since the length and execution time of each instruction are largely different, it is not suitable for pipeline control.

- **Microprogram control:**

Micro instruction refers to giving control directions to hardware such as various logic circuits and registers. This method executes instructions with a microprogram where these micro instructions are combined. It is also referred to as **stored logic control**.

In pipeline control, the execution cycle of instructions is divided into stages that mean “instruction fetching,” “instruction decoding,” “effective address calculation,” “operand fetching,” and “computing.” The method where this division is further broken down is referred to as **super-pipeline**. Super-pipeline increases the number of stages (depth of pipeline) and thereby reduces the processing time (pipeline pitch) of one stage in order to aim for performance enhancement.

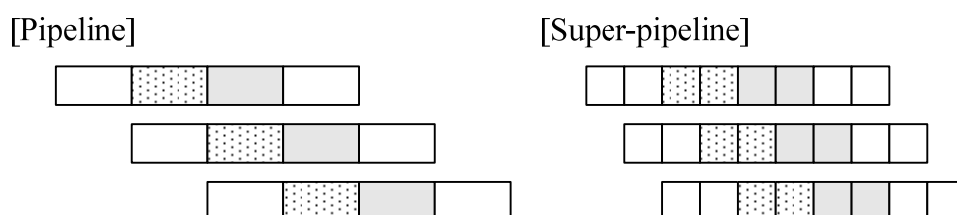


Figure 1-41 Comparison image of pipeline and super-pipeline

(3) Superscalar/VLIW (Very Long Instruction Word)

Superscalar and **VLIW** aim to enhance performance of the processor by simultaneously executing multiple instructions.

At the stage of executing a program, **superscalar** extracts multiple instructions that can be executed simultaneously, and executes them in parallel while the arithmetic unit to be used is dynamically decided. Therefore, the sequence of instructions written in a program may differ

from the sequence of instructions actually executed (this kind of execution method is referred to as **out-of-order execution**).

On the other hand, **VLIW** consolidates multiple instructions that can be executed simultaneously into one instruction at the stage of generating the object program (program translated into machine language) by compiling (translating into machine language) the program, and statically allocates the arithmetic and logical units to be used. (VLIW means that multiple instructions (functions) are consolidated into one instruction, and thus the length of instruction) increases. In VLIW, instructions that can be executed simultaneously are not dynamically selected at the time of execution, so the sequence of instructions written in the object program is always same as the sequence of instructions actually executed. (This kind of execution is called **in-order execution**).

(4) Parallel processing

Pipeline/super-pipeline, superscalar, and VLIW aim to enhance processor performance by parallel execution of instructions in the processor. On the other hand, there is another approach of enhancing the performance of a computer by the parallelization of processors. The architecture of these parallel computers is classified into 4 types from the relation between flow of instructions and flow of data.

- **SISD (Single Instruction stream Single Data stream)**

This computer processes one unit of data with one instruction. General computers that do not have parallelized processors correspond to this architecture.

- **SIMD (Single Instruction stream Multiple Data stream)**

This computer processes multiple units of data with one instruction. **Vector computers** equipped with a **vector processor (array processor)** that simultaneously computes multiple data in array with one instruction correspond to this architecture.

- **MISD (Multiple Instruction stream Single Data stream)**

This computer processes a single unit of data with multiple instructions. Computers that use this architecture are not used in practice (although they are closer to pipeline control, strictly speaking, they are different).

- **MIMD (Multiple Instruction stream Multiple Data stream)**

This computer processes multiple units of data with multiple instructions. **Multiprocessors** where multiple processors are multiplexed correspond to this architecture.

4 Auxiliary Storage

Data to be processed inside the computer must be stored in main memory beforehand. However, the capacity of main memory is limited, and therefore it cannot store all data. Accordingly, data is stored in a separate storage device, and the required data is moved into main memory for processing. In such cases, auxiliary storage devices play the role of storing the data that cannot be stored in main memory. Therefore, auxiliary storage devices basically have larger capacity than main memory, and have a characteristic of non-volatility. In this section, we will learn about the types and mechanism of typical auxiliary storage devices.

4 - 1 Magnetic Disk

Magnetic disk is the most extensively used storage medium and is a typical example of an auxiliary storage medium. A magnetic disk is a disc-shaped storage medium with magnetic material coated on the surface, and it stores data by magnetic variation. Data is stored in concentric circles, with one concentric circle referred to as a **track**, and the unit of dividing the track in a constant length is called a **sector**. In modern magnetic disks, it is common to use the constant density recording method (zone sector method), where the number of sectors is increased in the tracks on the outer side.

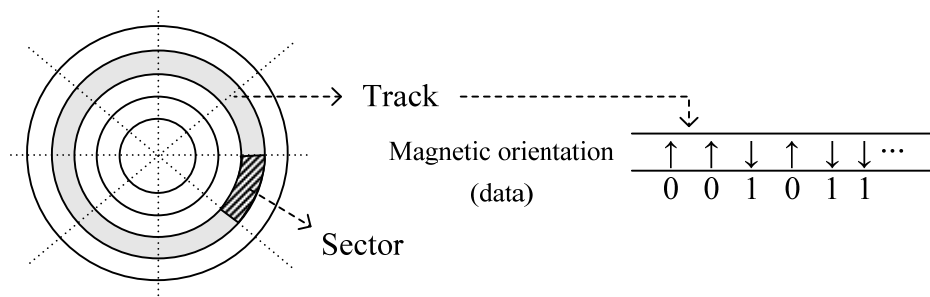


Figure 1-42 Magnetic disk

Magnetic head reads and changes the magnetic orientation of the magnetic disk. An electromagnet is embedded in the magnetic head, and it is fixed to the tip of an access arm to read and write data from and to the target track.

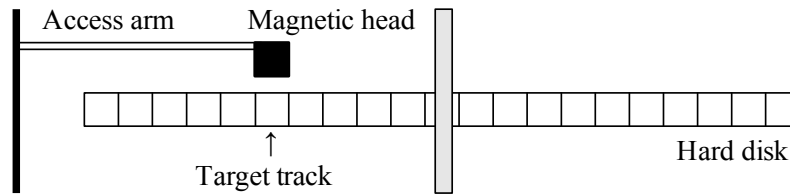


Figure 1-43 Magnetic disk and magnetic head

There exist the following types of magnetic disk-based auxiliary storage devices (auxiliary storage medium).

1) HDD (Hard Disk Drive)

This auxiliary storage device is mounted as a standard device in almost all computers. In addition to the built-in HDD provided inside the computer, an external HDD is also available that can be used by connecting it to the computer. By stacking multiple magnetic discs inside a firmly sealed case, it is possible to record a large amount of data. In order to read data from multiple magnetic discs that are stacked, multiple access arms having magnetic heads are mounted.

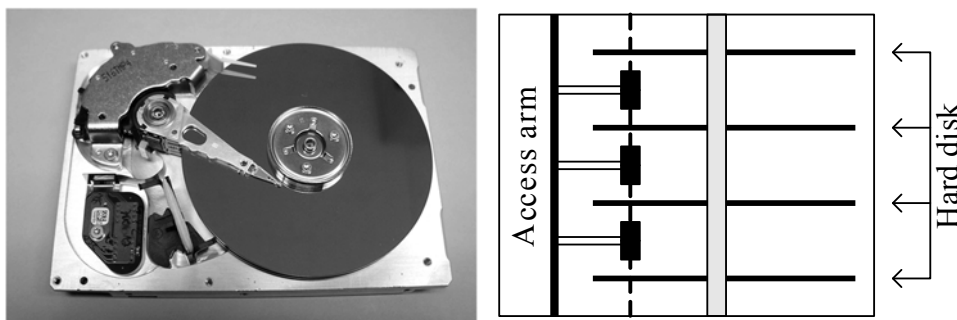


Figure 1-44 Structure of hard disk drive

Reading and writing of data from and to a hard disk drive is performed by moving the access arm horizontally and positioning the magnetic head on the target track of the hard disk. At that time, as evident from Figure 1-44, multiple tracks (tracks on a dashed line) can be read and written at the same position of the access arm (this set of tracks is called a **cylinder**). Reading and writing of data is performed for each cylinder separately.

In the hard disk drive, since the hard disk is continuously rotating at high speed, the reading and writing of information is performed without letting the magnetic head touch the hard disk (non-contact method). Therefore, it suffers from the disadvantage that it is sensitive to vibrations.

2) FD (Floppy Disk)

This is an auxiliary storage medium that can be carried by covering one magnetic disc (storage media that can be carried by releasing it from the device is called **removable**

media). Although it is very inexpensive, it suffers from the disadvantages that storage capacity is small and it is easily affected by magnetism. Therefore, it is hardly used these days. In the FDD (Floppy Disk Drive) used to read and write data on the floppy disk, a magnetic head comes in contact with the disk in order to read and write data (contact method). Therefore, the disk has a tendency to deteriorate the stored contents because of wear and abrasion.

3) ZIP

This auxiliary storage medium was developed by Iomega of the United States. Its storage capacity is larger than that of a floppy disk, and it is portable. However, it is hardly used these days.

4-1-1 Storage Capacity of Magnetic Disk Drive

Storage capacity is the amount of data that can be recorded in a storage device. The configuration of general magnetic disk drives is “magnetic disk drive > cylinder > track > sector.”

Therefore, the storage capacity of a magnetic disk drive can be determined in sequence from smaller to larger configuration units.

[Method of calculating storage capacity of magnetic disk drive]

- 1) From the amount of data that can be recorded in one sector and the number of sectors in one track, calculate the storage capacity of one track.
- 2) From the amount of data that can be recorded in one track and the number of tracks in one cylinder, calculate the storage capacity of one cylinder.
- 3) From the amount of data that can be recorded in one cylinder and the number of cylinders in the magnetic disk drive, calculate the storage capacity of the magnetic disk drive.

When the number of magnetic discs is small (e.g., FD), storage capacity is also calculated using the number of magnetic discs instead of cylinders.

Example: Calculate the storage capacity of a magnetic disk drive having the following specifications.

Number of cylinders/drive	300 cylinders
Number of tracks/cylinder	20 tracks
Number of sectors/track	30 sectors
Number of bytes/sector	500 bytes

- 1) Storage capacity of one track
 - = Storage capacity of one sector \times Number of sectors in one track
 - = 500 bytes/sector \times 30 sectors/track
 - = 15,000 bytes/track
- 2) Storage capacity of one cylinder
 - = Storage capacity of one track \times Number of tracks in one cylinder
 - = 15,000 bytes/track \times 20 tracks/cylinder
 - = 300,000 bytes/cylinder
- 3) Storage capacity of magnetic disk drive
 - = Storage capacity of one cylinder \times Number of cylinders in magnetic disk drive
 - = 300,000 bytes/cylinder \times 300 cylinders/drive
 - = 90,000,000 bytes/drive ... 90Mbytes/drive

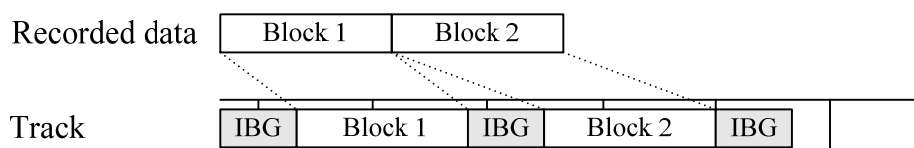
4-1-2 Recording Area in Magnetic Disk Drive

Recording area refers to the area required for recording data in the storage device. It is important how to handle the unit of data (recording method of data) for calculating the size of recording area required for recording data in the magnetic disk drive. In general magnetic disk drives, multiple **records**, which are units having logical meanings, are consolidated into a **block**, and a block is treated as the unit of input and output (normal records are also referred to as **logical records**, and blocks are also referred to as **physical records**). Consolidating multiple records into a block is referred to as **blocking**, and the number of records consolidated into a block is referred to as the **blocking factor**.

[Methods of recording data]

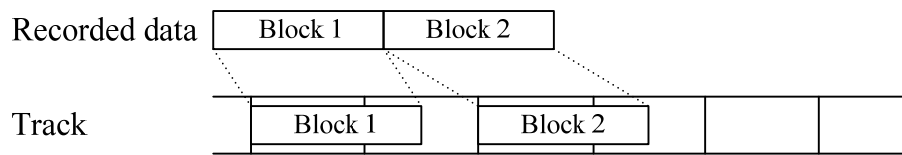
1) Variable method

This method reads and writes data in units of blocks. When data is recorded on a track, an **IBG (InterBlock Gap)** is inserted between two blocks so as to separate them.



2) Sector method

This method reads and writes data in units of sectors. Blocks are separated by not recording multiple blocks in the same sector. A concept called **cluster** is also used, where multiple sectors are consolidated and treated as the unit of input/output.



In the sector method, which is commonly used in modern PCs (hard disk drive), the size of the recording area required for recording data is calculated as follows:

[Method of calculating size of recording area of magnetic disk drive]

- 1) From the size of one block, calculate the number of sectors required for recording the data.
- 2) Calculate the number of blocks of all data to be recorded.
- 3) Calculate the number of sectors required for recording all data, and calculate the required number of tracks and the required number of cylinders in that sequence.

Example: For a magnetic disk drive that has the following specifications, calculate the number of cylinders required for recording 10,000 records with the blocking factor of 10 where one record is of 300 bytes. Data is recorded in the magnetic disk by using the sector method.

Number of cylinders/drive	300 cylinders
Number of tracks/cylinder	45 tracks
Number of sectors/track	30 sectors
Number of bytes/sector	512 bytes

- 1) Size of one block
 - = Size of one record \times blocking factor
 - = 300 bytes/record \times 10 records/block
 - = 3,000 bytes/block
- 2) Number of sectors required for recording one block
 - = Size of one block \div Storage capacity of one sector
 - = 3,000 bytes/block \div 512 bytes/sector
 - = 5.85 ... sectors/block
 - * Since 5 sectors do not suffice, it is rounded up to 6 sectors/block
- 3) Number of blocks to be recorded

= Total number of records ÷ blocking factor

= 10,000 records ÷ 10 records/block

= 1,000 blocks

4) Number of sectors required for recording all records

= Number of sectors required for recording one block

× Number of blocks to be recorded

= 6 sectors/block × 1,000 blocks

= 6,000 sectors

5) Number of tracks required for recording all records

= Number of sectors required for recording all records

÷ Number of sectors in one track

= 6,000 sectors ÷ 30 sectors/track

= 200 tracks

6) Number of cylinders required for recording all records

= Number of tracks required for recording all records

÷ Number of tracks in one cylinder

= 200 tracks ÷ 45 tracks/cylinder

= 4.44 ... cylinders

* Since 4 cylinders do not suffice, it is rounded up to 5 cylinders.

* In the calculations that are performed in steps 2) and 6), it is necessary to pay attention to the meaning of rounding-up the calculation results.

4-1-3 Average Access Time of Magnetic Disk Drive

Access refers to reading and writing data, and the time from placing an access request until the result is returned is called **access time** (in contrast, after an access request is received, the time taken until the drive is ready to accept the next access request is called **cycle time**).

Reading (or writing) of data in the magnetic disk drive is carried out in the sequence described below.

1) **Seek**

Move the magnetic head (access arm) up to the target track (cylinder). This time is called **seek time**.

2) **Search**

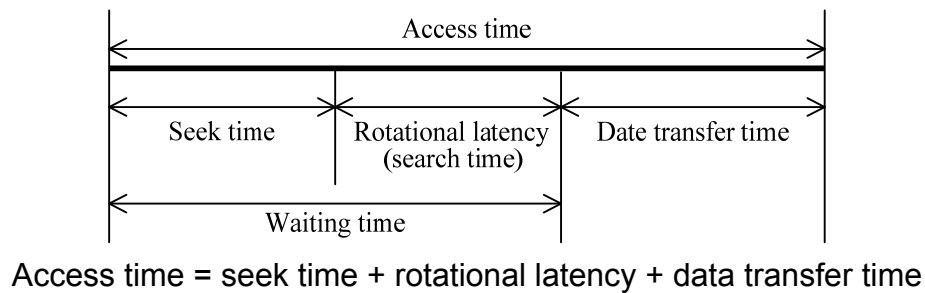
Wait until the head of the data to be read (or the position where data writing will start) comes right under the magnetic head. The time required for this process is called **rotational latency (search time)**.

3) Data transfer

The magnetic head will read (or write) the data. The time required for this process is called **data transfer time**.

Therefore, access time is a summation of these times.

[Access time of magnetic disk drive]



On the other hand, **average access time** is the average value of access time for reading (or writing) data of one unit of input/output. In this case, since the amount of data of one unit of input/output is the same, data transfer time is the same on each occasion. However, seek time and rotational latency are different on each occasion. Therefore, their respective average values (average seek time, average rotational latency) are used.

[Average access time of magnetic disk drive]

Average access time

$$= \text{average seek time} + \text{average rotational latency} + \text{data transfer time}$$

In order to calculate average access time, the values to be used as average seek time, average rotational latency, and data transfer time are shown below. Here, when average latency is used, it can be calculated as “average seek time + average rotational latency.”

1) Average seek time

This is the time required for the magnetic head to move up to the target track (cylinder). Therefore, seek time differs on the basis of both the position of the magnetic head when the access is started and the position of the target track. Usually, therefore, the average seek time stated in the specifications of magnetic disk is used.

2) Average rotational latency (average search time)

Since the magnetic disk rotates at a constant speed, in some cases there may not be any latency when the head of data is positioned directly underneath from the beginning (minimum rotational latency). On the other hand, it may be necessary to wait until the disk completes one rotation when the head of data has just passed (maximum rotational latency). Therefore, the average of minimum rotational latency and maximum rotation latency, which is 1/2 of the time required for the magnetic disk to complete one rotation, is used.

$$\text{Average rotational latency} = \frac{\text{Time required for magnetic disk to complete one rotation}}{2}$$

3) Data transfer time

Data is read (or written) when the target data passes underneath the magnetic head. In other words, when the magnetic disk completes one rotation, the track will complete one rotation underneath the magnetic head, and therefore data recorded in one track will be read (or written). Therefore, the **data transfer rate** of the magnetic disk can be calculated as follows:

$$\text{Data transfer rate} = \frac{\text{Storage capacity of one track}}{\text{Time required for magnetic disk to complete one rotation}}$$

Also, calculate data transfer time by dividing the amount of data (amount of transferred data) of one unit of input/output (data to be read or written) by the data transfer rate.

Average rotational latency and data transfer time are closely related to the rotational speed of the disk. Therefore, increasing the rotational speed of the disk shortens the time. On the other hand, the proportion of seek time, which is the physical time required for access arm movement, tends to increase in access time. Therefore, shortening the seek time by recording the data in consecutive area and thereby reducing the amount of access arm movement is useful for shortening the average access time. Moreover, when data is recorded in a piecemeal manner (**fragmentation**), the average access time will become longer. Therefore, it is necessary to perform **defragmentation** in order to eliminate fragmentation.

Example: Calculate the average access time for reading a data block of 2,000 bytes recorded in a magnetic disk drive having the following specifications.

Rotational speed	3,000 rotations/minute
Bytes/track	10,000 bytes
Average seek time	30 milliseconds

- 1) Time required for magnetic disk to complete one rotation
 $3,000 \text{ rotations} : 1 \text{ minute} = 1 \text{ rotation} : x \text{ milliseconds}$
 $3,000 \text{ rotations} : 1 \times 60 \times 10^3 \text{ milliseconds} = 1 \text{ rotation} : x \text{ milliseconds}$
 $3,000 \text{ rotations} \times x \text{ milliseconds} = 1 \times 60 \times 10^3 \text{ milliseconds} \times 1 \text{ rotation}$
 $x \text{ milliseconds} = 1 \times 60 \times 10^3 \text{ milliseconds} \times 1 \text{ rotation} \div 3,000 \text{ rotations}$
 $= 20 \text{ milliseconds}$
- 2) Average rotational latency
 $= \text{Time required for magnetic disk to complete one rotation} \div 2$
 $= 20 \text{ milliseconds} \div 2$
 $= 10 \text{ milliseconds}$
- 3) Data transfer rate
 $= \text{Storage capacity of one track}$
 $\div \text{Time required for magnetic disk to complete one rotation}$
 $= 10,000 \text{ bytes/track} \div 20 \text{ milliseconds/rotation (track)}$
 $= 500 \text{ bytes/millisecond}$
- 4) Data transfer time
 $= \text{Amount of transferred data} \div \text{Data transfer rate}$
 $= 2,000 \text{ bytes} \div 500 \text{ bytes/millisecond}$
 $= 4 \text{ milliseconds}$
- 5) Average access time
 $= \text{Average seek time} + \text{Average rotational latency} + \text{Data transfer time}$
 $= 30 \text{ milliseconds} + 10 \text{ milliseconds} + 4 \text{ milliseconds}$
 $= 44 \text{ milliseconds}$

Finally, note that the average access time calculated here is simply a value for a magnetic disk drive as an independent device. For example, when **disk cache** is used, the average access time of the magnetic disk drive (auxiliary storage device) as seen from the processor (main memory unit) will not be the average access time calculated here. In such cases, with consideration of the **hit ratio** (or **NFP**), it is necessary to determine average access time (**effective access time**) as the weighted average of access time of disk cache and the access time of the magnetic disk drive.

4 - 2 Optical Disc

Optical disc is a commonly used auxiliary storage medium (**removable media**) these days. While optical disc drives use different mechanisms depending on the device, all of them are high-capacity auxiliary storage devices where the reading and writing of data is performed by means of a laser beam. Optical disc drives can be of the **slot-in type** where a disc is inserted

into the slot as it is with an audio device, or of a **tray type** where a tray for mounting the disc comes out of the drive.

Although the structure of optical discs is basically the same as that of magnetic disks, the main difference is that tracks where data is recorded are spiral-shaped. Data is read by an optical head that detects the difference of reflected light after projecting a laser beam. However, seek time is longer because the optical head is heavier than a magnetic head. As a result, average access time of optical disc drives tends to be longer than that of magnetic disk drives.

Optical discs include **CD (Compact Disc)**, with standard storage capacity of about 700Mbytes, and **DVD (Digital Versatile Disc)**, which has large storage capacity (standard storage capacity: a single-sided, single-layer disc holding 4.7Gbytes, and a single-sided, double-layer disc holding 8.5Gbytes) by multiple layers and short wavelength of laser beams.

(1) Read-only

Read-only optical discs record information by creating a small hole called a pit on the surface of discs in the factory and by changing the reflection of the laser beam. They are inexpensive because bulk production is possible by pressing a master disc. However, the user cannot write new data on these discs. Typical examples of read-only optical discs are **CD-ROM (CD-Read Only Memory)** and **DVD-ROM**.

CD-ROM was originally developed for music. Therefore, music data and digital data can be mixed on the same disc. Moreover, when CD-ROM or DVD-ROM is produced in bulk, aluminum vapor-deposited layers (reflection layers) that are sandwiched between acrylic are pressed after adhesive agent is mounted with. When this adhesive agent peels off, resin (aluminum deposition) covering the surface will deteriorate (oxidize), and data can no longer be read. Therefore, the life span of CD-ROM and DVD-ROM is said to be about 20-30 years.

(2) Write-once

In **write-once** optical discs, while data written at the factory cannot be deleted or modified, data can be added (recorded) by creating a section where data can be recorded. Data can be written by making a pit, by burning an organic coloring matter that is coated on the disc with a strong laser beam. However, the burned organic coloring matter cannot be restored to its original state, and therefore, it is not possible to rewrite even the added data (WO (Write Once)). Typical examples are **CD-R (CD-Recordable)**, **DVD-R**, and **DVD+R**.

(3) Rewritable

In **rewritable** optical discs, data can be rewritten. As for the recording mechanism, heat and light are used to control the temperature of the recording surface of the disc, and data is recorded with metal phase-change technique that creates two states, namely, a crystalline state and a non-crystalline state. Since data can be added/deleted/modified, this is an auxiliary storage medium that can be handled in a similar way to a magnetic disk. Typical examples are **CD-RW (CD-ReWritable)**, **DVD-RW**, **DVD-RAM (DVD-Random Access Memory)**, and **DVD+RW**. Write-once and rewritable optical discs can use the same playback and recording drives (**CD-R/RW drive**, **DVD-R/RW drive**). However, because the laser beam reflection rate differs depending on different data recording methods, data may sometimes not be read correctly in read-only playback drives (**CD-ROM drive**, **DVD-ROM drive**).

[Methods of writing data in optical discs]

- **DAO (Disk-At-Once)**

In this method, all data is written in one go on the entire disc.

- **TAO (Track At Once)**

In this method, data is written separately for each track.

- **SAO (Session At Once)**

In this method, data is written separately for each session (a session is a recording unit composed of lead-in that indicates the start, data, and lead-out that indicates the end).

- **Multi-session**

This method allows multiple sessions to be recorded on one disc. This recording method can be performed with a device that supports TAO or SAO.

- **Packet writing**

Just like a magnetic disk, this method writes data in small units (packets). This method is expanding the range of applications of optical discs.

(4) Blu-ray disc

Blu-ray disc has drawn a lot of attention in recent years. It uses a blue-violet laser beam which has a much shorter wavelength than the laser beams that is used in conventional optical discs, which achieves high capacity. Most **Blu-ray drives** that is used for playback and recording of Blu-ray discs also support CD, DVD, etc. At present, this format is mainly used for entertainment purposes. However, people have started using Blu-ray discs as a form of auxiliary storage media for PCs.

4 - 3 Semiconductor Memory

Semiconductor memory (IC memory) is a storage device that uses IC such as RAM, ROM, and flash memory. In addition to being used as storage devices that constitute main memory and registers, it is also used as the recording medium of auxiliary storage devices.

(1) SSD (Solid State Drive)

SSD (Solid State Drive) is a flash memory-based auxiliary storage device that is expected to replace the HDD (Hard Disk Drive). In SSD, information is electrically read and written using flash memory. Therefore, unlike a hard disk drive, it does not require any drives (e.g., disk, access arm). For that reason, it has shorter access time because of no physical location seeking or rotational latency. Moreover, it has faster direct access (random access) that reads and writes data irrespective of the recording sequence of data. Additionally, it can be compact and lightweight, consumes less power, and is also resilient to vibrations or impact. However, there are limitations to the number of times data can be written (durability), and it suffers from the disadvantage that it is not suitable for applications where data needs to be rewritten frequently.

(2) RAM disk

RAM disk is an auxiliary storage device that uses a part of main memory (DRAM) as external storage by means of virtualization. In the early days, external RAM disks were used for expansion by combining a large amount of inexpensive RAM. However, it is rarely used these days. While high speed access is possible, data is cleared (volatility) once the power is off. Therefore, it is not suitable for storing data. (In order to save data, it is necessary to write it out to a hard disk.) In addition, data (file) that is recorded on a RAM disk is often called a **RAM file**, and the mechanism of data management is sometimes called a **RAM file system**. (A file which has the “.ram” extension and is used in a certain software product is also known as a RAM file. However, it needs to be noted that this is different from the RAM file mentioned above).

(3) USB memory/SD card (SD memory card)

USB memory is an auxiliary storage device that can easily be connected to and removed from the computer by equipping flash memory with a USB connector. On the other hand, **SD card (SD memory card)** is an auxiliary storage medium where flash memory is put on a chip, and it is used by insertion into the SD slot of computers or other devices such as digital

cameras, smartphones, and cell phones. Both USB memory and SD card are flash memory based removable media, and they are suitable for carrying data. There are unified standards of the USB plugs and receptacles for USB memory. However, note that SD cards and SD slots may be using different standards depending on the manufacturer.

4 - 4 Other Auxiliary Storage Media and Drives

(1) MO (Magneto-Optical disc)

MO (Magneto-Optical disc) is an auxiliary storage medium that magnetizes the magnetic material that is coated on the recording surface of disc and records data by giving the direction of the magnetic force. By increasing the temperature of the magnetic material with laser beams and eliminating magnetic force for clearing data, it is possible to record data by magnetizing it once again. In comparison with magnetic storage media, it has the excellent properties of conservation and environmental resistance, and it is suitable for long-term storage of data, and even high density recording is possible. (There are standards such as a storage capacity of 230Mbytes or 640Mbytes). However, a laser beam is used for reading data, so data can be read with one rotation of disc. But, both a laser beam and magnetic head are used for writing data, so the disk needs to rotate one time each for clearing and writing data (total of two rotations). Therefore, access speed becomes slow. Moreover, the standard of 640Mbytes is a higher-level standard than that of 230Mbytes. Therefore, while the drive for 640Mbytes can read and write a 230-Mbyte disc, the logic format (format of recording data) is not unified, so it may not be possible to exchange data between different models. For these reasons, it is hardly used these days.

(2) Magnetic tape unit

Magnetic tape unit is an auxiliary storage device that records data by magnetizing the tape (**magnetic tape**) where magnetic material is coated. The access speed of a magnetic tape unit is extremely slow, and only sequential access is possible where data is consecutively read and written in sequence. (A device that consecutively records like a magnetic tape unit is called a **streamer**.) However, cost per unit information is extremely inexpensive in the case of magnetic tapes, and this auxiliary storage medium can record a large amount of data. Therefore, it is used for backing-up the hard disk. In the past, people mostly used open reel-type magnetic tapes, which were not suitable for hand-carrying because of their large size and weight. These days, however, people mostly use cartridge-type portable **DDS (Digital Data Storage)** on the basis of **DAT (Digital Audio Tape)** technology that was developed for music.

(3) PC card

PC card is a card-type auxiliary storage medium used in notebook PCs and other devices, and its standard was defined by **PCMCIA (Personal Computer Memory Card International Association)**, an industrial body of the United States. PC cards for several applications have been available on the market, such as **memory cards** used for capacity expansion (increasing memory) of main memory, FAX modems, and adapters for LAN connection. But currently USB memory and SD cards prevail, and PC cards are rarely used.

5 Input/Output Unit

In order to use a computer, it is necessary to enter data in the computer, and send out the result of processing this data inside the computer. Here, a device that is used for entering data is called an input unit, and a device that is used for output is called an output unit.

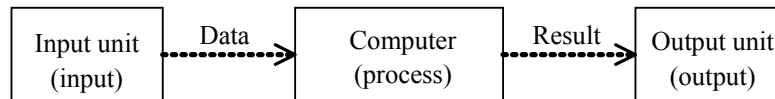


Figure 1-45 Flow of data processing

Various input units and output units are available. Each unit has appropriate functions that are suitable for input data or output format. This section describes the types, characteristics, and mechanism of typical input/output units.

5 - 1 Input Unit

Input unit is a device used for entering data in a computer. Input units include devices that enter characters or numbers by using signals, devices that read data written in a paper, and devices for entering position information (**pointing devices**).

5-1-1 Devices for Entering Characters and Numbers by Using Signals

(1) Keyboard

Keyboard is the most general purpose input device, and it is always supplied as a standard input device of PCs. When a key on the keyboard is pressed, the corresponding character or number is entered as a signal (code).

When an old keyboard is used, one depression of a key may work as if the same key were pressed several times.

This is sometimes referred to as **chattering**. However, chattering is originally a phenomenon in which multiple ON/OFF signals occur within a short period of time (a few milliseconds after pressing a push button), in response to pressing the button with a mechanical connection once. Therefore, strictly speaking, the above-mentioned behavior is not the same as chattering.



Keyboard

5-1-2 Device for Reading Data Written on Paper

(1) OCR (Optical Character Reader)

OCR (Optical Character Reader) optically reads the handwritten characters or printed characters on the basis of the strength of the reflected light, and enters them as data. OCRs developed in the early days could only read characters that were written in special OCR fonts. However, modern OCRs can sufficiently read even handwritten characters. Therefore, it is also used for reading a zip code (or postal code) that is written on a letter or a postcard. However, there is a risk of the reading mistake (misreading) of characters, and therefore, it is necessary to thoroughly check the entered data.



OCR

(2) OMR (Optical Mark Reader)

OMR (Optical Mark Reader) is a device that optically reads the portions that are marked with black color on a mark sheet by using the reflected light, and enters them as data. Instead of reading characters or numbers, it reads the marked position information. This input device converts this position information into the corresponding data for processing. Therefore, it cannot handle stains or such spots well, but there are fewer reading mistakes. It is used for reading (grading) answer sheets of exams where there are many examination candidates.



Example of mark sheet

(3) MICR (Magnetic Ink Character Reader)

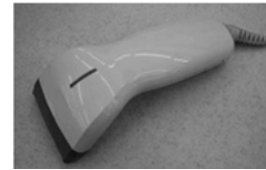
MICR (Magnetic Ink Character Reader) is a device for entering characters just like OCR; however, it can only enter characters of a special font written with a special ink mixed with magnetic material. It is not affected by stains because it reads characters that are printed with magnetized ink. Since characters cannot be created without a special device, it is used as a reading device for notes and checks.



Application of MICR

(4) Barcode reader

Barcode reader is a device that optically reads a **bar code** that represents data with bars (lines) of different thickness and different spacing. Its typical use is as the input device of a POS system, which is used in the cash registers of retail stores.



Barcode reader

[Main bar codes]

- **One-dimensional bar code**

This is a horizontal bar code where vertical lines of different thickness are placed in a band. When someone simply says “bar code,” this typically refers to the one-dimensional bar code. It is used extensively because it can be directly printed on packing material or containers, and printing size can be freely enlarged or reduced. In addition, it comparatively costs less, which is an added advantage.

Typical examples include **JAN code (Japanese Article Number code)**, **ITF code (Interleaved Two of Five code)**, and **Code128** (bar code that handles ASCII characters).

- **Two-dimensional bar code (QR code)**

This is a matrix-based bar code where information is represented with small squares within a larger square. It has three symbols for detection called position detection patterns, and can recognize rotation angle and reading direction. Therefore, information can be read from any direction. Moreover, it can be read with smartphones or cell phones without using any special reading device.



One-dimensional bar code



Two-dimensional bar code

(5) Image scanner

Image scanner is an input device that optically reads and enters images such as drawings and photographs that are written or printed on paper by using the same principle as facsimile. The image scanner enters drawings or photographs that are written or printed on paper as a dot image, which is a set of points. An image scanner where paper is fixed and reading device is moved for reading the image data is also called an **image sensor**.

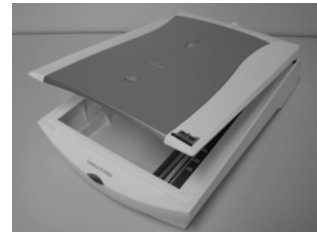


Image scanner

5-1-3 Pointing Device

(1) Mouse

Mouse is a device that feeds the amount of rotation and direction by rotating the ball that is located at the bottom. It got this name because its shape resembles a mouse. Pointing devices that have similar functions include the **trackball** or the **acupointer** where the ball is rotated by turning the bottom side up.



Mouse

(2) Joystick

Joystick is a device where a stick is moved back and forth and around, and position information is fed on the basis of the direction and its angle. It is used in operating game software.

(3) Touch screen (touch panel)

Touch screen (touch panel) is a device that feeds position information by touching the display screen with a finger.

[Input method of touch screen]

- **Infrared shielding method** (Infrared method)
It detects the position from the changes in infrared ray reflection that occurs when an infrared ray beam is shielded.
- **Capacitance method**
It detects the position from the changes in electrical charge on the surface which is caused by the static electricity in the portion that is touched.
- **Analog resistive film method** (Resistive film method)
It detects the position from the changes in the resistance values of the resistive film in the portion that is touched.
- **Matrix switching method**
It detects the position where the electrode switches that are arranged on the matrix are pressed.

(4) Light pen

Light pen is an input device that feeds position information by directly pointing and tracing on a display screen. The tip of the light pen has an optical sensor, and the position on the display can be detected from the movement of this sensor.

5-1-4 Other Input Devices

(1) Digitizer/tablet

Digitizer is an input device that feeds two-dimensional and three-dimensional drawing information as coordinate information by tracing a drawing on the panel. A compact size digitizer that can be used on a desk is specifically referred to as a **tablet**.

It is used for entering drawings in a design support system called CAD and creating data of an NC (Numeric Control) machine tool.



Digitizer

(2) Card reader

Card reader is used for reading information that is recorded in a card. Typical card readers include a **magnetic card reader** that is used for reading magnetically recorded information

in a bank cash card or a credit card, and an **IC card reader** that is used for reading information in an IC card (a card with an embedded IC chip). The IC card is used as an entrance key card or a card key.

(3) Biometric authentication device

Biometric authentication device is a device that feeds information for biometric authentication using physical and behavioral features of human beings. There are devices that feed (authenticate) information of the intended staff members who are targeted for authentication by an authentication method, such as a face authentication device, a palm authentication device, a fingerprint authentication device, and an iris authentication device.

(4) Sound input device

Sound input device is a device that identifies sounds (analog signals) of human beings and feeds data after these analog signals are converted into digital signals by using electronic circuits (**A/D converter**). It is also used as a biometric authentication device for authenticating voice pattern or voice print.

(5) Digital camera

Digital camera is a device that feeds the images taken by it to a computer. Instead of the film that is used in normal cameras, it uses **CCD (Charge Coupled Device)** for the conversion of graphic images into digital data. In most cases, **smart media**, which is a type of flash memory, is used as the recording medium of images. Moreover, a **digital video camera** is used for recording video.

5 - 2 Output Unit

Output unit is a device that is used for exporting the processing results outside computers. Output units include displays that show the results and printers that print the results.

5-2-1 Display

Display is a device that displays images in color by combining three primary colors of light, namely **RGB (Red, Green, Blue)**. Screen is an aggregation of points (**dots**), and the higher the density of dots, the higher the display quality of images. This is also called the **resolution** of the screen, and **dpi (dots per inch)** is used as a unit of resolution.



[Screen resolution standards of display]

Standard name	Screen size
VGA (Video Graphics Array)	640×480
SVGA (Super VGA)	800×600
XGA (eXtended Graphics Array)	$1,024 \times 768$
SXGA (Super XGA)	$1,280 \times 1,024$

Displays of present-day PCs mostly use a **multiscan method** that can support various resolutions. However, for supporting high resolution, graphic memory (**VRAM (Video RAM)**) that supports high resolution is required. Storage capacity required for VRAM is decided by the screen resolution and the number of colors per one bit (the number of bits that is required for displaying a color).

Example: On a display that has the resolution of $1,280 \times 1,024$, calculate the storage capacity of VRAM that is required for displaying about 16,000,000 colors by assigning 256 grades of intensity to each color of RGB.

- 1) Number of bits required for displaying one color
 $256 = 2^8$, therefore 8 bits/color
- 2) Number of bits required for displaying color information of 1 dot
 $= 8 \text{ bits/color} \times 3 \text{ colors/dot}$
 $= 24 \text{ bits/dot}$
- 3) Number of bits required for one screen
 $= \text{Number of bits required for one dot} \times \text{Number of dots on one screen}$
 $= 24 \text{ bits/dot} \times (1,280 \times 1,024) \text{ dots/screen}$
 $= 31,457,280 \text{ bits/screen} \cdots \text{About 4 Mbytes/screen}$

In addition, for increasing the display speed and the precision of images, a **graphic accelerator** (a type of video chip) is required. In the raster scan display, where images are scanned line by line, there is an output method called **interlaced mode**. This method extracts every other scanning line and displays one image over two rounds, and thereby, increases the display speed. However, flickering can easily occur on the screen with this method, so in most of the cases, **non-interlaced mode** is used where scanning lines are displayed in sequence from the top.

(1) CRT display

CRT display is an output device that uses a cathode-ray tube just like television. Light is emitted by applying electron beams onto a phosphor screen, and any character or image can be drawn depending on the beam strength. While it has advantages in that the screen is easy to see and display speed is also fast, it suffers from disadvantages in that the device is large (because of depth) and power consumption is also high. Furthermore, there is another problem of burning the image on the display when the same image is displayed for a long time. In order to prevent this burning, software called a **screen saver** is used.

(2) LCD (Liquid Crystal Display)

LCD (Liquid Crystal Display) is an output device that uses liquid crystals whose permeation rate of light varies depending on voltage. As compared with CRT displays, it is thin and lightweight, and consumes less power. Therefore, it is mostly used in a notebook PC; however, these days it is also used in a desktop PC. Since liquid crystals themselves do not emit any light, either a reflection board is embedded or a back-light method is used where light is thrown from the rear. As compared with an **STN liquid crystal display**, which controls multiple pixels with one semiconductor, the response speed of a **TFT liquid crystal display**, which controls one pixel with one transistor, is faster and its viewing angle is also broad. Therefore, TFT LCDs are more widely used these days.

(3) PDP (Plasma Display Panel)

PDP (Plasma Display) is an output device where gas is sealed between two panels of glass and light is emitted by applying voltage. While it has higher luminance and wider viewing angle in comparison with other methods, power consumption is also large because high voltage is required.

(4) OLED (Organic Light Emitting Diode) display

OLED (Organic Light Emitting Diode) display is an output device that uses organic compounds that emit light when voltage is applied. Unlike a liquid crystal display, it emits light on its own on a glass substrate. Its response speed is fast, image quality is of high luminance because of high contrast, and the viewing angle is also wide. In addition, power consumption is small, so it has potential to become the next-generation display. However, its life is short.

5-2-2 Printer

Printer is a device for color printing by using three primary colors known as **CMY** (Cyan [bright light blue color], Magenta [bright red-purple color], and Yellow). However, proper black color cannot be formed even if these three colors are mixed. Therefore, in most cases, four colors of CMYK are used where black is added as the key.

Most printers print characters and drawings as an aggregation of points (**dots**). Therefore, the higher the number of dots per unit area, the higher the printing quality. This is referred to as the **resolution** of a printer, and it is expressed in the same unit as for displays, namely, **dpi** (**dots per inch**).

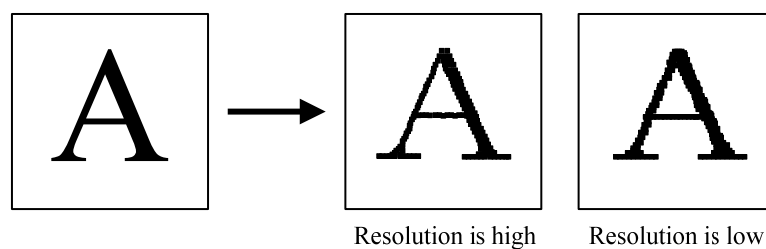


Figure 1-46 Characters are an aggregation of dots

(1) Impact printer

Impact printer is a device that prints by mechanically striking ink ribbon on paper with a print head. Although it generates noise, it is possible to simultaneously print multiple copies by placing carbon paper between multiple sheets of paper.

1) **Dot impact printer**

This is a **serial printer** that prints each character separately. From the back of the ink ribbon, it strikes the print head that has the pins each of which corresponds to a dot and these pins are arranged in the shape of characters. Kanji characters and graphic symbols can also be printed if they can be represented as an aggregation of dots.

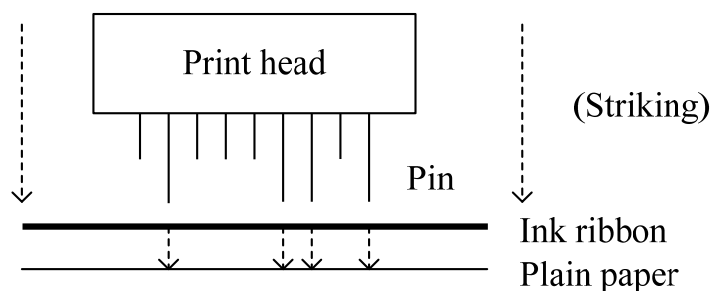


Figure 1-47 Dot impact printer

2) Line printer

This printer prints all characters in one line at a time. It uses either a print drum method, where all characters in one line are printed by striking ink ribbon on the paper while the print drum rotates, or a dot method where the number of print heads is equal to the number of characters.

(2) Non-impact printer

Non-impact printer is a device that prints by transferring ink or toner onto paper by using heat, etc. While it does not produce much noise and it can print various graphic symbols, it cannot handle simultaneous printing of multiple copies by using carbon paper.

1) Thermal printer

This is a general term for printers that use heat for printing.

- **Thermal printer**

This printer uses heat and special heat-sensitive paper.

- **Thermal wax transfer printer** (thermal transfer printer)

This printer transfers waxed color ink that is applied on the ink ribbon of four colors (CMYK) by melting it with heat.

- **Dye sublimation thermal transfer printer**

This printer applies heat to four-color (CMYK) sublimation dyes mounted on a film in order to convert the dyes directly to the gaseous state, which is absorbed by special paper.

2) Inkjet printer

This is a **serial printer** that prints each character separately, in units of dots by blowing ink in the shape of characters from the print head. These days, the **photo printer** is commonly used where photo quality (image quality like photographs) is produced by using four or more color inks.



Inkjet printer

3) Laser printer

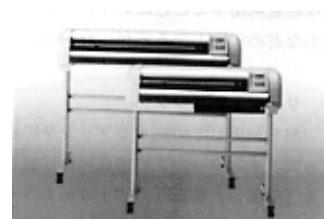
This is a **page printer** that prints in units of pages by affixing toner (powder ink) onto a photoconductive drum by using a laser, and then transferring the toner onto the paper. Since it uses the same principle as a copier, it is also referred to as an electrophotographic printer. A printer or a printer driver creates a print image in the bitmap format. Therefore, the size of characters and spacing between lines can be freely adjusted and even images

can be printed. (The printer control code for creating page images is called **page description language**, and **PostScript** or **ESC/Page** is available). As the performance indicator of a laser printer, in addition to resolution dpi (number of dots per one inch), the number of pages that can be printed in one minute (**ppm (pages per minute)**) is used.

5-2-3 Other Output Devices

(1) Plotter

Plotter is a device that prints graphics and is used for printing drawings that are created with CAD, etc. There is the **flatbed plotter**, which moves the pen in *X* axis and *Y* axis by fixing the paper (this is referred to as an **XY plotter**), and there is the **drum plotter**, which moves the pen only in the *Y* axis direction while the paper wrapped on the drum is moved in the *X* axis direction.



Plotter

(2) Projector

Projector is an output device that projects the data that is inside the computer. It is generally used for enlarged projection of display images on a large screen. The brightness of the light source of the projector is expressed in units of **lumen (lm)**.



Projector

(3) Electronic paper

Electronic paper is a very thin (about 0.1mm) display device that is made of soft material and has the characteristics of paper. Power consumption is very small, and data can be repeatedly reproduced and erased. This is one of the applied technologies of **MEMS (Micro-Electro-Mechanical Systems)**, which is a nanometer level micro-electro-mechanical device and manufacturing technology.

(4) Voice synthesizer

Voice synthesizer converts data that is inside the computer (digital data) into analog signals by using an electronic circuit (**D/A converter**), and artificially creates and generate a human voice or other sounds. It is used in screen readers (screen reading software) that read out the

information on the display, and is also used for making announcements in public institutions, transport facility, and event venues. Moreover, **vocaloid**, which sings a song by using the voice of a sampled person and by feeding melody and lyrics, is also one of the methods of using a voice synthesizer in a broad sense.

5 - 3 Other Input/Output Units

If an input unit is defined as a device that is used for entering data into a computer, and an output unit is defined as a device that is used for exporting the results of internal processing, in addition to general input/output units, the following devices are also classified as input/output units.

(1) Communication control unit

Communication control unit is a device that is used for various controls when the computer is connected to a network. It performs serial-parallel conversion of data (serialization and deserialization of characters), error control on data, etc. Since it receives data from the network and transmits data to the network, it is classified as a type of input/output device.

[LAN interface card]

This is a card-type interface device used for connecting a computer (mainly a PC) to a LAN (Local Area Network) which is a private communication network that is laid in a limited area). **Wired LAN interface card**, which is connected by using cable, and **wireless LAN interface card**, which is connected with wireless means such as electromagnetic waves, are available.

(2) Drive unit

Drive unit is a device that is used for operating other machinery or machinery. It converts the directions that are given by a computer into a mechanical operation (e.g., an operation of the robot arm that is used in a factory) and transmits it to the outside. Therefore, it can be called a type of output device. **Sensors** (devices that convert the measured values into electric signals) and **switches** that are used in machines with built-in drive units are also input devices, in a broad sense.

- **Chattering**

Chattering is a phenomenon where in response to pressing the push button switch of a mechanical contact once, a series of “on” and “off” signals is generated during a few milliseconds of pressing the switch. As a measure for preventing false operation, there is a method of waiting for the “on” and “off” signals to become stable.

(3) Imaging device

Imaging device is a device that feeds images. Digital cameras and digital video cameras are also classified as imaging devices. There are also electron microscopes and radio telescopes that import and analyze input images as digital data inside the computer.

5 - 4 Input/Output Control Methods

Input/output control methods are methods that control exchange of data between the input/output units and the main memory unit. When a program under execution is issued an input/output instruction, an **SVC interrupt** occurs to execute the interrupt program of the input/output control (the input/output control program). After that, when the input/output operation is completed, an **input/output interrupt** occurs to indicate the completion of input/output.

(1) Program control (direct control)

Program control (direct control) interprets the input/output control program with the arithmetic and logical unit of the CPU and gives directions to input/output units, and in this manner, data is exchanged between the input/output units and the main memory unit via the CPU. While this is the most basic control method, there is a large difference in the operating speed between CPU and input/output units, and therefore, it reduces the efficiency of utilization of the CPU.

(2) DMA (Direct Memory Access) method

In the **DMA method (Direct Memory Access method)**, data is directly exchanged between the input/output units and the main memory unit without going through the CPU, because input/output control is performed by a special device called a DMAC (DMA Controller). When a DMA request (input/output request) is given to DMAC by the CPU, DMAC independently controls input/output, and in the meantime the CPU can execute another process, which improves its efficiency of utilization. It is widely used, mainly as the input/output control of PCs.

(3) Channel control

Channel control uses a dedicated processing unit (**input/output channel** or **channel**) for input/output. When the **channel program** is sent to an input/output channel from the CPU, the input/output channel independently fetches the required information and decodes the instructions. In this manner, the input/output channel controls the input/output independently of the CPU. It is mainly used as the input/output control of general-purpose computers.

[Classification of input/output channels]

- Selector channel: ... This exclusively controls one input/output unit at a time.
- Multiplexer channel: ... This simultaneously controls multiple input/output units.
 - Byte multiplexer channel:
This controls low-speed data transfer in units of bytes.
 - Block multiplexer channel:
This controls high-speed data transfer in units of blocks.

5 - 5 Input/Output Interfaces

Input/output interfaces are rules (standards) for connecting peripheral devices, such as input devices and output devices, to computers. Also, software that controls the peripheral devices connected to input/output interfaces is called a **device driver**.

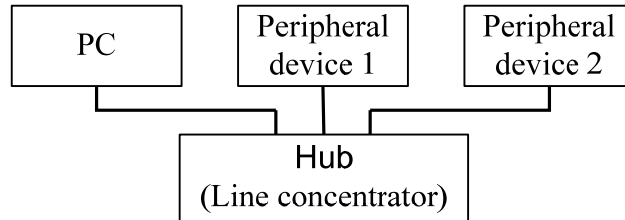
5-5-1 Classification of Input/Output Interfaces

Input/output interfaces can be classified into **serial interface** (serial data transfer method in units of bits) and **parallel interface** (parallel data transfer method in units of bytes or words). Also, since digital signals are used inside the computer, in order to connect the peripheral devices that support analog signals, it is necessary to perform conversion between analog signals and digital signals by using **analog input/output interfaces** (**analog input/output board**).

[Connection mode (topology) for connecting peripheral devices]

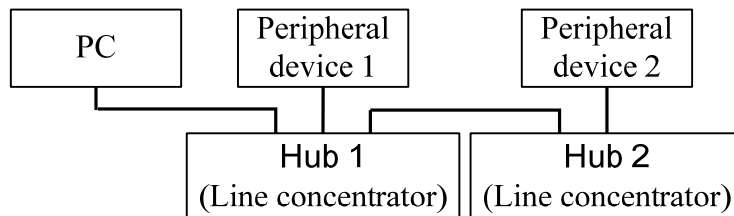
- **Star connection**

In this method, multiple devices are connected via a **hub** (line concentrator).



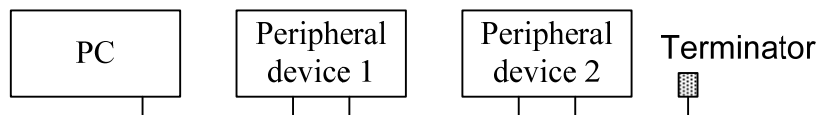
- **Cascade connection**

In this method, multiple devices are connected by having a multilevel connection of hubs (line concentrator). Tree-shaped connection modes like star connection and cascade connection are sometimes collectively referred to as a **tree connection**.



- **Daisy chain connection**

In this method, multiple devices are connected like a daisy chain (bunching) method from a device connected to the PC. Generally, at the other end of a cable, a terminating device (termination resistance) called a **terminator** is connected.



5-5-2 Types of Input/Output Interfaces and their Characteristics

(1) RS-232C (Recommendation Standard-232C)

RS-232C is a serial interface used for connecting a computer to communication devices (data circuit-terminating equipment) like modems. It is standardized by **EIA (Electronic Industries Association)**, and it allows two-way transmission of data. These days it is also used for connecting to input/output devices such as a mouse.

- **Interface board**

This is a board that allows the number of ports (slots) to be increased by mounting on the expansion slot of a PC. (The analog input/output board is also an interface board.) There are some products that support serial interface board, parallel interface board, and data transfer method (input/output interface). For example, in order to increase RS-232C interface ports, you can mount a serial interface board or an RS-232C interface board.

(2) SCSI (Small Computer Systems Interface)

SCSI is a parallel interface used for connecting computers like PCs to peripheral devices. It is standardized by **ANSI (American National Standards Institute)**, and it allows two-way transmission of data. It is mainly used for high-speed data transfer in hard disk drives and CD-ROM drives. With daisy chain connection from a device that is connected to the PC, SCSI can connect up to seven peripheral devices. A SCSI ID (identification number: 0 to 7) is assigned in order to identify the connected devices. In this case, the connection sequence does not matter as long as the ID is unique; that is, it need not be in the sequence of 0 to 7.

[Advanced SCSI standards]

- SCSI-2
 - Fast SCSI: Transfer speed of 10 Mbps with 8-bit bus
 - Wide SCSI: Transfer speed of 20 Mbps with 16-bit bus
- SCSI-3
 - Ultra SCSI: Transfer speed of 20 Mbps with 8-bit bus
 - Ultra Wide SCSI: Transfer speed of 40 Mbps with 16-bit bus

In SCSI-3, serial interface technique based on the serial SCSI is standardized, which allows high-speed data transfer at 50 to 200 Mbps.

(3) USB (Universal Serial Bus)

USB is a serial interface standardized in 1996, and it can connect various peripheral devices such as a keyboard, a mouse, and a modem. By using a USB hub, up to 127 devices can be connected in the tree configuration. Multiple standards for the shape of connectors are defined according to the device to be used.

[Data transfer modes of USB]

- **USB1.1**

It is available in low-speed mode (1.5 Mbps) that is used for connecting low-speed peripheral devices such as a keyboard and a mouse, and full-speed mode (12 Mbps) used for connecting somewhat high-speed peripheral devices such as a printer and a scanner.

- **USB2.0**

There are three types of data transfer modes. A mode is added to two data transfer modes of USB1.1. This third mode is called high-speed mode (480 Mbps) and is used for connecting high-speed peripheral devices such as external hard disks.

- **USB3.0**

In addition to the three types of data transfer modes of USB 2.0, SuperSpeed mode (5,120Mbps) is added. Also power management is improved in the standard.

(4) IEEE 1394

IEEE 1394 is a serial interface that allows high-speed transfer of data and is used for connecting multimedia-related devices such as digital cameras. It is available in three types of data transfer modes (100 Mbps, 200 Mbps, and 400 Mbps), and its main characteristic is that devices can be connected even without going through a PC. This standard is owned by **IEEE (Institute of Electrical and Electronics Engineers)**, and Apple's FireWire and Sony's i.Link are trade names of IEEE 1394.

[Common functions of USB and IEEE 1394]

- **Bus power method** (Bus powered)

In this method, power is supplied from a PC via connection cable. The amount of power that is supplied by this method is limited. Therefore, operation of devices that have large power consumption cannot be ensured.

- **Hot plug**

In this method, a cable (device) can be connected and disconnected while the power supply of the computer is on.

(5) Centronics

Centronics is a parallel interface used for connecting with printers, and it is standardized as IEEE1284 by **IEEE (Institute of Electrical and Electronics Engineers)** including up to the extended specifications. There are standards such as D-Sub 25 pins (24 pins) and 36 pins.

(6) GPIB (General Purpose Interface Bus)

GPIB is a parallel interface that is used for connecting with measuring instruments, and it is standardized as IEEE 488.1. Up to 31 peripheral devices can be connected.

(7) Serial ATA (Serial Advanced Technology Attachment)

Serial ATA is a high speed interface that is developed by modifying the parallel interface ATA (ATA/ATAPI-4) to the serial interface. Parallel interface ATA (ATA/ATAPI-4) was the official standard of **IDE (Integrated Device Electronics)** that was standardized by **ANSI (American National Standards Institute)**. It is generally used for connecting high-speed data transfer devices such as built-in hard disk and optical disc devices like CD-ROM drive. At present, serial ATAIII that has the communication speed of 6 Gbps is most widely used. While serial ATA basically connects devices and ports (controllers) on a one-to-one basis, it is possible to increase the number of ports by using a **port multiplier**. In the conventional ATA, a master-slave connection was used where two devices that have a master-slave relationship were connected to one cable. However, this method is not used in serial ATA.

(8) IrDA (Infrared Data Association)

IrDA is an association for creating standards for data communication using infrared rays, and it is also a serial interface that is created by this association. The maximum communication distance is 1 meter, and it is necessary to place the communication ports within ± 15 degrees without any shields such as partitions. Infrared communication that does not require connection cables is used for data exchange between personal digital assistants like smartphones or tablet terminals, and for data exchange between personal digital assistants and notebook PCs.

(9) Bluetooth

Bluetooth is a wireless technology standard that connects devices such as cell phones and PCs, and is an interface for exchanging data and voice by using the frequency band of 2.4GHz (2.402 to 2.480 GHz), which can be used all over the world. Unlike infrared rays, communication is possible even if there is a block or an obstacle between devices. Therefore, it allows the transfer of data between devices that are separated with low partitions.

[Bluetooth standards]

- Number of devices which can be connected

In a general standard, logically up to 7 devices can be connected simultaneously. But there is a standard that allows almost unlimited number of devices (about 2^{32} devices) to be connected by expanding the address bits from 3 to 32.

- Communication speed

Communication speed is determined by the version. In the case of Bluetooth 3.0, the maximum communication speed is 24 Mbps. In the more advanced standard of Bluetooth 4.0, the maximum communication speed can be limited to 1Mbps in order to support the low power consumption mode (BLE (Bluetooth Low Energy)).

- Communication distance

Communication distance is determined by the class. The following table gives approximate values of communication distance of each class.

Class	Communication distance
Class 1	About 100m
Class 2	About 10m
Class 3	About 1m

(10) HDMI (High-Definition Multimedia Interface)

HDMI is a standard of communication interface that transmits voice and video as digital signals. It was jointly created by Hitachi Ltd., Panasonic Corporation, Phillips, Thomson Multimedia, Sony Corporation, and Toshiba Corporation, with Silicon Image of the United States taking the lead. With a single cable where voice, video, and control signals are integrated, wiring of AV devices can be simplified to just one wire. While it was mainly developed as an input/output interface for home appliances and AV devices, at present it is widely used in PCs as well. The base of HDMI is the serial interface **DVI (Digital Visual Interface)**, which produces high quality images by transmitting digital signals instead of **analog RGB**, which sends video signals by using analog signals of three primary colors (RGB) of light.

5-5-3 Device Driver

(1) Device driver

Device driver is software that controls peripheral devices. Device driver is specific software

that is required for each peripheral device, and peripheral devices cannot be used without device drivers. For example, in order to use a printer, a device driver (printer driver) for the printer is required. Moreover, software that manages the device drivers is called a **device manager**. When multiple devices are connected, the device driver has another role to play in synchronizing with each device (aligning the timing of operation).

While each device driver for basic peripheral devices is provided in advance, we need to obtain a device driver for the other peripheral device (e.g., a newly purchased peripheral device) from the Internet or CD-ROM that is supplied with the device and then **install** it separately (i.e., install software in the computer).

(2) BIOS (Basic Input/Output System)

BIOS is software (program) that provides the standard means of input and output in computers. When the computer is started, the OS (software for controlling the computer) embeds the device drivers of the peripheral devices that are detected by the BIOS so that the use of peripheral devices become possible.

(3) PnP (Plug and Play)

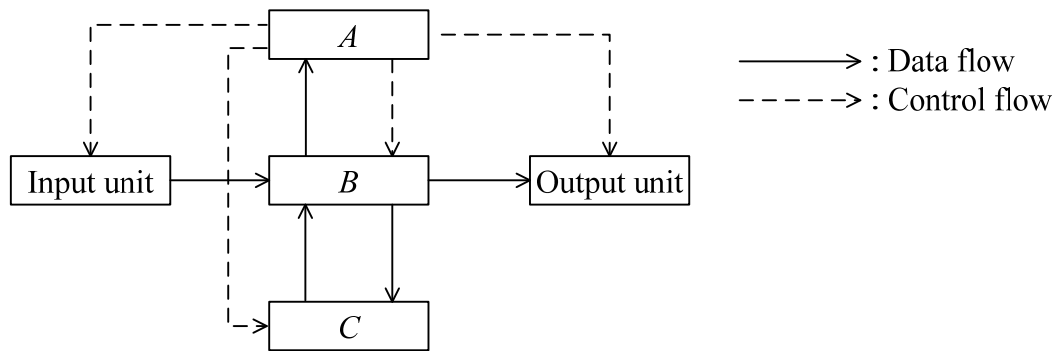
PnP (Plug and Play) is a function (mechanism) that allows the immediate use of a peripheral device by just connecting it to a computer. As for the basic mechanism, as soon as a peripheral device is connected, the OS recognizes the connected device and automatically sets the device driver so that the device can be used. Since the user does not need to perform any manual settings, even a beginner who does not have any knowledge of the device driver can easily use the computer and the peripheral device.

In order to implement a **hot plug** (a function that allows the connection and disconnection of a device while the power supply is kept on) that is offered by USB or IEEE 1394, it is necessary that the OS of the applicable computer support Plug and Play.

Chapter 1 Exercises

Q1

Which of the following is an appropriate combination of terms or phrases to be inserted into *A* through *C* in the figure that shows the basic configuration of a computer?



	<i>A</i>	<i>B</i>	<i>C</i>
a)	Arithmetic and logical unit	Storage unit	Control unit
b)	Storage unit	Control unit	Arithmetic and logical unit
c)	Control unit	Arithmetic and logical unit	Storage unit
d)	Control unit	Storage unit	Arithmetic and logical unit

Q2

How many times is the number of bit patterns that can be represented with 32 bits larger than the number that can be represented with 24 bits?

- a) 8 b) 16 c) 128 d) 256

Q3

Concerning four symbols of prefixes that show multiplication by the power of 10, namely, G (Giga), k (Kilo), M (Mega), and T (Tera), which of the following shows the correct magnitude relation between them?

- a) $G < k < M < T$ b) $k < G < T < M$
c) $k < M < G < T$ d) $M < T < G < k$

Q4

Which of the following is the decimal number that is equal to the binary number 101.11?

- a) 5.11 b) 5.3 c) 5.55 d) 5.75

Q5

Which of the following is the combination of binary, octal, decimal, and hexadecimal numbers that represent the same number?

	Binary	Octal	Decimal	Hexadecimal
a)	111	10	8	8
b)	1010	12	10	A
c)	1100100	256	100	64
d)	11111111	377	256	FF

Q6

Which of the following is the character code that is used in computers but does not have provisions for kanji characters?

- a) ASCII code b) EUC
c) Unicode d) Shift JIS code

Q7

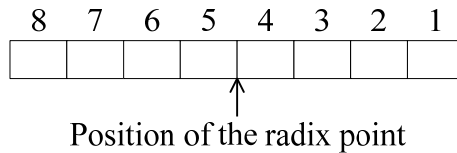
How many bytes are required when a 6-digit signed decimal number is represented in packed decimal?

- a) 3 b) 4 c) 6 d) 7

Q8

Which of the following is the decimal number -5.625 that is represented as a binary number using the 8-bit fixed point format?

Here, the position of the radix point is between the 4th bit and the 5th bit, and 2's complement is used for negative numbers.

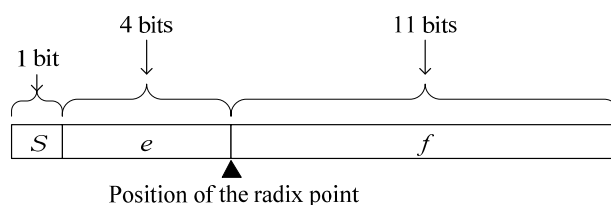


- a) 01001100 b) 10100101 c) 10100110 d) 11010011

Q9

When a numerical value is represented in the 16-bit floating point format as shown in the figure, which of the following is the normalized representation of the decimal number 0.25?

Here, normalization is the operation that adjusts exponent and fraction (mantissa) so that the most significant digit of the fraction (mantissa) does not become 0.



S: Sign of fraction (mantissa)
(0: Positive, 1: Negative)
e: Exponent (negative numbers are represented as 2's complement with 2 as radix)
f: Fraction (mantissa)
(binary number Represented as an absolute value)

- | | | | | | | | | | | | | | |
|---|------|-------------|-------------|---|------|-------------|---|---|------|-------------|---|------|-------------|
| <p>a) <table border="1" style="display: inline-table; text-align: center;"><tr><td style="width: 10%;">0</td><td style="width: 10%;">0001</td><td style="width: 80%;">10000000000</td></tr></table></p> <p>c) <table border="1" style="display: inline-table; text-align: center;"><tr><td style="width: 10%;">0</td><td style="width: 10%;">1111</td><td style="width: 80%;">10000000000</td></tr></table></p> | 0 | 0001 | 10000000000 | 0 | 1111 | 10000000000 | <p>b) <table border="1" style="display: inline-table; text-align: center;"><tr><td style="width: 10%;">0</td><td style="width: 10%;">1001</td><td style="width: 80%;">10000000000</td></tr></table></p> <p>d) <table border="1" style="display: inline-table; text-align: center;"><tr><td style="width: 10%;">1</td><td style="width: 10%;">0001</td><td style="width: 80%;">10000000000</td></tr></table></p> | 0 | 1001 | 10000000000 | 1 | 0001 | 10000000000 |
| 0 | 0001 | 10000000000 | | | | | | | | | | | |
| 0 | 1111 | 10000000000 | | | | | | | | | | | |
| 0 | 1001 | 10000000000 | | | | | | | | | | | |
| 1 | 0001 | 10000000000 | | | | | | | | | | | |

Q10

In order to obtain 32 times of a positive integer number that is represented in binary, how many bits should be shifted to the left? Here, there should not be any overflow.

- a) 4 b) 5 c) 6 d) 32

Q11

Which of the following is a memory device that allows rewriting and erasing data with electric signals and can retain the data even when the power is off?




- a) DRAM
- b) SRAM
- c) Flash memory
- d) Mask ROM

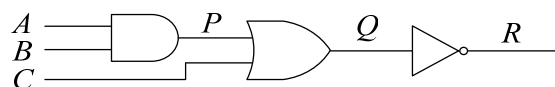
Q12

An instruction is composed of the instruction part and the address part. Among the methods that are used for creating an effective addresses from the address part, which of the following is the description of the absolute addressing?

- a) Using the value of a stack pointer as the reference address, adding the address part of the instruction as displacement from the reference address, and creating the effective address
- b) Using the value of the program counter as the reference address, adding the address part of the instruction as displacement from the reference address, and creating the effective address
- c) Using the content of the base register as the reference address, adding the address part of the instruction as displacement from the reference address, and creating the effective address
- d) Using the address part of the instruction as the effective address

Q13

In the logic circuit as shown in the figure, when $A=1$, $B=0$, and $C=1$, which of the following is an appropriate combination of P , Q , and R ? Here,  shows the AND circuit,  shows the OR circuit, and  shows the NOT circuit.



	P	Q	R
a)	0	1	0
b)	0	1	1

c)	1	0	1
d)	1	1	0

Q14

Among the combinations of an access time and a hit ratio related to cache memory and main memory, which of the following has the shortest effective access time of main memory?

	Cache memory		Main memory
	Access time (nanoseconds)	Hit ratio (%)	Access time (nanoseconds)
a)	10	60	70
b)	10	70	70
c)	20	70	50
d)	20	80	50

Q15

In a computer, one instruction is executed in the sequence of step 1 through step 6 shown in the table. How many nanoseconds does it take if six instructions are executed by using the pipeline process as shown in the figure? Here, an execution time of each step is 10 nanoseconds, and there is no branch instruction nor possible other instruction that disrupts the execution of the pipeline process.

Table Execution steps of instruction

Step	Process contents
1	Fetching of the instruction code part
2	Decoding of instruction
3	Fetching of the address part
4	Calculation of effective address
5	Fetching of data
6	Execution of computing

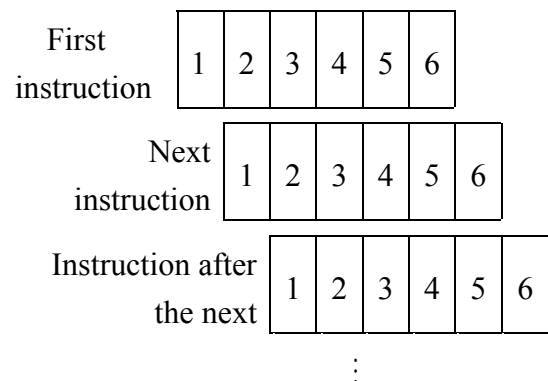


Figure Pipeline process of instruction execution

- a) 50 b) 60 c) 110 d) 300

Q16

Which of the following is suitable for multimedia processing for the reason that a single instruction performs the same operation on multiple sets of data in parallel?

- a) MIMD b) MISD c) SIMD d) SISD

Q17

There is a hard disk with a rotational speed of 5,000 rotations/minute and an average seek time of 20 milliseconds. The storage capacity of this hard disk is 15,000 bytes per track. What is the average access time (in milliseconds) that is required for transferring one block of data? Here, one block contains 4,000 bytes?

- a) 27.6 b) 29.2 c) 33.6 d) 35.2

Q18

Which of the following is an appropriate method for recording data on CD-R?

- a) Heating the recording film of a magnetized disc with laser beams, and recording by changing the direction of magnetization with a magnetic head
- b) Recording by changing the direction of magnetization of magnetic material that is coated on the disc with a magnetic head
- c) Having dual layer structure by combining discs and recording by changing the phase of record layer with laser beams
- d) Burning with laser beams on the disc that is coated with organic coloring matter, and recording by forming burn marks called pits on the layer of organic coloring matter

Q19

Which of the following is a type of coordinate reading device that is designed to feed two-dimensional and three-dimensional position information in computers, and is used for entering a CAD drawing and creating data of an NC (Numerical Control) machine tool?

- a) OCR b) XY plotter
c) Digitizer d) Light pen

Q20

Which of the following is the display that does not require a back-light because it emits light on its own when voltage is applied, and has features of low voltage drive and low power consumption?

- a) CRT display
- b) TFT liquid crystal display
- c) Plasma display
- d) OLED (Organic Light Emitting Diode) display

Q21

Which of the following is the most appropriate indicators that are used for evaluating the performance of a laser printer?

- a) The number of dots per 1 inch (2.54 cm) and the number of pages that can be printed in 1 minute
- b) The number of matrix dots that is used for printing one character and the number of characters that can be printed in 1 second
- c) Line spacing at the time of printing and the number of lines that can be printed in 1 second
- d) The types of characters at the time of printing and the number of characters that can be printed in 1 second

Q22

Which of the following is an appropriate description concerning the features of USB?

- a) It uses a high-speed transfer method that is suitable for transferring data that requires real-time data transfer such as voice and video. It can be connected with a daisy chain or tree structure, and it can even be connected without a PC that is used as a host computer.
- b) Peripheral devices are connected through a PC that is used as a host computer. There are multiple data transfer modes, and generally, a printer and a scanner are used in full-speed mode, while a keyboard and a mouse are used in low-speed mode.
- c) It is a serial interface, and originally was a standard that is used for connecting modems. However, it is also used for connecting PCs to peripheral devices.
- d) It is a parallel interface used for connecting compact computers such as PCs to peripheral devices such as a hard disk and a laser printer.

Chapter 2

Information Processing System

1 Processing Type of Information Processing System

The functions that are common to computer systems used at various places include input of certain data, execution of processing according to a fixed procedure, and output of result. These functions are collectively referred to as the information processing systems, but depending on the usage method and equipment configuration, the information processing systems are classified as various types.

1 - 1 Non-interactive Processing System and Interactive Processing System

(1) Non-interactive processing system

A **non-interactive processing system** is a system in which a set of instructions is issued all together to the computer. Once the computer starts the processing, the user (i.e., human) cannot intervene in the processing at all.

(2) Interactive processing system

An **interactive processing system** is a system in which data is processed by commands that a computer user sends one by one while the user looks at the information that is displayed on the screen. It can be called a method by which a human proceeds with their work as if the human were interacting with the computer.

In an interactive processing system, humans are also incorporated as part of the system, so it is necessary to improve the productivity of the user. Therefore, in order to implement a system that is easy for humans to use, it is necessary to enhance the **user interface**. A typical user interface includes the **window** and **icons** shown in Figure 2-1.

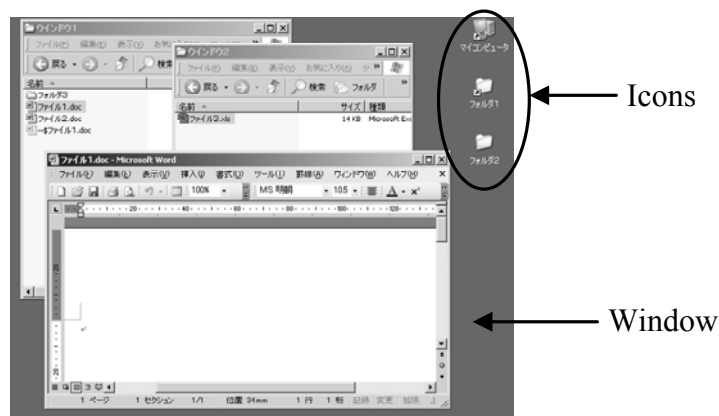


Figure 2-1 Example of a user interface

1 - 2 Batch Processing System and Real-time Processing System

1-2-1 Batch Processing System

A **batch processing system** is a system that accumulates the data to be processed by the computer and then processes it all together at a particular time. It is used in periodic tasks (e.g., calculation of employees' salary) that do not need to be performed immediately.

(1) Center batch processing

In **center batch processing**, data is accumulated at the site where the process is requested, and taken to the computer center after every fixed period or at every appointed date, and then processed.

- **Open batch processing**

This is a method that the user performs all processes from input to operation.

- **Closed batch processing**

This is a method that the operator performs all processes from input to operation.

- **Cafeteria method**

This is a method that the user performs input and the operator performs the operations.

(2) Remote batch processing

In **remote batch processing**, the processing that is requested from an off-site **RJE (Remote Job Entry) terminal** are sent to the computer center, accumulated, and then processed periodically.

[Transaction processing]

Transaction processing is a general term for the processes that are used to update the master file and database on the basis of the processing requests that are collected in the transaction file. The transaction processing requires the following **ACID characteristics**:

- **Atomicity**

It is a property that either “All are executed” or “None is executed.”

- **Consistency**

It is a property that inconsistencies do not occur because of the processing of the transaction.

- **Isolation** (separation)

It is a property that transactions do not interfere with one another.

- **Durability** (persistence)

It is a property that the recorded result continues to be retained even if a fault occurs.

1-2-2 Real-time Processing System

A **real-time processing system** is a system in which the computer immediately starts processing each time a process is requested. It is used in tasks that need to be performed immediately (when there is a strong restriction in the processing time) and in a situation where the system can perform work in place of humans.

- **Hard real-time system**

It is a real-time system such as an airbag control system in which there are strong restrictions in the response time, and if the processing does not finish within that time, fatal damage such as threats to life is sustained.

- **Soft real-time system**

It is a real-time system such as a seat reservation system in which there are restrictions in the response time, but no fatal damage is sustained even if the processing does not finish within that time.

(1) Real-time control processing system

A **real-time control processing system** is a system that replaces humans and sequentially interprets and processes the information obtained by a sensor or a set of sensors. It is used for controlling industrial robots in factories or used in autopilot systems of airplanes.

(2) OLTP (On-Line Transaction Processing) system

An **online transaction processing system** is a system that sends remotely-generated processing requests (transactions) to the computer center and performs the processing then and there. It is also known as **online real time processing** and is used in the ATM system of banks. In the online transaction processing, in most cases, the master file and database in the computer center are used as the processing targets. The online transaction processing is also a type of transaction processing, and therefore, requires **ACID characteristics**.

(3) TSS (Time Sharing System)

A **TSS (Time Sharing System)** is a system that several users perform the interactive

processing on a single computer as if each of them exclusively owned the computer. In reality, the usage time is divided up into minute units of time (**time slice**) for each user, and the use of the computer is assigned sequentially to each user.

1 - 3 Centralized Processing System and Distributed Processing System

A **centralized processing system** is a system that processes the data and resources centrally at one location, and all early information processing systems are this type. In contrast, a **distributed processing system** is based on a concept developed along with the progress that is made in network technology and is a system that achieves **load distribution** and **function distribution** by decentralizing the data and resources.

The characteristics of the respective system are compiled as follows:

Type of system Comparison items	Centralized processing system	Distributed processing system
System construction and operation	Easy	Difficult
Computer load	High (all processing performed on one computer)	Low (distributed among several computers)
System reliability	Low (the entire system goes down when there is a failure in one computer)	High (follow-up is possible even if there is a failure in one computer)
Security assurance	Easy	Difficult

1-3-1 Classification of Distributed Processing Systems

Depending on the relationship between the connected computers, the distributed processing systems are classified into a **horizontal distributed system** and a **vertical distributed system**.

(1) Horizontal distributed system

A **horizontal distributed system** is a distributed processing system in which all connected computers are in an equivalent position.

(a) P2P (Peer to Peer)

This is a horizontal distributed processing system that directly connects computers that

are in an equivalent position. While system installation is easy and the cost can also be kept low, all computers need functions that enable them to respond to the processing requests. Also, the shared resources within the network must also be managed. Therefore, this form is not suitable for large systems.

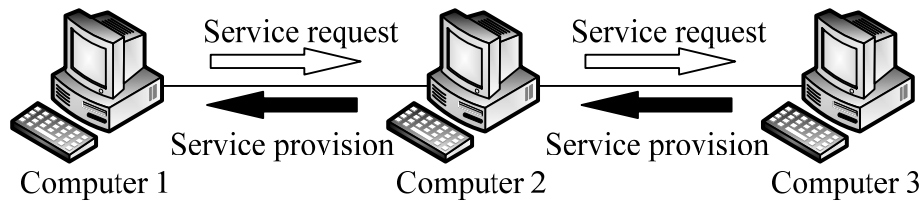


Figure 2-2 Image of a horizontal distributed system (peer to peer)

(2) Vertical distributed system

A **vertical distributed system** is a distributed processing system in which a clear hierarchical relationship is assigned to the connected computers.

(a) Client/server system

This is a vertical distributed processing system in which the computers connected to the network are classified into the **client** requesting the processing and the **server** providing the processing. In the client/server system, the client and server need not be the same computer nor OS (software controlling the computer). Moreover, the server can have a client function by which it can request another server to do some of the processes if necessary.

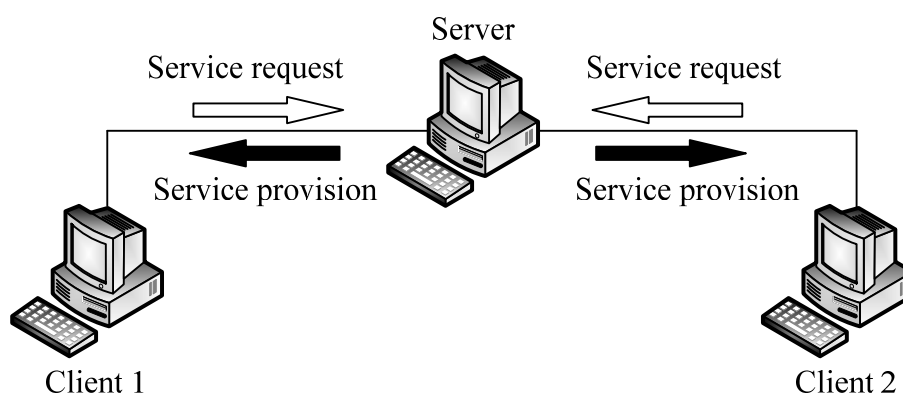


Figure 2-3 Image of a vertical distributed system (client/server system)

The typical types of servers used in a client/server system are described below. In the **server virtualization** technology, one server is used as multiple servers, and a single server can support several functions (e.g., the functions of a print server and file server).

- **Print server**

This server is equipped with a high-performance printer and manages the print processing.

- **File server**

This server is equipped with a large-capacity auxiliary storage device and manages files (or data) in a consolidated manner.

- **Database server**

This server is equipped with a large-capacity auxiliary storage device and manages the database in a consolidated manner.

- **Communication server (Gateway server)**

This server manages the connection between computers through different types of networks.

- **PROXY server**

This server is used to establish a connection with an external network (e.g., the Internet) in place of the client (acts as a proxy.)

- **Three-tier client/server system**

The past client/server systems were mainly **two-tier client/server systems** in which processing and calculation of data were performed at the client side and the server was requested only to do the processes of the database (e.g., search or update). However, when function expansion of applications is performed in a two-tier client/server system, it becomes necessary to update the application in all client terminals. Thus, a **three-tier client/server system** that is logically divided into the three-tier structure detailed below is designed.

Server	Database access layer
	Function layer
Client	Presentation layer

- (1) **Database access layer** (data layer)

This layer is used for accessing the database and referencing the required data.

- (2) **Function layer** (application layer)

This layer is used for processing (e.g., calculating and analyzing) messages (i.e., SQL statements) and data.

- (3) **Presentation layer**

This layer is used to implement the user interface (i.e., data input and result display) for exchanging data with the user.

A three-tier client/server system can independently manage the respective functions. Also,

since most of the functions are maintained at the server side, this system is also effective from the viewpoint of the application development work and function expansion. In addition, because of the fact that the server undertakes the functions of the database access layer and the function layer, the communication volume between the client and the server is reduced, thereby resulting in a reduction in communication load.

[Functions of the three-tier client/server system]

- **Stored procedure function**

This is a method that frequently used commands (e.g., a series of SQL statements for database query or update) are prepared beforehand in the server. This method was conceived of to eliminate the fact that the SQL statement communication between the client and the server results in network load in the two-tier client/server system that uses a database server, but it is also used in the three-tier client/server system.

- **Group commitment function**

This is a function by which several writing operations are performed in a synchronous manner in a server environment where several operations are performed in parallel.

- **Thin client system**

The concept that applications are placed at the server side and only the minimum functions are maintained at the client side is called **thin client**. The concept used in the thin client system (the software (or server) that implements the mechanism (i.e., system) of the thin client is called **thin client agent**). In the thin client system, there is no need to record the application (i.e., program) at the client side, and therefore, there is no auxiliary storage device at the **thin client terminal**; that is, the data is recorded in the **network storage**, which is a storage unit that can be used via the network. As a result, prevention of information leakage can be expected. Moreover, just like the thin client system, the procedures provided by the application (i.e., program) at the server side can be used by the client side as if they were present on the client side's computer. This communication method is called **RPC (Remote Procedure Call)**. RPC, which is one of the communication methods between programs, enables a computer to request another computer to handle part of the process.

- **Fat client**

This is a traditional method that all applications and the entire data are retained at the client side.

- **Rich client**

This is a method of downloading applications at the client side and then using them if necessary.

(b) **Web system**

This is a system that processing is requested to the **web server**, which provides information transmission services over the Internet called web services, from a **web browser** (e.g., software for seeking information from the web service and displaying it). This system can be referred to as a type of the client/server system. A typical web system is a system in which information is sought from the database. In such a case, the commonly used configuration is to divide the three-tier client/server system into two servers: the database access layer that becomes the database server, and the function layer that becomes the **application server** (i.e., web server). The client performs only the role of the presentation layer that displays the result received according to the description of **HTML (HyperText Markup Language)** on the screen. Here, HTML is the description language for web pages. Therefore, in most cases, a thin client terminal that has only the minimum required functions is used in a web system.

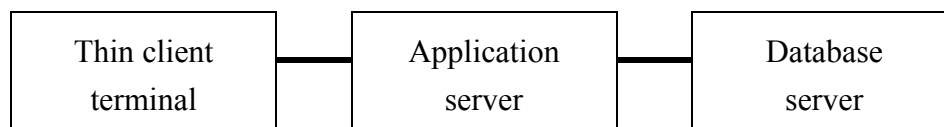


Figure 2-4 Configuration example of web system

(c) **Cloud computing**

This is a system that the user can easily receive services with high scalability (expandability) and availability by the computer resources (e.g., hardware and software) that are provided via the network. Cloud indicates that the user can use the service even without any knowledge of the mechanism at the other end of the cloud (i.e., the mechanism of providing the service). The user can use the service at any time as long as the minimum required environment (e.g., a client terminal like the PC, a web browser, and an Internet-connected environment) is available.

The typical services provided by cloud computing are described below:

- **SaaS (Software as a Service)**

This is a service that provides the functions of software. Generally, the **multi-tenant method** that is shared among several users is adopted.

- **PaaS (Platform as a Service)**

This is a service that provides the platform for running applications. A virtualized server and necessary databases are provided, and the user deploys and operates his/her own application.

- **IaaS (Infrastructure as a Service)**

This is a service that provides the infrastructure. Only a virtualized server is provided, and all other necessary OS and applications are prepared by the user.

- **DaaS (Desktop as a Service)**

This is a service that provides the desktop environment for the terminal. The virtual desktop environment on the cloud infrastructure is used from the thin client terminal.

[Backup site]

When a website (i.e., a collection of web pages) like a web system or a cloud computing system is used, a **backup site** must be prepared to deal with faults. The following backup sites become available in order of the fault recovery time from shortest to longest.

- **Hot site**

The data of a backup site that is configured by the same system is always updated to the most recent status via the network and is kept on standby.

- **Warm site**

The required data is uploaded periodically to the backup site and stored there.

- **Cold site**

Only the backup site is prepared, and the required data is transferred when a fault occurs.

1-3-2 Parallel Processing System

A **parallel processing system** is a system that achieves a reduction in the processing time and large-scale calculation processing by performing the distributed processing in parallel (i.e., by performing parallel processing).

(1) Multiprocessor system

Multiprocessor system is a general term for a system in which several computers are combined together. However, when we simply say “multiprocessor,” it could also indicate a single computer on which several processors (i.e., processor cores) are loaded. In a broad sense, cluster and grid computing are also included in a multiprocessor system.

(2) Cluster

A **cluster** (or **cluster system**) is a system in which several computers are linked and used as if they were a single high-performance computer; the method by which such a system is configured is called **clustering**. A cluster can be highly reliable because it can prevent the entire system from stopping by making other computer take over the processing when a fault

occurs in some computers configuring the system.

(3) Grid computing

Grid computing is a system (or mechanism) by which several computers that are spread across a wide region are connected over a network and processed in parallel, which enables their use as a virtual high-performance computer. It is a method by which a user can use the computer resources (e.g., CPUs, databases, and applications) that are connected over a network anytime, anywhere without being aware of the mechanism. It is used in HPC (High Performance Computing) that involves high computational complexity.

- **HPC (High Performance Computing)**

It is a process that involves extremely high computational complexity, which is performed to resolve natural phenomena. In comparison to the general parallel processing, it is sometimes referred to as **large-scale parallel processing**. An example of high performance computing includes the project for mapping the human genome (gene).

2 Configuration of High-reliability System

A point to consider in a system configured by computers and peripheral devices is the occurrence of a failure in devices. It is not possible for a device created by a human to have no failures at all. Therefore, the system configuration must be designed in view of the importance of the system.

2 - 1 Series System

A **series system** is a system configuration in which a backup system is not prepared for a fault. It has a reduced cost but low reliability, and therefore, is used in a system that does not pose any problems even if a device failure leads to stopping of the system.

(1) Simplex system

A **simplex system** is a system in which all devices are connected in a series. It can be referred to as the basic configuration in an online transaction processing system.

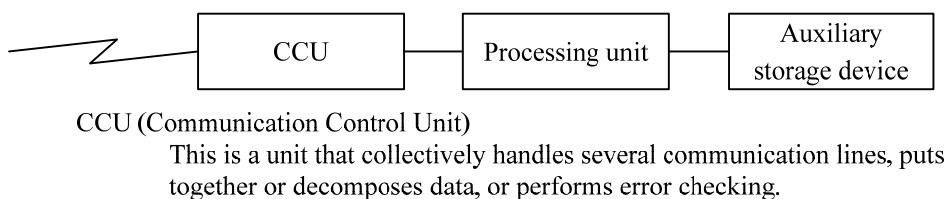


Figure 2-5 Configuration of a simplex system

(2) Tandem system

A **tandem system** is a system in which several processors are connected in series in order to distribute the load of the processors. Even if only one of the several processors fails, the entire system stops functioning.

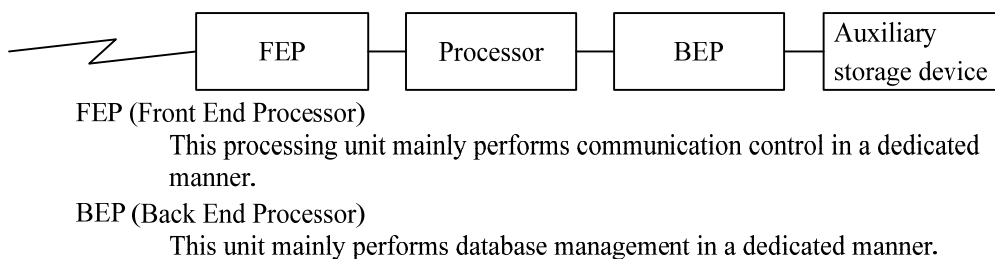


Figure 2-6 Configuration of a tandem system

2 - 2 Parallel System

A **parallel system** is a system configuration in which a backup system is prepared beforehand. Although the parallel system involves a large cost, it is used in systems that require high reliability and have a social impact at the occurrence of a system failure.

(1) Duplex system

A **duplex system** is a system where a backup system is prepared and switched to the backup system in the event of a fault. This system is configured by a regularly used **primary system** (**currently used system**) and a **secondary system** (**backup system**) that is on standby in case of a failure. Generally, an online operation is performed in the primary system and a local operation such as batch processing is performed in the secondary system.

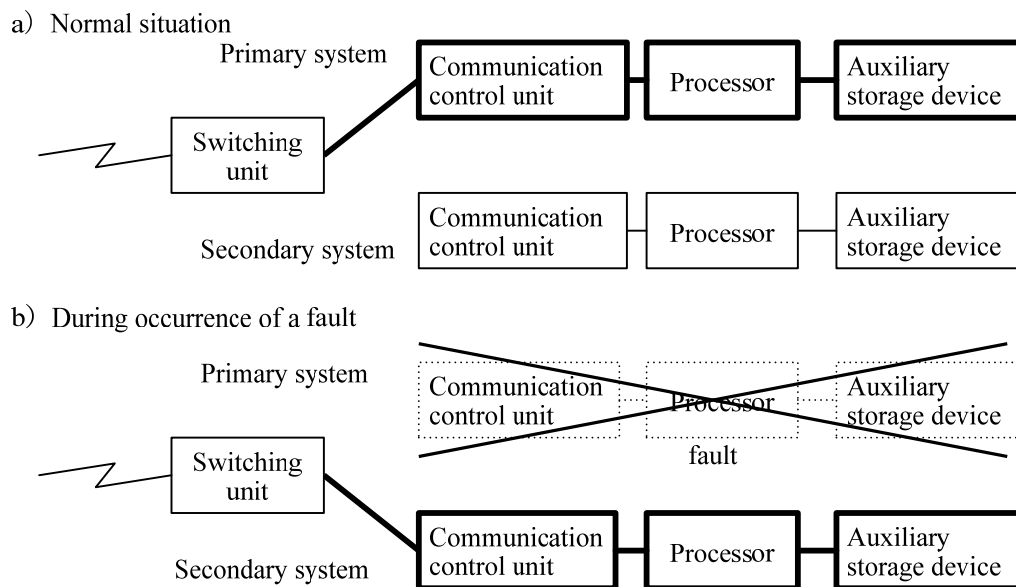


Figure 2-7 Configuration of a duplex system

- **Cold standby method**

This is a method that a separate local operation is executed (or stopped) in the secondary system (i.e., backup system). When a fault occurs in the primary system (i.e., currently used system), the operational system used in the primary system (i.e., currently used system) is activated in the secondary system (i.e., backup system), and the processing is inherited.

- **Hot standby method**

This is a method that the same operational system as the primary system (i.e., currently used system) is activated and kept on standby in the secondary system (i.e., backup system). The secondary system (i.e., backup system) monitors the primary system (i.e.,

currently used system) and immediately inherits the processing of the primary system (i.e., currently used system) when a fault occurs.

(2) Dual system

A **dual system** is a method that exactly the same processing is performed in two or more systems and the normality is confirmed (i.e., **cross-checked**) by collating the results. Although the cost per transaction is more than that of the duplex system, the reliability is extremely high as the processing is continued on the remaining system(s) when one system fails. It is used in systems (e.g., a medical system) concerning human life that must not stop under any circumstance.

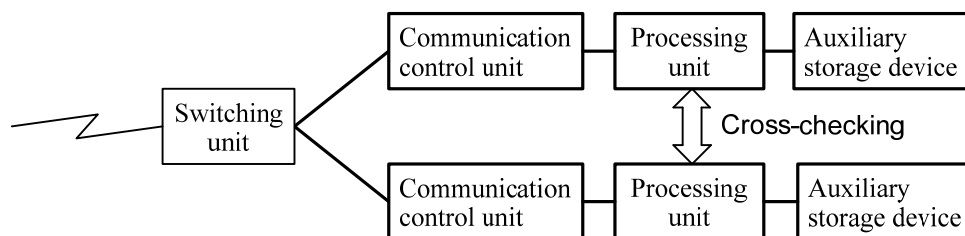


Figure 2-8 Configuration of a dual system

2 - 3 Multiplexing System

A **multiplexing system** is a system that improves reliability through the use of several devices. A parallel system can be referred to as a multiplexing system in which the entire system is targeted.

(1) Fault tolerant system

A **fault tolerant system** is a system that is designed to be able to retain the necessary functions of the overall system even if a fault occurs in part of the system.

[Hardware-based implementation method]

The computer (e.g., server) and hard disk configuring the system are duplicated in order to improve the **availability** (i.e., the ability of the system to be available when needed).

[Software-based implementation method]

For example, in **N-version programming**, several programs that have the same functions are executed simultaneously, and after the results are compared, the result that agrees with the majority of the programs is adopted.

Moreover, a **fail-soft structure**, such as the degraded operation of a system in which the main functions are not stopped even if several functions have been done away with, is also one of the fault tolerant systems.

(2) Multiprocessor system

A **multiprocessor system** is a system in which the processor is multiplexed. Each processor performs a parallel processing for a single purpose. Moreover, in the event of a fault, the corresponding processor is cut off, and the processing can be continued among the remaining processors, which results in an improvement in reliability.

In a multiprocessor system, the processing performance (i.e., processing speed) is enhanced each time the number of processors to be multiplexed is increased. However, according to **Amdahl's law**, which states that “the improvement in the processing performance in the parallel processing of a program is limited by the sequentially processed part,” even if the number of processors is increased by n times, the processing speed does not necessarily become n times (or processing time is $1/n$).

(a) **Tightly coupled multiprocessor system**

This is a system in which processors that are controlled by the same OS (i.e., software that controls the computer) use a shared main memory unit and perform the processing at the same time. Each task (i.e., unit in which the work to be processed is divided) of the system can be executed on any processor. As a result, the load can be distributed in smaller units, but a function for securing **synchronization** (i.e., timing of the processes) between tasks is necessary.

(b) **Loosely coupled multiprocessor system**

This is a system in which processors that are controlled by the each OS use an individual main memory unit and perform the processing at the same time. The consistency of the processing is achieved by transmitting the information of the main memory unit through a high-speed I/O port. A **cluster** can be referred to as such a method.

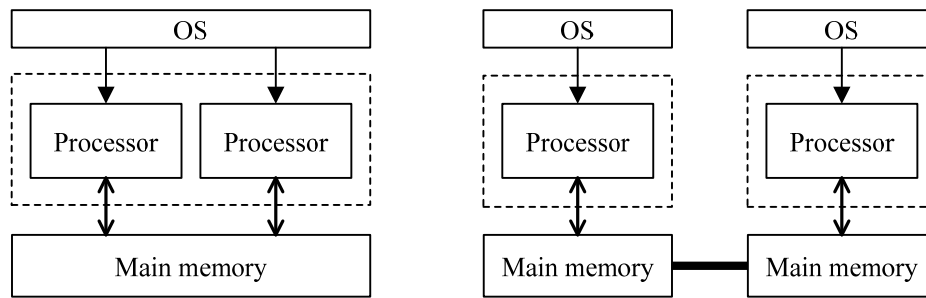


Figure 2-9 Image of a tightly coupled/loosely coupled multiprocessor system

- **Polyprocessor system**

This is a multiprocessor system in which the distribution of roles is performed for each processor.

- **Load sharing system**

This is a multiprocessor system in which tasks (or load) are distributed depending on the situation.

(3) RAID (Redundant Arrays of Inexpensive Disks)

RAID is a multiplexing system (or a device multiplexing technology) for hard disks. RAID improves reliability by handling several hard disks in an integrated manner and can be accessed without stopping the system even when a fault occurs. It is classified as described below depending on the recording method and position of data and redundant bits (i.e., bit used in error detection and correction).

- **RAID0 (striping or disk striping)**

The access speed is improved by splitting and writing a set of data on several disks. There is no spare disk for improving reliability

- **RAID1 (mirroring or disk mirroring)**

The reliability is improved by writing the same content at the same time in two disks that have the same capacity and using one disk for backup. Since the data is duplicated, the usage efficiency of the disk becomes 50% of the total capacity.

- **RAID2**

The access speed and reliability are improved by using several disks for saving the data and a disk for data recovery (**hamming code**).

- **RAID3**

The access speed and reliability are improved by using several disks for saving the data and a disk for data recovery (**parity code**).

- **RAID4**

The access speed and reliability are improved by performing data access in units of sectors in lieu of units of bits used for RAID3.

- **RAID5**

This is a configuration that the parity data recorded in the parity disk for data recovery in RAID4 is distributed to the disks for saving the data. The access speed and reliability are improved by reducing the load of the parity disk.

- **RAID6**

This is a method that two parity codes are generated for data recovery in RAID5.

RAID is also used in **network storage**. Typical network storage devices include **NAS (Network Attached Storage)** and **SAN (Storage Area Network)**.

NAS is a storage device that is used as a file server by connecting it directly to the network. File systems supporting several protocols (e.g., CIFS in Windows and NFS in UNIX) are provided within the storage. Therefore, data sharing between different OSs and machines (or servers) can be easily performed in units of files. On the other hand, SAN is a network storage system that is dedicated to data storage. While the load on the network is lower than in NAS, a demerit is that it is difficult to share data between OSs and machines (or servers) that have different file systems.

3 Evaluation of Information Processing System

In order to operate the information processing system, you must evaluate various functions of the system beforehand. However, the information processing system has various modes, and the usage purpose is also different. Therefore, it is difficult to evaluate on the basis of only one index or indicator. Therefore, there is a need for indexes or indicators that evaluate the information processing system from various viewpoints.

3 - 1 Evaluation of the Processing Power

The indexes (or indicators) for evaluating the processing power of the information processing system include the index (or indicators) for evaluating the processing power of the overall system and the index (or indicators) for evaluating the processing power of the CPU that is the core of the system

3-1-1 Evaluation of the Overall System

(1) Performance indicator of the processing time

The performance indicators of the processing time of the system include the **turnaround time** and the **response time**. Basically, the shorter these times, the higher the evaluation.

- **Turnaround time**

This is the time that elapses from the start of a processing request to the system until the acquisition of the complete output result. It is used as the indicator for performance evaluation of a batch processing system.

Example: Turnaround time of the batch processing system

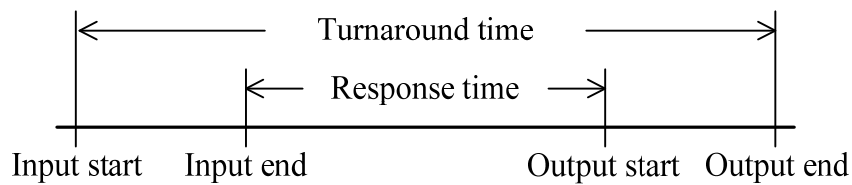
$$= \text{CPU processing time} + \text{input/output time} \\ + \text{overhead time (e.g., the waiting time)}$$

- **Response time**

This is the time that elapses from the completion of a processing request to the system until the start of the result output. It is used as the indicator for performance evaluation of an online system.

Example: Response time of an online transaction processing system

$$= \text{CPU processing time} \\ + \text{overhead time (e.g., the line transmission time} \\ \text{and the terminal processing time)}$$



(2) Performance indicator of the amount of work

Throughput is the performance indicator of the amount of work in a system. Basically, the larger the amount of work that can be processed, the higher the evaluation.

- **Throughput**

This is the amount of work that is processed within a unit time (fixed time) in the information processing system. **TPS (Transactions Per Second)** (i.e., number of transactions that can be processed in one second) can be used as the unit. For example, when the upper limit of the CPU utilization is 80% in a web server in which 5 milliseconds of CPU time is taken for processing one transaction, the throughput of the web server is calculated as follows:

$$\begin{aligned}\text{Throughput of the web server} &= 1 \text{ second} \div 5 \text{ milliseconds/transaction} \times 0.8 \\ &= 160 \text{ TPS (transactions/second)}\end{aligned}$$

However, this is the throughput of an individual web server, and the throughput of the entire information processing system in which several devices (i.e., servers) are combined together is the throughput of the device (i.e., server) that has the lowest processing power.

In addition to evaluation of the operating information processing system, the turnaround time/response time and the throughput are used as target values (i.e., indicators) during development or enhancement of the information processing system. First, the necessary target values (i.e., system performance) are set. Next, the various performance requirements and system configuration (i.e., enhancement plan) are designed in order to implement the target values. This is called **capacity planning**.

[Basic procedure of capacity planning]

- 1) Collecting the workload
Collect information, such as the type and amount of processing and the processing time that are required for the system. (The load information during peak time is particularly important.)
- 2) Determination of the performance requirements of the system
Analyze the collected information, and determine the performance requirements of the system on the basis of the **load forecast** and creation of the prototype.
- 3) **Sizing**
Estimate the performance requirements of the hardware, software, and peripheral devices necessary for fulfilling the performance requirements of the system.
- 4) Evaluation/improvement
Evaluate the results of sizing, and then make improvements (i.e., tuning) if necessary. Even after the system starts operating, repeatedly perform evaluation and improvements in order to ensure that the storage capacity and the system throughput are not insufficient.

3-1-2 Evaluation of the CPU

(1) Clock frequency

Clock frequency represents the speed at which a clock signal is generated. Since the operation of the CPU is performed in synchronization with this signal, basically, the larger the clock frequency, the higher the processing power becomes. However, since the **CPI (Cycles Per Instruction)**, which is the number of clocks necessary for executing one instruction, may differ depending on the CPU, the performance of the CPU cannot be compared (evaluated) only through the clock frequency.

(2) MIPS (Million Instructions Per Second)

MIPS represents the number of instructions that can be executed in 1 second in units of one-million instructions. A CPU with 1 MIPS is a CPU that can execute one million instructions in 1 second. Basically, the larger the number of instructions that can be executed, the higher the evaluation becomes. However, there is no sense in using this indicator as it is for comparing CPUs of different models in which the content of the instructions to be executed is also different.

MIPS is determined on the basis of the execution time of one instruction (or the **cycle time**

from the receipt of an instruction until the next instruction can be received.) However, since the execution time of an instruction differs depending on the type, a method of calculating the weighted average on the basis of the occurrence rate of an instruction is used to calculate the average instruction execution time. At this time, on the basis of the existence of instructions that are used often and instructions that are not used very often, an **instruction mix** is used as the occurrence rate of standard instructions depending on the processing of applications. Typical instruction mixes include the **commercial mix**, which focuses on a transfer instruction used frequently in business processing calculations and the **Gibson mix**, which focuses on an operation instruction used frequently in scientific and technological calculations. Moreover, the instruction execution time for executing one instruction can also be calculated from the CPI and clock frequency.

Example: When a CPU that operates at a clock frequency of 500 MHz requires an average of 5 CPI for executing one instruction, what is the performance (in MIPS) of the CPU?

- (1) Average execution time of one instruction
 - = CPI × Clock time (basic operation time)
 - = 5 clocks/instruction × (1 second ÷ 500,000,000 clocks)
 - = 0.00000001 seconds/instruction
- (2) Number of instructions that can be executed in 1 second
 - = 1 second ÷ Average execution time of one instruction
 - = 1 second ÷ 0.00000001 seconds/instruction
 - = 100,000,000 instructions → 100 MIPS

Moreover, as a further interpretation of the unit of MIPS, it is understood that a CPU with 1 MIPS can execute one instruction in 1 microsecond. Therefore, the calculation of MIPS can be summarized as shown below.

$$1 \text{ MIPS} = 1 \times 10^6 \text{ instructions/second}$$

$$= \frac{1 \times 10^6 \text{ instructions}}{1 \text{ second}} = \frac{1 \times 10^6 \text{ instructions}}{1 \times 10^6 \text{ microseconds}} = \frac{1 \text{ instruction}}{1 \text{ microsecond}}$$

$$= 1 \text{ instruction/microsecond}$$

- The average instruction execution time of a CPU with x MIPS is $1/x$ (microseconds).
- The performance of a CPU with the average instruction execution time of y (microseconds) is $1/y$ (MIPS).

Example: When the instruction mix of a CPU is the values that are shown in the table below, what is the performance (in MIPS) of the CPU?

Instruction type	Instruction execution time	Occurrence rate
Register-to-register operation	0.1 microseconds	40%
Register to/from memory operation	0.3 microseconds	50%
Unconditional branch	0.6 microseconds	10%

(1) Average execution time of one instruction

$$= \Sigma (\text{Instruction execution time} \times \text{Occurrence rate})$$

$$= 0.1 \text{ microseconds} \times 0.4 + 0.3 \text{ microseconds} \times 0.5 + 0.6 \text{ microseconds} \times 0.1$$

$$= 0.04 \text{ microseconds} + 0.15 \text{ microseconds} + 0.06 \text{ microseconds}$$

$$= 0.25 \text{ microseconds}$$

(2) CPU performance (MIPS)

$$= 1 \div \text{Average instruction execution time (microseconds)}$$

$$= 1 \div 0.25$$

$$= 4 \text{ (MIPS)}$$

Nowadays, **GIPS (Giga Instructions Per Second)**, which represents the number of instructions that can be executed in 1 second in units of giga (10^9) instructions, may also be used.

(3) **FLOPS (FLoating-point Operations Per Second)**

FLOPS is a unit that represents the number of floating-point instructions that can be executed in 1 second. A CPU with 1 FLOPS is a CPU that can execute one floating-point operation in 1 second. Currently, **TFLOPS (Tera FLOPS)**, which represents the number of floating-point instructions that can be executed in 1 second in units of tera (10^{12}) instructions, is used as the performance evaluation indicator of a supercomputer and such other high-performance computer.

3-1-3 Techniques of Performance Measurement

(1) Benchmark test

A **benchmark test** is a method of measuring the processing performance (e.g., the time

required) by executing a standard program (i.e., software) for an evaluation in line with the usage purpose. It is called **benchmarking** to perform measurement (or the measurement standard), and the program that is used is called a **benchmark program**. Since the measurement target of each benchmark test is different, it is necessary to understand the characteristics of the system in an integrated manner by executing several benchmark tests.

- **TPC benchmark**

This is a benchmark defined by the TPC (Transaction Processing Performance Council) that promotes standardization of an OLTP system for its evaluation, and TPC-A through TPC-D are defined depending on the applications. System cost per TPS or unit performance (cost/TPS) is used as the performance indicator. In the currently used TPC-C (for business processing OLTP), a unit set with reference to 1 minute (tpmC, Price/tpmC) is used.

- **SPEC benchmark**

This is a benchmark defined by the industrial standard body, SPEC (System Performance Evaluation Council), in order to evaluate the performance of the CPU. This benchmark test can be performed on UNIX.

- **SPECint**: A benchmark focusing on integer type operations
- **SPECfp**: A benchmark focusing on floating-point type operations

- Other benchmarks

- Dhrystone: For evaluation of fixed-point number operations
- Whetstone: For evaluation of floating-point number operations
- Livermore Fortran Kernel: For evaluation of a vector computer/compiler
- Linpack: For evaluation of floating-point number operations (simultaneous linear equation solution program)

(2) Kernel program test

Kernel program test is a method of executing a simple program such as a basic computing problem, measuring the execution time, and comparing the CPU performance.

(3) Monitoring

Monitoring is a method of incorporating beforehand a mechanism for measurement in the information processing system that is the target of measurement and measuring the actual operation status.

- **Software monitoring**: A method of incorporating a program (i.e., software) for measurement

- **Hardware monitoring:** A method of incorporating a device (i.e., hardware) for measurement

3 - 2 Evaluation of Reliability

3-2-1 Concept of Reliability

Various information processing systems are operated in a modern society. Conversely, it can be said that a modern society cannot be set up without an information processing system. It is but obvious that an information processing system playing such an important role is required to have high reliability. The concepts for constructing a highly reliable system are classified as shown below.

(1) Concept of a fault

(a) **Fault tolerant** (fault tolerant technique)

This is a concept of creating a mechanism for minimizing the effect of a fault on the basis of the assumption that a fault will definitely occur in a device or system created by a human. A system that is constructed on the basis of this concept is called a **fault tolerant system**.

• **Fail-soft**

This is a type of fault-tolerance and is a concept based on which the operation is continued by partially restricting the effect of the fault. An example of this concept is **degraded operation** where the main important functions are not stopped even if several functions have been done away with. In the case of a power failure in a hospital, the lighting is kept to a minimum and the power of the private power generator is concentrated on artificial life support.

(b) **Fault avoidance** (fault avoidance technique)

This is a concept that a fault is not allowed to occur in a system at any cost. Since in reality it is not possible to prevent a fault from occurring, it can be referred to as a concept (or technique) that the possibility of the occurrence of a fault is brought as close to zero as possible.

(2) Concept of safety

(a) **Fail-safe**

This is a concept of stopping the system in order to avoid the impact of the occurrence of a failure on other parts by giving priority to safety. An example of this

concept is the control system of a traffic signal. When a failure occurs, all signals are turned to red to prevent accidents.

(b) **Foolproof**

This is a concept of preventing an operator from making a mistake so that the mistake of the person operating the system does not affect the overall system. Examples of this concept include washing machines that do not operate unless the lid is closed and the checking and correction of input data for any mistakes.

3-2-2 Index of Reliability

There are indexes and indicators that evaluate such as the reliability of an information processing system. This set of indexes and indicators is called **RASIS** which is obtained by arranging together the first letter of the following five items. It is also called **RAS** which is obtained by arranging together the first letter of the top three items.

Index/Indicator	Meaning
Reliability	Assuring that the system is operating properly
Availability	Assuring that the system is in the appropriate operating status when it is to be used
Serviceability	Assuring that measures (i.e., recovery) are easy to perform when a fault occurs.
Integrity	Assuring the correctness of the data
Security	Assuring the security protection of the system and data

Among these five indexes and indicators, R (Reliability), A (Availability), and S (Serviceability) are evaluated by using the scale shown below.

(1) **MTBF (Mean Time Between Failures)**

This is the mean operation time from the recovery of the system (or device) from a failure until the occurrence of the next failure. It is used to represent R (i.e., Reliability).

(2) **MTTR (Mean Time To Repair)**

This is the mean time from the occurrence of a failure in the system (or device) until its recovery. It is used to represent S (i.e., Serviceability).

(3) **Availability**

This is the probability of the normal operation of the system (or device) and is used to represent A (i.e., Availability). Availability indicates the ratio of normal operation of the system per unit time and can be calculated using MTBF and MTTR as shown

below.

$$\text{Availability} = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$$

The probability that the system (or device) is not operating with respect to the probability that the system (or device) is operating (availability) is called **non-availability**. Non-availability indicates the ratio of the system failure per unit time and can be calculated as shown below.

$$\text{Non-availability} = 1 - \text{Availability} = \frac{\text{MTTR}}{\text{MTBF} + \text{MTTR}}$$

Example: The operating status of a device when it is operated for 600 hours is as shown in the table below. Calculate the MTBF, MTTR, and availability of this device.

Period of time	Number of elapsed hours	Status
0 to 120 hours	120 hours	Normal operation
120 to 140 hours	20 hours	Repair
140 to 290 hours	150 hours	Normal operation
290 to 300 hours	10 hours	Repair
300 to 480 hours	180 hours	Normal operation
480 to 510 hours	30 hours	Repair
510 to 600 hours	90 hours	Normal operation

(1) MTBF (Mean Time Between Failures)

$$\begin{aligned}
 &= \text{Total operating time} \div \text{Number of failures} \\
 &= (120 \text{ hours} + 150 \text{ hours} + 180 \text{ hours} + 90 \text{ hours}) \div 3 \\
 &= 180 \text{ hours}
 \end{aligned}$$

(2) MTTR (Mean Time To Repair)

$$\begin{aligned}
 &= \text{Total repair time} \div \text{Number of failures} \\
 &= (20 \text{ hours} + 10 \text{ hours} + 30 \text{ hours}) \div 3 \\
 &= 20 \text{ hours}
 \end{aligned}$$

(3) Availability

$$\begin{aligned}
 &= \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}} \\
 &= \frac{180 \text{ hours}}{180 \text{ hours} + 20 \text{ hours}} \\
 &= 0.9
 \end{aligned}$$

* In this example, the operating status of Figure A is averaged as shown in Figure B.

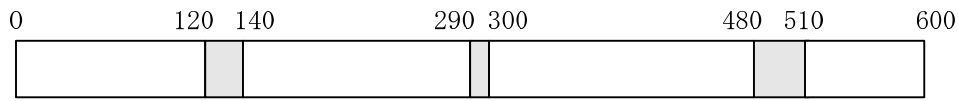


Figure A Operating status of a device

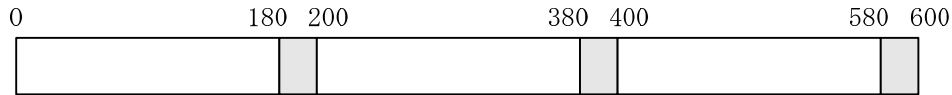


Figure B Averaged status

3-2-3 Availability and Failure Rate of the System

Generally, a system is configured by several devices. At this time, the availability of the overall system can be calculated from the availability of each device.

(1) Availability of a series system

In the case of a **series system**, the system operates normally as a whole only when all devices are operating. In the case of a series system shown in Figure 2-10, the operating status of each device is summarized as shown in the table below.

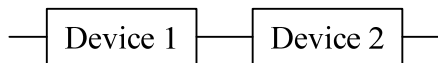


Figure 2-10 Example of a series system

	Device 1	Device 2
1)	○ Operating	○ Operating
2)	○ Operating	× Failure
3)	× Failure	○ Operating
4)	× Failure	× Failure

According to this table, the system operates as a whole only in the case of 1) when both devices are in the operating status. Therefore, the availability of a series system is calculated by the expression below.

$$\text{Availability of a series system} = \text{Availability of device 1} \times \text{Availability of device 2}$$

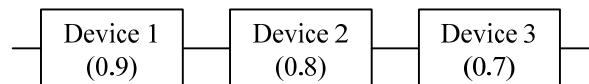
Similarly, even if the number of devices is increased to three or four, the series system operates normally only when all devices are operating. Therefore, the availability of the series system is calculated as the result obtained by multiplying the availability of all devices that make up the

system.

Availability of a series system in which n devices are connected
 $= \text{Availability of device 1} \times \text{Availability of device 2} \times \dots \times \text{Availability of device } n$

Since the availability is less than 1, the value keeps getting smaller with each repeated multiplication. That is, the higher the number of devices connected in a series system, the lower the availability becomes.

Example: Calculate the availability of the system shown in the figure below. The numeric value within parentheses is the availability of each device.



Availability of the series system

$$\begin{aligned}
 &= \text{Availability of device 1} \times \text{Availability of device 2} \times \text{Availability of device 3} \\
 &= 0.9 \times 0.8 \times 0.7 \\
 &= 0.504
 \end{aligned}$$

(2) Availability of a parallel system

In the case of a **parallel system**, the system may operate as a whole even when not all devices are operating. In the case of a parallel system shown in Figure 2-11, the operating status of each device is summarized as shown in the table below.

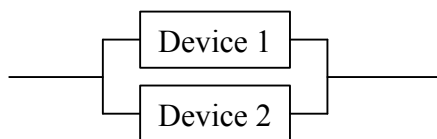


Figure 2-11 Example of a parallel system

	Device 1	Device 2
1)	○ Operating	○ Operating
2)	○ Operating	× Failure
3)	× Failure	○ Operating
4)	× Failure	× Failure

At this time, the functionality of the system is conserved (or the system operates as a whole) even if only either of the two devices is operating, and thus, the system operates as a whole in the case of 1) through 3). Thus, the availability of the overall system can be calculated by adding up the probability of 1) through 3). However, since in this way the calculation becomes tedious, it is assumed that among all operating conditions, the system is in a non-operating status only in the case of 4), and the total probability of 1) through 3) is calculated by subtracting the probability of 4) from the total probability “1” of all operating conditions. Therefore, the availability of a parallel

system is calculated by the expression shown below.

Availability of a parallel system

$$= 1 - \text{Non-availability of device 1} \times \text{Non-availability of device 2}$$

$$= 1 - (1 - \text{Availability of device 1}) \times (1 - \text{Availability of device 2})$$

Similarly, as shown in Figure 2-12, consider a case in which three or more devices are connected in parallel.

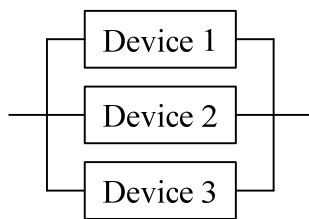


Figure 2-12

Example of a parallel system
with three devices

	Device 1	Device 2	Device 3
1)	○ Operating	○ Operating	○ Operating
2)	○ Operating	○ Operating	× Failure
3)	○ Operating	× Failure	○ Operating
4)	× Failure	○ Operating	○ Operating
5)	○ Operating	× Failure	× Failure
6)	× Failure	○ Operating	× Failure
7)	× Failure	× Failure	○ Operating
8)	× Failure	× Failure	× Failure

In this case, the following combinations can be assumed in consideration of the minimum number of devices with which the system operates normally as a whole.

[When the system operates as a whole if one or more devices are operating]

$$\text{Availability of the system as a whole} = 1 - \text{Probability of 8)}$$

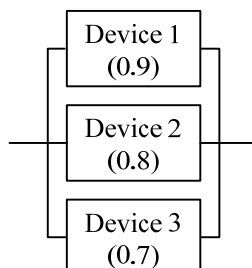
[When the system operates as a whole if two or more devices are operating]

Availability of the system as a whole

$$= \text{Probability of 1)} + \text{Probability of 2)} + \text{Probability of 3)} + \text{Probability of 4)}$$

In a parallel system, the more devices you connect, the higher the availability becomes.

Example: Calculate the availability of the system shown in the figure below that operates normally if two or more devices are operating. The numeric value within parentheses is the availability of each device.



Availability of a parallel system

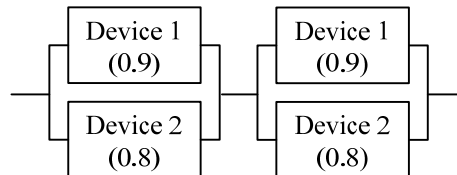
(when the system operates if two or more devices are operating)

$$\begin{aligned}
 &= \text{Probability that all three devices are operating} \\
 &\quad + \text{Probability that only two devices are operating} \\
 &= (0.9 \times 0.8 \times 0.7) \quad \cdots \text{Probability of 1)} \\
 &\quad + 0.9 \times 0.8 \times (1 - 0.7) \quad \cdots \text{Probability of 2)} \\
 &\quad + 0.9 \times (1 - 0.8) \times 0.7 \quad \cdots \text{Probability of 3)} \\
 &\quad + (1 - 0.9) \times 0.8 \times 0.7 \quad \cdots \text{Probability of 4)} \\
 &= 0.504 + 0.216 + 0.126 + 0.056 \\
 &= 0.902
 \end{aligned}$$

(3) Availability of a series/parallel mixed system

The availability of a system in which both a series system and a parallel system are mixed is calculated on a part-by-part basis, and then the availability of the overall system is calculated.

Example: Calculate the availability of the system shown in the figure below. The numeric value within parentheses is the availability of each device, and the parallel part is fine even if only one of the devices is operating.



Availability of a series/parallel mixed system

$$\begin{aligned}
 &= \text{Availability of the No. 1 parallel part (left half)} \\
 &\quad \times \text{Availability of the No. 2 parallel part (right half)} \\
 &= \{1 - (1 - 0.9) \times (1 - 0.8)\} \times \{1 - (1 - 0.9) \times (1 - 0.8)\} \\
 &= 0.98 \times 0.98 \\
 &= 0.9604
 \end{aligned}$$

(4) Failure rate of the system

Failure rate is the probability of the occurrence of a failure (or a fault) in a device or system. Since a failure occurs once after the device has been operating for a fixed period of time, the failure rate of the individual device becomes the reciprocal of MTBF.

$$\text{Failure rate} = \frac{1}{\text{MTBF}}$$

Moreover, when several devices (or components) have been combined together, the overall failure rate becomes the total sum of the failure rate of each device (or component).

Failure rate of the system = Σ Failure rate of each individual device

Note: Σ indicates the total sum.

Example: Calculate the MTBF of a CPU configured by 100,000 components, each with a failure rate of 10^{-8} .

(1) Failure rate of the CPU

= Failure rate of component 1

+ Failure rate of component 2 + \dots + Failure rate of component 100000

= $10^{-8} \times 100,000$

= 10^{-3}

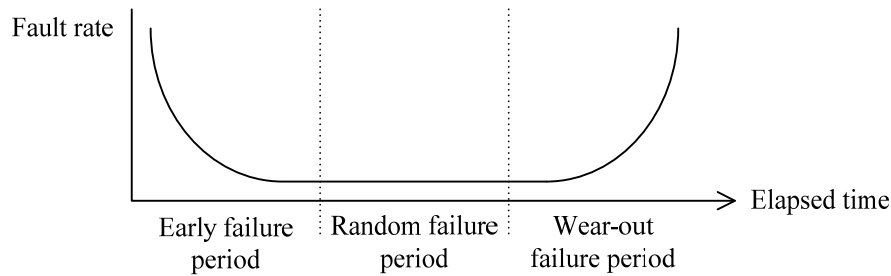
(2) MTBF of CPU

= $1 \div \text{Failure rate of CPU}$

= $1 \div 10^{-3}$

= 1,000

If the change in the failure rate of the hardware is represented by a graph with the elapsed time plotted on the horizontal axis and the failure rate (or occurrence frequency of failure) plotted on the vertical axis, this results in a **bathtub curve** in most cases.

[Bathtub curve]

- Early failure period: Period when the early failure occurs during or after installation
- Random failure period: Period when a failure occurs because of an accidental operation error or such other factor
- Wear-out failure period: Period when a failure occurs because of the degradation of a machine that has exceeded its useful life

3 - 3 Evaluation of Cost Efficiency

The indicator for evaluating the cost efficiency of an information processing system is **TCO (Total Cost of Ownership)**. TCO is the total cost including all the costs that are incurred during and after system installation.

(1) Initial cost

Initial cost is a one-time cost that is incurred during system installation. The typical initial cost includes the following costs:

- **Hardware purchase cost**
This is the cost of purchasing hardware (i.e., equipment) configuring the system.
- **Software purchase cost**
This is the cost of purchasing software when commercially available software (i.e., a set of applications) is used with the system.
- **Software development cost**
This is the cost of developing software when software developed exclusively by one's own company is to be used with the system. This cost is incurred as the personnel cost of development staff members when a product is developed in-house and as the outsourcing cost when another company is requested to carry out the development process.

(2) Operational cost (running cost)

Operational cost (running cost) is a periodic/persistent cost that is incurred during the operation after system installation. The typical operational cost includes the following costs:

- **Hardware rental/lease cost**

This is the cost during lease of hardware (i.e., equipment) configuring the system.

- **Operator cost**

This is the cost of salaries paid to operators operating the system.

- **Facilities maintenance cost**

This is the maintenance cost for maintaining and managing facilities (e.g., the computer center and network environment.)

The **direct cost** refers the cost that has a direct relationship with the target, such as the hardware and software purchase cost. On the other hand, the **indirect cost** refers to the cost that has an indirect relationship with the target. This is not an absolute but a relative classification. For example, in the case of a full-time operator of the target system, the personnel cost of the operator is classified as a direct cost. However, in the case of an operator of the overall in-house system including the target system, the personnel cost must be classified as an indirect cost.

4 Human Interface

A human interface is the contact point between a human and a system (i.e., computer) or service. Particularly, a user interface that is the contact point between a human and a computer is emphasized in an interactive processing system. In order to create or provide a system or service that is easy for humans to use, it is necessary to have a sufficient understanding of the human interface.

4 - 1 Human Interface Technology

(1) Information architecture

Information architecture is a representation technique that is used to arrange information in an easy-to-search manner and transmit it in an easy-to-understand manner.

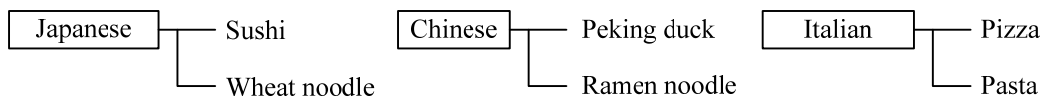
The information architecture starts from the task of arranging the information to be transmitted. The procedure of arranging general information is as shown below.

1) Organization of information

Classify and arrange the information to be transmitted in alphabetical order and category order. For example, when information concerning a restaurant is to be transmitted, it is classified into categories, such as sushi, wheat noodle, Peking duck, ramen noodle, and pizza, pasta. At this time, the collection of the classified information is called a **chunk** and the name (e.g., sushi, wheat noodle) given to the collection of information is called a **label**.

2) Structuring of information

Classify and arrange the organized information in a hierarchical order. For example, classify sushi and wheat noodle as Japanese, Peking duck and ramen noodle as Chinese, and pizza and pasta as Italian (this process is also referred to as “Tagging”).



In order to transmit information that is thus classified and arranged in an easy-to-understand manner, use **navigation** (mechanism of guiding to the destination). For example, on the Internet, by using a site map in which labels are displayed in a hierarchical order, a quick search for the target information is enabled.

(2) Human interface

A **human interface** is the contact point between a human and a system (i.e., computer) or service. In a human interface, consideration is given to the two points below.

(i) Usability

This is the “ease of use” for a user. In **ISO 9241-11 (JIS Z 8521)**, the concept of usability is defined as described below.

[Concept of usability [ISO 9241-11 (JIS Z 8521)]]

Usability is defined as “extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use.”

- Effectiveness:

The accuracy and completeness with which users achieve specified goals

- Efficiency:

The resources expended in relation to the accuracy and completeness with which users achieve goals

- Satisfaction:

The freedom from discomfort, and positive attitudes towards the use of the product

(ii) Accessibility

This is the “ease of access” for a user. In **ISO/IEC 24786 (JIS X 8341)**, the following considerations are required as information accessibility.

[Considerations for Information Accessibility [ISO/IEC 24786 (JIS X 8341)]]

Enable elderly and disabled persons to operate information and communications equipment along with software and to use services provided with them, without any obstruction.

ISO 9241-110 (JIS Z 8520) defines seven dialog principles that are necessary for designing an ergonomically desirable **interactive system**. These seven dialog principles constitute the policy for avoiding poor usability of usage difficulty in a user interface.

[Seven dialog principles [ISO 9241-110 (JIS Z 8520)]]

(1) Suitability for the task

(5) Controllability

(2) Self-descriptiveness

(6) Error tolerance

- (3) Conformity with user expectations (7) Suitability for individualization
 (4) Suitability for learning

Note: The numerical order (1) through (7) does not indicate any priority.

[Interactive system]

Since “interactive” means bidirectional, an interactive system is a system in which information is generally exchanged in both directions.

However, according to ISO 9241-110 (JIS Z 8520) as shown below, an interactive system is used in slightly restricted meaning that is defined in **ISO 13407 (JIS Z 8530)**.

“Combination of hardware and software components that receive input from, and communicate output to, a human user in order to support his or her performance of a task”

When user information is entered in this interactive system, in addition to the use of the keyboard and mouse by the user, an interface that makes use of voice and video is used.

- **Natural-language interface**

This is an interface through which humans enter information by using commonly used words (natural language). An example of this interface is a system in which processing progresses in a conversational mode through the use of **voice recognition**.

- **Non-verbal interface**

This is an interface through which information is entered by using expressions and movements rather than words. An example of this interface is a system in which information is entered through the movement of eyes and hands by using **image recognition** and **video recognition**. During image recognition and video recognition, **feature extraction**, in which a pattern expressing the features of the image is extracted, is performed.

For thinking of a human interface, the analysis of user operation with reference to the “Five aspects of the human machine interface” advocated by Professor Yamaoka of Wakayama University is also effective.

[Five aspects of the human machine interface]

(1) **Physical aspect**

This is the body-related aspect (e.g., the height of the operating panel and the posture during operation).

(2) **Information aspect**

This is the aspect concerning exchange of information (e.g., the ease-of-understanding and ease-of-viewing information). Examine the information

display method in view of humans performing **selective perception** in which humans select and accept only interesting information and useful information.

(3) **Temporal aspect**

This is the aspect concerning time (e.g., the work time and operation reaction time). **Learning functions** that enable rapid reproduction of the same operation are effective in shortening the work time.

(4) **Environmental aspect**

This is the aspect concerning environment (e.g., lighting, ambient temperature, and noise).

(5) **Organizational aspect**

This is the aspect concerning the operations (e.g., manuals).

(3) GUI (Graphical User Interface)

GUI is a technique of improving the usability and accessibility of the system and services through the use of graphics (i.e., pictures). The **Window** and **Icons** shown in Figure 2-1 are also an example of GUI tools.

- **Window**

It is possible to open several windows on the display and use each window as an independent single display.

- **Icon**

An icon indicates a resource and a command (**shortcut** to each application) through an easy-to-understand figure or pattern. Since an icon can be used simply through selection by moving the arrow on the screen with the help of the mouse, a beginner can easily operate the icon.

- **Radio button (Radio box)**

This is used to select only one item from the selection items (i.e., buttons).

- **Check box**

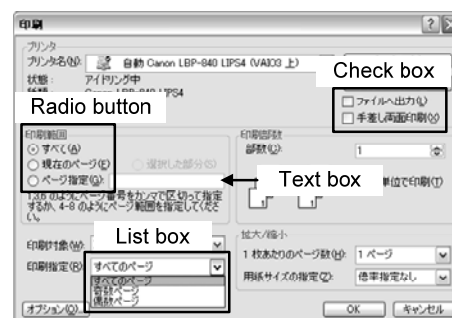
This is used to select several (i.e., more than one) items simultaneously from the selection items (i.e., buttons).

- **List box**

This is used to select only one item from the selection items (i.e., list).

- **Text box**

This is used to enter characters.



- **Menu bar**

This displays a row of items (i.e., menus) that can be selected by arranging them in line.

- **Pull-down menu**

When a menu is selected from the menu bar, a more detailed menu that is arranged vertically is displayed.

- **Pop-up menu**

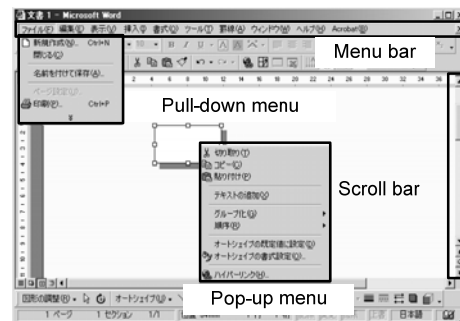
The menu is displayed as a pop-up by right-clicking at a specific location.

- **Scroll bar**

This is used to move the display area of the screen.

- **Progress bar**

This displays the progress status of the processing by using a graphical scale (i.e., length of the bar).



4 - 2 Interface Design

4-2-1 Screen Design and Form Design

(1) Screen design

Screen design refers to designing the content and layout to be displayed on the screen. During the interface design of the screen, the “Shneiderman’s Eight Golden Rules” are used very often.

[Shneiderman’s Eight Golden Rules]

- (1) Strive for consistency
- (2) Enable frequent users to use shortcuts
- (3) Offer informative feedback
- (4) Design dialog to yield closure
- (5) Offer simple error handling
- (6) Permit easy reversal of actions
- (7) Support internal locus of control
- (8) Reduce short-term memory load

In addition to the “Shneiderman’s Eight Golden Rules,” it is necessary to give consideration

to the screen layout and the information relationship displayed on the screen, in order to perform screen design.

[Points to be considered in screen design]

- Unify the display position of the titles and messages on the screen so as to standardize the screen layout (also standardize how to use the function keys.) ... (1)
- Standardize the display location of error messages, but do not standardize the content (help to understand the cause of an error and the appropriate coping technique.) ... (5)
- Arrange the reference items from left to right and from top to bottom. Also pay attention to the format of the original entry sheet, and arrange related items adjacent to each other.
- Enclose the items entered on the screen within blank box or brackets to emphasize the fact that these are input locations.
- Examine the use of the most appropriate GUI tools when data is entered on the screen. In addition to stepwise selection for beginners through the menu, enable direct selection using **shortcut keys** for skilled persons. ... (2)
- Display the progress status during processing through a progress bar so as to notify the user about the proper operation of the system. ... (4)
- Provide the **Undo** function to enable return to the previous status. ... (6)

Also, it is necessary to give consideration to **foolproof** in the design of the input screen and examine the methods for **input check** of the data. The following are typical methods of an input check:

- **Numeric check**
This ensures that data other than numeric values is not entered in numeric value items.
- **Limit check**
This makes sure that a numeric value falls within the specified range.
- **Format check**
This ensures that the data is in the specified format.
- **Duplication check**
This ensures that the same data is not duplicated.
- **Matching check**
This collates the data with the master file to ensure that the data is correct.
- **Balance check**
This ensures that the separately tabulated totals are consistent with one another.
- **Logical check** (Validity check)
This ensures that the data is logically correct.

- **Sequence check**

This ensures that the numbers are arranged in a sequence (and no number is missing).

- **Combination check**

This ensures that there is no contradiction in the combination of several related items that have been entered.

- **Check digit check**

This uses the **check character** to inspect the input data.

(2) Form design

Form design is used to design the items and layout of the forms (e.g., reports) to be generated from the system. The basic procedure of form design is as follows:

- 1) Comprehensive consideration

This process includes comprehensive consideration for the output purpose of the forms, output cycle (time period), output timing, distribution list, output volume (generated amount), and so on.

- 2) Decision concerning the output method and output medium

This process decides the most appropriate output method and output medium on the basis of the comprehensive consideration. In addition to paper, writing to CD and screen display are considered as the output media.

- 3) Creation of output items and form layout

This process considers the necessary output items and arranges them in an easy-to-understand manner on the forms. At this time, it registers the document model (e.g., borders and ruled lines) in the printer, and also considers the use of **form overlay** where data is superimposed and printed.

4-2-2 Code Design

In the **code design**, the data to be coded is selected from the data items used within a system and a code table is created for each target.

JP20CTV: 20-inch color TV made in Japan

FR16MTV: 16-inch monochrome TV made in France

- **Synthetic code**

It is a code in which several types of codes are combined together.

During code design, the following points should be considered.

- **Scope of the code**

When there is a possibility that the code may be used in a system other than the target system, do not make the code design in view of only the target system. Basically, systems should be able to use the codes commonly. If there is an existing code or standard code, it should be put to use.

- **Usage period of the code**

Depending on the usage period of codes, the number of necessary codes may be extremely large. For example, when the product code is designed on the basis of the sequence code and the current number of product types is 800, it is possible to code 1,000 types of products in three-digit numbers, which may not be a problem. However, if as many as 50 types of new products are developed every year, the codes will become insufficient after four years.

- **Maintenance of codes**

The maintenance method of codes is also taken into consideration in the code design stage. Particularly, the codes table is decided in consideration of the person in charge of creation and the creation timing so that no contradiction occurs in data.

- **Importance of the code**

In the case of a code of important data, it is necessary to add a **check digit** in order to prevent an input error by the operator.

[Check digit]

Modulus 10 is a typical method of calculating the check digit to be added to a code.

<An example of calculating the check digit>

- (1) Multiply each digit of the code “3612” with the weight (5, 4, 3, 2 in an order starting from the left of the code), and then add the result of each multiplication.

$$3 \times 5 + 6 \times 4 + 1 \times 3 + 2 \times 2 = 46$$

- (2) Divide the added result by 10, and use the remainder as the check digit.
(There is another method in which the remainder is further subtracted from 10.)

$$46 \div 10 = 4 \text{ Remainder } 6 \rightarrow \text{Code “36126”}$$

Note: If you enter this code erroneously as “36026”,

$$(3 \times 5 + 6 \times 4 + 0 \times 3 + 2 \times 2 - 6) \div 10 = 3 \text{ Remainder } 7$$

then the remainder does not become 0, which indicates an error in the code.

4-2-3 Human Interface Techniques

(1) Web design

Web design refers to designing of a web page or website used on WWW (World Wide Web), which is a mechanism of transmitting information on the Internet.

(a) **Web usability**

It is the “ease of use” of a web page or website. The column “Alertbox” by Jakob Nielsen, Ph.D. may be referenced with regard to web usability. In this regard, the following are cited as the five quality components of web usability:

- **Learnability:** How easy is it for users to accomplish basic tasks the first time they encounter the design?
- **Efficiency:** Once users have learned the design, how quickly can they perform tasks?
- **Memorability:** When users return to the design after a period of not using it, how easily can they reestablish proficiency?
- **Errors:** How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- **Satisfaction:** How pleasant is it to use the design?

(b) **Web accessibility**

It is the “ease of accessing” a web page or website. Concerning web accessibility, **W3C** (WWW Consortium), which performs standardization of WWW recommends the following **WCAG 2.0 (Web Content Accessibility Guidelines 2.0)**.

[**WCAG 2.0 (excerpt)**]

Principle 1: Perceivable (Information and user interface components must be presentable to the users in ways they can perceive.)

- | | |
|-------------------------|------------------------|
| 1.1 - Text alternatives | 1.2 - Time-based Media |
| 1.3 - Adaptable | 1.4 - Distinguishable |

Principle 2: Operable (User interface components and navigation must be operable.)

- | | |
|---------------------------|-------------------|
| 2.1 - Keyboard accessible | 2.2 - Enough time |
| 2.3 - Seizures | 2.4 - Navigable |

Principle 3: Understandable (Information and the operation of user interface must be understandable.)

- | | |
|------------------------|-------------------|
| 3.1 - Readable | 3.2 - Predictable |
| 3.3 - Input assistance | |

Principle 4: Robust (Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.)

- 4.1 - Robust

(c) **Style sheet**

It is a sheet that defines the document structure of a web page and decorative information (e.g., **frame**, font size, and line spacing) of a document. A web page is created by using HTML (HyperText Markup Language: Descriptive language for the web page). **CSS (Cascading Style Sheets)** are style sheets that can define the layout of a web page independent of HTML. By using a style sheet, it is possible to create a standardized website by only providing the information to be put up on the web page through a predetermined procedure. Although CSS is interpreted at the browser side, the operation may differ depending on the browser even for the same code. Therefore, a **cross browser** and **progressive enhancement** are used.

• **Frame**

It is a presentation technique that the web page to be displayed is divided into several parts. It also represents decorative designs such as ornamental lines and ruled lines that are added to each frame.

• **Cross browser**

It is a technique of eliminating differences in operation depending on browser specifications so that the website and web applications operate normally. By performing a different process in each browser, the execution of a different operation by the same code is avoided.

- **Progressive enhancement**

It is a concept of web design that the basic information is transmitted regardless of the PC environment, and an attempt is made to provide a rich experience to users who are using a high-function browser. It emphasizes the fact that information is transmitted accurately regardless of the browsing environment (i.e., browser), on the basis of a three-layer structure including “a logical structure that the text content is transmitted with authenticity,” “visual design control,” and “site behavior control.”

(d) **Navigation**

Navigation in a website refers to a mechanism that enables fast and accurate access to the desired information (i.e., web page) on the site. The typical navigation used in a website includes the following:

- **Site map**

It is a mechanism that enables fast search of the desired information by hierarchically displaying the labels added for understanding the information of web pages.

- **In-site search**

It is a function that displays the relevant web page or the location within a web page when the search keyword is entered. It includes the full text search that targets the entire text and the index search that searches for the registered keyword.

(2) **Human-centered design**

Human-centered design is an approach to interactive system development that focuses specifically on making systems usable. It is established as **ISO 13407** and translated as **JIS Z 8530** (In 2010, ISO 13407 was integrated into ISO 9241 as ISO 9241-210.)

In ISO 13407 (JIS Z 8530), four characteristics are provided as the principles of human-centered design.

[Principles of human-centered design]

- a) The active involvement of users and a clear understanding of user and task requirements
- b) An appropriate allocation of function between users and technology
- c) Iteration of design solutions
- d) Multi-disciplinary design

Moreover, four processes are cited as human-centered design activities based on these principles. As shown in Figure 2-13, it is desirable that these human-centered design activities (i.e., processes) be started from an early stage of information processing system development and executed repeatedly until the requirements are fulfilled.

[Human-centered design activities]

- a) Understand and specify the context of use
- b) Specify the requirements of users and organizations
- c) Produce design solutions
- d) Evaluate designs against requirements

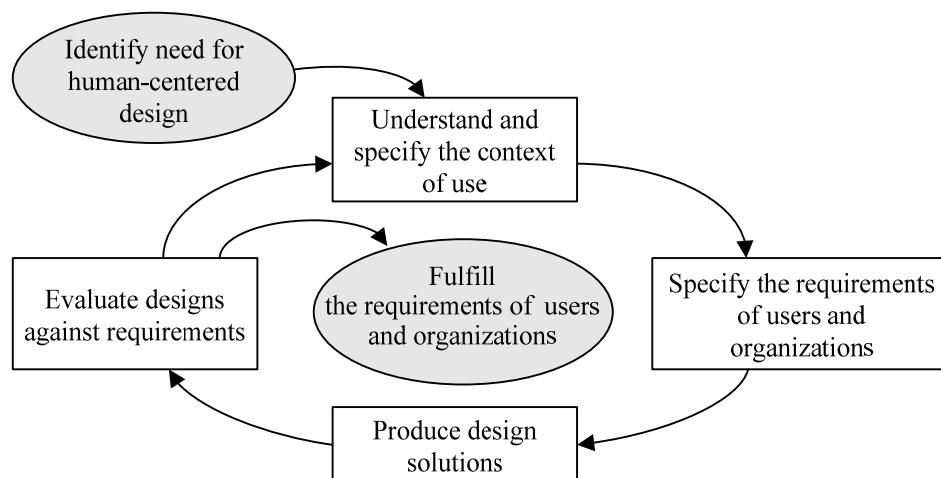


Figure 2-13 Interdependence of human-centered design activities

(3) Universal design

A **universal design** refers to a design that provides comfortable and easy-to-use environments and services to all persons regardless of the existence of differences in the age, culture, ability, and disabilities. The “Seven Principles of Universal Design” proposed by Dr. Ron Mace are frequently used to achieve universal designs.

[Seven principles of universal design]

- (1) Equitable use: The design is useful and marketable to people with diverse abilities.
- (2) Flexibility in use: The design accommodates a wide range of individual preferences and abilities.
- (3) Simple and intuitive use: Use of the design is easy to understand, regardless of the user's experience, knowledge, language skills, or current concentration level.
- (4) Perceptible information: The design communicates necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities.
- (5) Tolerance for error: The design minimizes hazards and the adverse consequences of accidental or unintended actions.
- (6) Low physical effort: The design can be used efficiently and comfortably and with a minimum of fatigue.
- (7) Size and space for approach and use: Appropriate size and space is provided for approach, reach, manipulation, and use, regardless of user's body size, posture, or mobility.

Moreover, there are some more policies of universal design. **WCAG 1.0** (1995) and **WCAG 2.0** (2008) are compiled as guidelines for enabling access (i.e., setting in a high accessibility status) to web content particularly for persons with disabilities by an internal organization of W3C, **WAI (Web Accessibility Initiative)**. **WAI-ARIA (Web Accessibility Initiative-Accessible Rich Internet Applications)** is also provided as specifications for accessibility of dynamic web content using video and sound.

The universal design is used to pursue not just ease-of-use (i.e., usability) but also ease-of-access (i.e., accessibility) by any person. For example, mechanisms are prepared for changing the font size and color for visually impaired people and enabling voice input for persons who cannot use the mouse because of being unable to use their hands.

Although the universal design is meant for all persons, there is another concept called **information barrier free** that targets elderly persons and persons with disabilities. While universal design is a concept that an obstacle-free environment is designed or provided right from the beginning, information barrier free is a concept that an information communication environment that does not have (is free of) any obstacles is provided by eliminating the obstructions (i.e., barriers) for elderly persons or persons with disabilities. The concept of information barrier free is stipulated in **JIS X 8341** (Guidelines for elderly persons and persons with disabilities — Information and communications equipment, software, and services —).

4-2-4 Usability Evaluation

Usability evaluation refers to evaluation of the usability (i.e., ease-of-use) of a system and website that is created (or developed) on the basis of the human interface design.

(1) Qualitative evaluation

The qualitative evaluation in usability evaluation refers to the technical evaluation of the system and website interface. It is mainly implemented for the purpose of evaluating the “effectiveness” and “efficiency” of usability.

[Methods of qualitative evaluation]

- **Usability test** (user test)

It is a method that a usability engineer assigns a task to a subject (i.e., user) and detects the problems of the user interface from the actions and speech of the subject during the course of the execution of the task. When a website is evaluated, a **log data analysis method** (**access log analysis**) that the access log (e.g., record of web page access information) is studied in order to analyze how the web page has transitioned is also used in combination.

- **Heuristic evaluation**

It is a method that members, such as usability engineers and designers with abundant technical knowledge and experience, are gathered, and the user interface is evaluated on the basis of the existing empirical rules (i.e., heuristics). In this way, the problems are brought to light. It is one of the **expert reviews** (i.e., review by an expert) in which the opinion of the user is not reflected.

(2) Quantitative evaluation

The quantitative evaluation in usability evaluation refers to the method that the system and website are compared with others and evaluated relatively. It is mainly implemented for the purpose of evaluating the “satisfaction level of the user.”

[Method of quantitative evaluation]

- **Questionnaire survey/Interview method**

It is a method that the opinion of users who are using a system and website is collected through a questionnaire for several unspecified persons and face-to-face interviews. This makes it easier to collect information about dissatisfaction of the user that is compared with other systems and websites.

5 Multimedia

In modern society, information processing systems are being used in various fields. Among these, the system that has frequently been used recently is the multimedia system. The multimedia system is a system that uses multimedia data such as moving images, still images, sound, and text by integrating it on a computer and can be represented in various forms.

5 - 1 Multimedia Technology

5-1-1 Multimedia

Multimedia refers to **interactive** (i.e., bidirectional) media. Information, such as characters, images, pictures, and sound, is digitalized (i.e., encoded) to integrate information media.

Different types of information available on the Internet that we use are registered on a Web server. (This is known as **web content**.) A large part of the web content is multimedia content in which characters, images, and sound are combined together. The software used for saving and managing the web content, and constructing a website is called **CMS (Contents Management System)**. A typical CMS is **Wiki** by which the editing of a web page is performed easily from the browser. Wikipedia, which is called the encyclopedia of the Internet, is a typical usage example of Wiki.

Links (i.e., information for accessing related pages) are embedded in a large part of the web content, and different types of information can be searched for by following the links. Web content that is correlated through a link including text information is called **hypertext**, and web content (i.e., multimedia content) that is correlated through a link including images is called **hypermedia**.

In the authoring environment for creating the multimedia content, the software called **multimedia authoring tool** is mostly used. This software enables easy integration of material (i.e., files) such as documents, image/video, and sound through the mouse operation. At this time, **PDF (Portable Document Format)** may be used as one of the formats of document files. PDF is a format of document files developed by Adobe Systems that enables print images to be displayed in almost the same manner even in different applications by converting them into PDF files as they are, without any changes. When a print image is converted to a PDF file, the file size becomes smaller than the original data file, but editing cannot be performed in the original application. However, the characteristics of a PDF file also result in advantages such as prevention of data change and secondary use.

Because of its property, the data files of multimedia have a larger size than the data files of only text information. Therefore, on the Internet, the technique called **streaming** is used for

playing back at the download of a data file of video and music.

5-1-2 Sound Processing

Sound processing is a process that analog sound is handled as digital data. The techniques of **A/D conversion** by which an analog signal is encoded into a digital signal include **PCM (Pulse Code Modulation)**. The procedure of encoding (i.e., A/D conversion) by PCM is as shown below.

[Encoding procedure by PCM]

1) Sampling

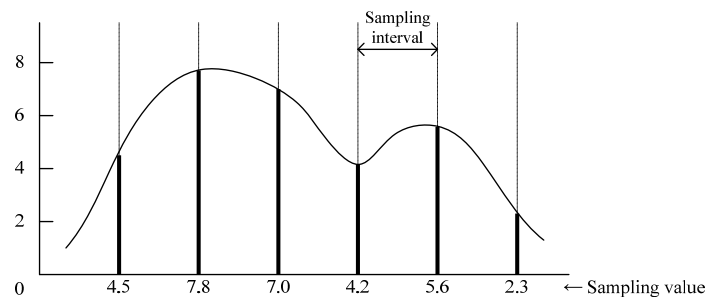
The analog signals to be encoded are sampled at regular intervals. At this time, the **sampling frequency** indicating the sampling interval is determined on the basis of Shannon's sampling theorem.

- **Shannon's sampling theorem:**

When the highest frequency of the target analog signal is f , the original analog signal can be recovered if sampling is performed at a frequency of $2f$ or above.

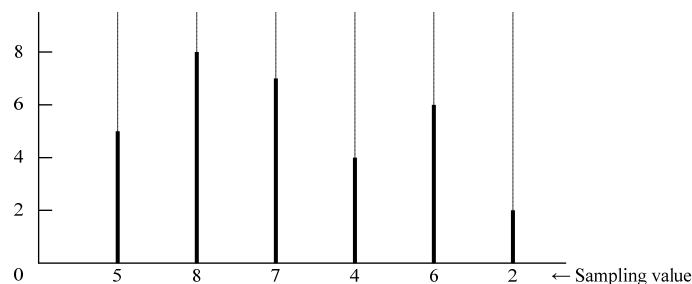
Note: Frequency is the number of vibrations in 1 second.

The unit is Hertz (Hz).



2) Quantization

The sampled sampling values are rounded to the nearest integer numbers.



3) Encoding

A quantized integer number is encoded by representing it as a binary number. (The number of bits used as a code is called the **quantization bit rate**.)

Quantized value	5	8	7	4	6	2
	↓	↓	↓	↓	↓	↓
Encoded value	0101	1000	0111	0100	0110	0010

Example: When sound data that has the highest frequency of 2,000 Hz is encoded by a PCM method of 8-bit quantization bit rate, how many bits of digital data is contained in the entire sound data encoded in 1 second? Here, the sampling frequency is the minimum required frequency determined by Shannon's sampling theorem.

(1) Sampling frequency

= Highest frequency of analog signal (i.e., sound data) $\times 2$
 = 2,000 Hz $\times 2$
 = 4,000 Hz ... Sampling is performed 4,000 times in 1 second.

(2) Number of bits obtained by encoding the sound data of 1 second

= Sampling frequency of 1 second \times Quantization bit rate
 = 4,000 times/second \times 8 bits/one time
 = 32,000 bits/second

As clear from the above example, the amount of digital data after encoding is determined on the basis of the sampling frequency (i.e., sampling interval) and quantization bit rate.

Moreover, **D/A conversion** that a digital signal (i.e., code) is restored as an analog signal (i.e., sound data) can be implemented by performing the encoding procedure in the reverse manner. At this time, in order to be able to perform restoration at a quality level that is closest to the original analog signal, either the sampling frequency is increased (sampling interval is shortened) and the number of sampling values is increased, or the quantization bit rate is increased to perform detailed staging of sampling values.

The typical formats (i.e., standards) of a sound file are as follows:

- **MIDI (Musical Instruments Digital Interface)**

It is a file format that an electronic musical instrument and PC are connected and music data is exchanged. Not sound as such, but the rendition information for playing back the sound is handled.

- **MP3 (MPEG1 Audio Layer3)**

It is a high-quality sound compression and decompression format using the sound

technology of MPEG (moving images compression and decompression format) standardized by ISO. It is also used in Internet music distribution and portable players.

- **WAV (RIFF Waveform Audio Format)**

It is a sound data format that is mainly used in Windows systems. Various formats of data, such as ADPCM and WMA, can be stored in the container format.

- **ADPCM (Adaptive Differential Pulse Code Modulation):**

It is a sound signal compression method in which the PCM technique is applied.

- **WMA (Windows Media Audio):**

It is a streaming technique of music distribution that was originally developed by Microsoft. Recently, it has been used as a sound compression technique.

5-1-3 Still Image Processing

Still image processing is the process of handling images that do not move. The basic mechanisms of image representation include the color representation method and image quality.

- **Color representation method**

This method uses the **three primary colors of light (RGB)** and the **three primary colors of color (CMY)**. Generally, RGB is used in the color representation of the display, and CMY is used in the color representation of the printer. However, it is common to use the four colors of CMYK by adding black (Key) to the three primary colors of color.

- **Image quality**

This is represented by **resolution** and **gradation**. Resolution represents the density of the **pixels** (i.e., dots) configuring an image in the number (**dpi (dots per inch)**) of pixels (i.e., dots). Gradation represents the stage of the color that can be represented in each pixel.

The typical formats (i.e., standards) of a still image file are as follows:

- **BMP (Bit MaP)**

It is a format that represents the characters and images as a set of dots. A flaw that exists is that the amount of information generally increases more than in other formats.

- **TIFF (Tagged Image File Format)**

It is a format that enables handling of bit map images of various formats through the addition of tags (i.e., identifiers).

- **JPEG (Joint Photographic Experts Group)**

It is an international standard that supports full color and is defined by JPEG, an institution of ISO. It is a still image compression and decompression format that enables adjustment of the image quality and data amount by regulating the amount of compression. Because of an extremely high compression rate, it is used more frequently than other still image file formats. The compression formats of JPEG include **lossless compression** in which the original data is completely recovered and **lossy compression** in which the original data is not completely recovered. However, the lossy compression format is generally used.

- **GIF (Graphic Interchange Format)**

It is a still image compression and decompression format that supports 256 colors. Although it is a lossless compression format, the amount of data is comparatively less, and it is also used in image processing on the WWW.

- **PNG (Portable Network Graphics)**

It is a still image compression and decompression format that supports full color and is an expansion of GIF. It is a lossless compression format, and the amount of data is more than JPEG.

- **Exif (Exchangeable Image File Format)**

It is a format that additional information (i.e., meta data), such as the photography time, is attached and saved in the image data taken by a digital camera.

5-1-4 Moving Image Processing

Moving image processing is the process of handling images that move. During the moving image processing of the computer, a series of still images is displayed in sequence. Each image is handled as a **frame**. At this time, the number of frames displayed in 1 second is the **frame rate**, which is represented in **fps (frames per second)**.

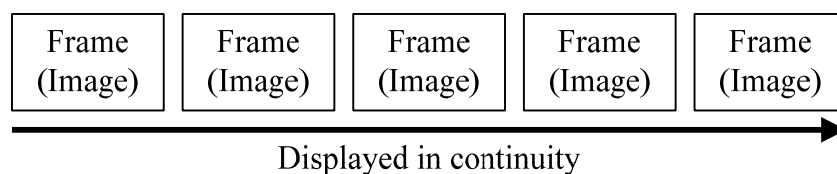


Figure 2-14 Image of moving image processing

The typical formats (i.e., standards) of a moving image file are as follows:

- **MPEG (Moving Picture Experts Group)**

It is an international standard of moving image data that is defined by MPEG, an institution of ISO. It is a moving image compression and decompression format for which several standards are defined according to the quality of the image to be compressed.

MPEG-1: A compression and decompression technique of images that have a quality of video level

MPEG-2: A compression and decompression technique of television pictures and high-vision pictures

MPEG-4: A compression and decompression technique assuming the use of cellular communication

MPEG-7: A notational system of meta data assuming the implementation of a high-speed search engine

- **QuickTime**

It is a moving image compression and decompression format used in Apple's "QuickTime." It makes use of the **motion JPEG**) method, where JPEG images are played back continuously, and can also be used in Windows.

- **VRML (Virtual Reality Modeling Language)**

It is a file format for displaying 3D graphic data as pictures.

- **AVI (Audio Video Interleaving)**

It is a moving image data format that is mainly used in Windows systems. It uses **RIFF** (Resource Interchange File Format), which is a format of multimedia files.

[CSV (Comma Separated Value) format]

CSV format is a file format used in spreadsheet software and database software. It is a format in which the data of each field is separated by a comma, and each record is separated by a linefeed code.

5-1-5 Compression and Decompression of Information

Because of their nature, the data files of multimedia have a larger size (amount of data) than the data files of only text information. If such a large data file is used as it is, a large recording area is required for storing the data file, and the load also increases when the file is exchanged across the network. Thus, in order to improve the efficiency of data storage and reduce the network load, the size (i.e., amount of data) of information is reduced in accordance with fixed rules (i.e., procedures). The process of reducing the size of information is called **compression** and the process of restoring the compressed information is called **decompression**.

The concept (i.e., format) of compression is broadly classified into the following two types of compression:

- **Lossless compression**

It is a compression technique that enables a complete return to the original data from the compressed data.

- **Lossy compression**

It is a compression technique that does not enable a complete return to the original data from the compressed data. In the case of images, since humans hardly ever notice the differences even when the restoration process is more or less incomplete, it does not result in a big problem. In comparison with lossless compression, the **compression rate**, which indicates the compression effect of the amount of data, can be increased; that is, the amount of data can be reduced.

The typical information compression and decompression formats (i.e., standards) are as follows:

- **JPEG (Joint Photographic Experts Group)**

It is an international standard of a still image compression and decompression format that supports full color.

- **MPEG (Moving Picture Experts Group)**

It is an international standard of a moving image compression and decompression format.

- **ZIP**

It is a file compression format supported by several types of free software in addition to software products such as WinZip. It is used not only for compressing a single file but also for **archiving**, where several files are collectively compressed into one file. At an international level, it is a de facto standard (i.e., industry standard) file compression format.

- **GZIP (GNU ZIP)**

It is a file compression format that is mainly used in UNIX. Although its name includes ZIP, it is not compatible with ZIP, and there is no archive function either.

- **LZH**

It is a file compression format supported by the free software LHA. It is mainly used as a file compression format within Japan.

- **MR (Modified Read) / MMR (Modified Modified Read)**

It is a compression and decompression format (i.e., encoding format) that is mainly used for facsimile.

(1) CG (Computer Graphics)

CG (Computer Graphics) is a technique of drawing images (i.e., graphics) and performing editing by using a computer.

[Software for CG]

- **Painting software**

It is graphics software by which images are drawn in units of dots. The images thus created are called raster graphics. Since images are created by drawing lines with dots on the path traced by the mouse, this software can be used easily even by beginners.

- **Drawing software**

It is graphics software by which images are drawn by calculating the direction and length (i.e., vector data) of lines. The images thus created are called vector graphics. The notches (jaggies) of the lines are not noticeable even when the drawn image is expanded.

[CG techniques]

- **Anti-aliasing**

It is a technique by which the jaggies that occur in slanting lines and curved lines are made inconspicuous.

- **Texture mapping**

It is a technique by which a texture is presented by pasting an image or a pattern onto the surface of a modeled object.

- **Blending**

It is a technique by which the information on the degree of transparency is superimposed to represent a translucent image.

- **Ray-tracing**

It is a technique by which a ray of light that has reached the eye point is traced in the reverse manner to perform rendering.

- **Clipping**

It is a technique by which only the portion to be displayed is cut out from the entire image.

- **Shading**

It is a technique of forming shadows on the surface of an object in order to provide the appearance of solidity.

- **Morphing**

It is a technique of changing the nodal points to smoothly change from one shape to another.

- **Rendering**

It is a technique of displaying the data of an object as a two-dimensional image or picture.

- **Polygon**

It is a polygon that is used to approximate a closed solid or a curved surface through three-dimensional CG.

- **Key frame method**

It is a method for generating animation by interpolation between key frames.

(2) Three-dimensional picture (3D)

A **three-dimensional picture (3D)** is a picture (and the technique of creating pictures) that has depth and the appearance of solidity. It is created by three-dimensional computer graphics (**3DCG**) and **motion capture**, which uses a technique of attaching a sensor to each joint of a human body, feeding the information, such as coordinates and acceleration, detected by the sensors during movement to a computer, and then converting it into digital data. Moreover, in combination with **virtual surround** (e.g., a technique of virtually playing back 5.1-ch sound through two speakers or regular headphones), realistic content can be created.

(3) VR (Virtual Reality) / AR (Augmented Reality)

VR (Virtual Reality) refers to experiencing the virtual world that is generated by using CG as an actually existing world. On the other hand, **AR (Augmented Reality)** refers to extension of the real world by combining the real world and VR. In AR, information is added to actually existing things by VR and also emphasized.

(4) Other examples of application

A **multimedia system** that uses multimedia data, such as moving images, still images, sound, and text, by integrating it on a computer has several application examples.

- **Internet broadcasting**

It refers to broadcasting through the medium of the Internet.

- **Non-linear image editing system**

It is a method of editing images as digital data on a computer.

- **CAD (Computer Aided Design)**

It is a computer system that supports the design of a product by using the CG techniques.

- **Simulator**

It is a program or mechanism for a simulation approach to complex scientific phenomena that occur in reality. The prediction of the course of a typhoon and the wind tunnel experiment determining how the surrounding air flows when a car is running can be reproduced by the computer.

- **Video game**

It is a game that can be played on the computer.

- **Video on demand**

It is a service that distributes video according to the request of the user.

- **Virtual mall (Cyber mall)**

It is a virtual shopping mall constructed on the Internet. Although there is no shopping mall, it is possible to actually buy real products.

Chapter 2 Exercises

Q1

In a client/server system, which of the following is the most appropriate function to be processed at the server side?

- a) Process of displaying the output data
- b) Process of updating the database
- c) Checking the format of entered data
- d) Process of displaying the pull-down menu

Q2

Which of the following is the computer system where one computer is in the standby state while the other computer operates normally?

- a) Dual system
- b) Duplex system
- c) Multiprocessor system
- d) Load sharing system

Q3

Which of the following is a component of a fault tolerant system?

- a) RAID0
- b) Duplexing of hard disk
- c) Schedule backup
- d) Data encryption

Q4

Which of the following is an appropriate description of the performance evaluation of a system?

- a) In OLTP (On-Line Transaction Processing), the MIPS value is used in the performance evaluation of the system.
- b) The response time and turnaround time are performance indicators from the viewpoint of the system operations manager.
- c) If the utilization rate of system resources becomes high, the response time also

generally improves accordingly.

- d) The number of transactions or jobs that can be processed within a unit time is important for evaluation of the system performance.

Q5

In a processor that has a clock time of 3 nanoseconds, when the number of clocks that is necessary for the execution of an instruction and the occurrence rate of the instruction are shown in the table below, what is the approximate performance (in MIPS) of this processor?

Type of instruction	Number of clocks necessary for the execution of an instruction	Occurrence rate
Register-to-register operation	4	40%
Register to/from memory operation	8	50%
Unconditional branch	10	10%

- a) 5 b) 30 c) 50 d) 100

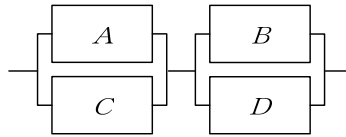
Q6

RASIS is a set of indexes or indicators for operating a system in a secure and stable manner. Among the indexes or indicators of RASIS, which of the following is the index that uses the probability of operation of the system?

- a) Availability b) Integrity
c) Reliability d) Security

Q7

In the system configuration as shown in the figure below, which of the following is the closest availability of the overall system? Here, *A*, *B*, *C*, and *D* indicate the devices, and the availability is 0.9 for *A* and *C* and 0.8 for *B* and *D*.



- a) 0.72 b) 0.92 c) 0.93 d) 0.95

Q8

Which of the following is the GUI tool that is used when only one item is selected from the multiple items that are mutually exclusive?

- a) Scroll bar b) Check box
c) Progress bar d) Radio button

Q9

Which of the following is an appropriate purpose of appending a check digit to a customer code?

- a) In order to detect an input error of the customer code
b) In order to arrange customers in order of their acquisition when a customer list is created
c) In order to enable classification of customers in groups on the basis of areas
d) In order to enable analogical inference of a specific customer

Q10

When a browser-based service of employee information is started, it is considered to publish a full-color face photograph of employees. Which of the following is the most appropriate image compression format for reducing the load on the in-house network?

- a) GIF b) JPEG c) MIDI d) MPEG

Which of the following is the most appropriate explanation of virtual reality?

- a) It refers to representing a world that is created within a computer as if it were a real world, by using CG techniques.
- b) It refers to displaying an image in mosaic form first and then gradually displaying clearer for the purpose of improving the GUI, instead of sequentially displaying an image from the top.
- c) It refers to a simulation approach to the wind tunnel experiment that is used for designing a car or an airplane through the use of a computer, and for testing whether or not the expected results are obtained.
- d) It refers to supporting the design of a product on a computer by using CG techniques.



Chapter 3

Software



1 Classification of Software

Software is a word that was created to contrast with hardware, and it occupies a position between a person and a computer so that the computer can be used efficiently. The broad definition of software includes documents including manuals for the use of computers, but software generally refers to a program (a collection of instructions).

1 - 1 Systematic Classification of Software

In terms of systematic classification, software can be broadly classified as **system software** and **application software**. System software is a group of programs that enable effective use of the functions of the devices (i.e., hardware) that constitute a computer. Application software is a group of programs that provide functions corresponding to the aims of the user. In other words, it can be said that system software focuses more on hardware, and application software focuses more on people.

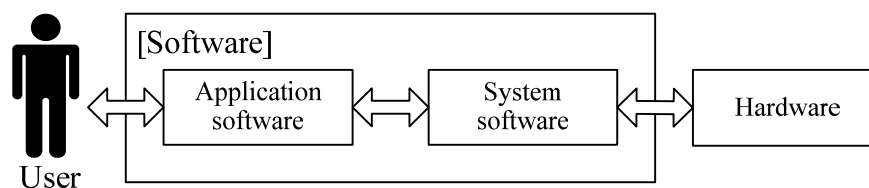


Figure 3-1 Software function

The fundamental software that manages hardware resources and controls the computer is called the **OS (Operating System)**. Depending on the interpretation of “control the computer,” an OS is classified as either a “broadly defined OS” or a “narrowly defined OS.”

Application software		
System software (broadly defined OS)	Middleware	Language processor
	Service program	
	Control program	
	(narrowly defined OS)	
Hardware		

Figure 3-2 Systematic classification of software

Recently, this systematic classification has become very vague. Figure 3-2 should be understood as just one example of the systematic classification of software.

(1) Control program

A **control program** is software that efficiently manages and uses resources such as hardware, and provides an efficient operating environment. This is also called a narrowly defined OS. The core of control programs is a program called a **kernel**. Normally, software runs in **user mode**, which has restrictions on CPU usage. However, when an **interrupt** (e.g., a process request from application software) occurs and the kernel takes control, the mode changes to **kernel mode** (**supervisor mode**) which has no restrictions on CPU usage. Depending on the function, kernels are classified as either of the following:

- **Microkernel**

This is a kernel that has only the absolute minimum number of functions, such as memory management and process management.

- **Monolithic kernel**

This is a kernel with many functions, such as input/output management and file management.

When a computer is powered on, an **IPL (Initial Program Loader)** runs to launch the control program (OS). This series of processes is called a **boot**, and the program used in the boot is called a **bootstrap** (or **bootloader**). Since it is necessary to save the bootstrap even when the computer is powered off, in early computers it was stored in ROM. However, these days a **flash bootloader** that is stored in flash memory is often used. Furthermore, other boot methods, such as **multiboot** which enables the user to select a control program (i.e., OS) to launch from multiple candidates, and **network boot** which enables a boot to be performed over a network, are also used.

(2) Service programs (utility programs)

A **service program** (i.e., **utility program**) is software that provides standard functions to support users when they use a computer. Typical service programs include a data management utility (i.e., **archiver**) that provide an **archive** function to combine multiple files into one file and restore it to their original multiple files, and a disk management utility (e.g., disk defragmentation of Windows) that provide functions for the effective use of a hard disk.

(3) Language processor

A **language processor** is software that is used to translate a program that is created in a language (i.e., **programming language**) which people can easily understand into a language (i.e. **machine language**) that can be understood by computers. Typical language processors include **compilers** that translate the original program (i.e., **source program**) into a machine language program (i.e., **object program**) in a batch. In order to run a program, it is necessary to build up an executable program (i.e., **load module**) by using a **linkage editor**, which is a service program.

(4) Middleware

Middleware is software that is located between the OS and application software. It provides basic functions that are common to a variety of purposes. By providing basic functions with a unified interface, it facilitates the development of application software that does not rely on an OS nor hardware. It is difficult to accurately distinguish middleware because some is embedded as part of the OS and some is purchased additionally by the user. Generally, software that is classified as middleware includes the types below.

- **Linking software between application programs**

This is software that enables data compatibility and other functions between application programs.

- **DBMS (DataBase Management System)**

This is software that manages a database that handles data in an integrated way.

- **Communication management system**

This is software that manages communication between computers or servers

- **Software development tool**

This is software that supports software development with programming languages and other tools.

- **Operations management tool**

This is software that supports or manages the operations of a computer (or server).

- **TP (Transaction Processing) monitor**

This is software that manages transaction processing.

In a broad sense, (1) to (5) below are also classified as middleware.

(1) Shell

This is software that interprets user operations with a **command interpreter** and conveys them to (or gives instructions to) the OS. It is sometimes classified as an OS function.

- **Redirect function**

This is a function of shell that enables a target of standard input (keyboard) or standard output (display) to be changed to a file. In the case of changing standard output to a file, it is possible to specify whether to overwrite or append.

(2) API (Application Program Interface)

This is a mechanism (i.e., interface) for using the features provided by an OS (e.g., commands, functions) from an application program. By using the same OS and APIs, program compatibility can be increased, and the required number of person-hours can be reduced at the time of porting the program to another platform.

- **Web API**

This is a mechanism that enables the use of functions that are useful in the creation of web content. It is usually provided on the Internet (Web) by website operators.

(3) Library

This is a file that groups programs with highly versatile and specific functions in a reusable format. (It sometimes refers to mechanisms that manage and provide files.) Normally, a library is incorporated into another program for use and cannot be executed as a standalone program.

- **Source library**

This is a library in source code (i.e., source program) format.

- **Object library**

This is a library in object code (i.e., object program) format. It sometimes refers to a **class library**, which is a collection of classes (i.e., integrated data and procedures) that are used in object orientation.

- **Load library**

This is a library that is incorporated when a load module is generated. It normally refers to libraries that are embedded using a linkage editor.

- **DLL (Dynamic Link Library)**

This is a library that is incorporated by the OS during the time of execution. It is sometimes classified as a type of load library.

(4) Componentware

This is software that has a specific function and is used as a component of program during program development. The development of an application by combining componentware is called component-based programming.

- **JavaBeans**

This is a specification for handling programs, as a component, which are developed with the programming language Java.

- **ActiveX**

This is a general name for Internet-related components and a component technology that are provided by Microsoft.

- **CORBA (COMmon Request Broker Architecture)**

This is a standard specification that enables the exchange of messages between objects, and it includes a specification called CCM (CORBA Component Model) concerning components.

(5) Development framework

This is software that provides generally-used functions that are often required during application development, and works as a base for applications. It can be called a general framework for standardizing and streamlining application development.

(5) Application software

Application software is software that provides a wide range of functions that correspond to the aims of the user. Other than system software that focuses on hardware, most software is classified as application software.

Application software is broadly classified into two groups depending on the aims of usage.

- **Individual application software**

This is software that is custom made specifically for the resolution of problems that are unique to an individual or a company. Since this software is custom made, the functions provided fully satisfy the individual or the company, but in comparison with other application software, costs build up and the period required for development is long. There are also problems such as uncertainties over the reliability of the completed software.

- **Common application software**

This is standard software that deals with common business operations that are performed by a company or an individual (e.g., payroll calculation, account ledger management). If software that has been developed already is purchased, the cost is

lower and the reliability is high, but dissatisfaction can occur because it does not necessarily match the requirements of each individual operation nor the purpose of use. Hence, most application software is equipped with functions for **customization** that allows partial changes to be made and saved according to the user's environment and demands, and so on.

Typical application software includes those described below.

(i) **Word processing software**

This is software that is used to create documents. It allows text editing and diagram/graph insertion. Fonts that are used in word processing software and such other software include: **fixed-width fonts** where the characters have the same width, **proportional fonts** where the width of the characters is different, **bitmap fonts** where characters are represented with a combination of dots, and **outline fonts** where the outline of a character is represented with the information of a curved-line function. Outline fonts are suitable for enlarged display because bitmap fonts appear jagged when enlarged.

- | | |
|--------------------------------|---|
| • Tab function: | This is a function for formatting a document. It moves the cursor to a specified position on the same line in order to give a consistent format for print or display. |
| • Macro function: | This is a function that allows a procedure to be defined in advance and called for use when required. |
| • Merge print function: | This is a function that prints many copies of the same document but changes part of the text by inserting data from another file. |
| • Line breaking rule function: | This is a function that prevents certain symbols such as a period from being placed at the start of a line, and other symbols, such as “¥” (i.e., currency symbol) or “(” (i.e., left parenthesis), from being placed at the end of a line. |

(ii) **Spreadsheet software**

This is software that creates tables or performs calculations between cells by entering data or an expression into cells that are arranged in rows and columns on a worksheet.

- **Plug-in software** (extensions, add-ins)

Plug-in software does not work as a standalone program but can be integrated into applications, such as spreadsheet software, to provide additional functions. It can be added or removed as required. Typical plug-in software includes Flash Player, which plays back multimedia data. Many items of plug-in software are available for free from the Internet or other sources.

(iii) **Database software**

This is software for creating, using, and managing a database. It also allows to create simple data processing programs by using a dedicated language.

(iv) **DTP (DeskTop Publishing) software** (digital publishing system)

This is software for creating commercial printed materials. It has many editing functions such as font adjustment and layout. In order to perform layout work or control a high-precision printer, it has a built-in **PDL (Page Description Language)** function. (Typical PDLs include **PostScript** and **ESC/Page**.) Furthermore, a **WYSIWYG (What You See Is What You Get)** function prints what the screen displays as it is. It is implemented in a wide variety of application software including DTP software.

(v) **Graphics software**

This is software that is used to create and handle graphics and images on a computer. It includes **painting software** and **drawing software**.

(vi) **Presentation software**

This is software that is used to create materials for presentations and is used in the presentation itself. In some cases, it may have animation functions, such as movies, and sound functions including sound effects. For presentations, it is better to use a **projector** that has a higher number of **lumens**, which is a unit that represents the brightness of a light source.

(vii) **Groupware**

This is software that supports the joint activities of an organization with computers. Typical functions include e-mail, digital conference, schedule management, and **workflow management** (a function to specify and manage business processing procedures and the flow of documents and information.)

(6) OS (Operating System)

Below is a description of typical types of **OSs** (Operating Systems).

- **MVS (Multiple Virtual Storage)**

This is a typical **general purpose computer OS** that was developed by IBM. It has been remade and improved to become its successor z/OS.

- Multi-user, multitasking function
- Implements large virtual address space that supports any file organization

- **Windows**

This is an **OS for PCs** that was developed by Microsoft. It is installed on most PCs around the world. It is a series, and a new version is frequently released.

- Rich GUI function based on a wide variety of icons
- Multitask function using a window system
- Plug and play function that allows peripherals to be used by simply connecting them
- Rich network function to support access to the Internet

- **UNIX**

This was originally an OS that was developed for minicomputers by AT&T Bell Laboratories, but it has now been developed into an **OS for PCs**. Since the specification has been widely released, it is easy for vendors or developers to include it in their own products, and many improved versions are widely used.

- Composed of the kernel and the shell
- Multiuser, multitask (i.e., multiprocessing) function
- Network function that enables simple implementation of distributed processing
- Input/output devices can be handled in the same way as files

- **Linux**

This is an **OS for PCs** that builds on and improves the UNIX approach. It is a typical example of an **open OS** (i.e., OS of open source software).

- The kernel that is the core of the OS is released and distributed for free
- Multiuser, multitask (i.e., multiprocessing) function
- System management with **single-user mode**

- **Mac OS**

This is an **OS for PCs** that Apple developed for its own PCs (Macintosh). Some versions for sale include UNIX technology in their kernel.

- Sophisticated operability with GUI
- Well-developed graphics / DTP software

- **Windows CE**

This is a **real-time OS** that was developed by Microsoft and is embedded in household appliances, and so on. The real-time OS is used for **hard real-time systems**, which

require a process corresponding to an event to be finished within a set time, and such other systems.

1 - 2 Classification by Software License

A **software license** is a document that describes the items that must be obeyed when the software is used (e.g., permission/prohibition/conditions for usage, modification, redistribution). Generally, it is included in the **usage licensing agreement** that is made between the right holder of the software and the user.

Software that is classified from this perspective includes those described below.

- **Packaged software**

This is software that is commercially available to the general public. It is sometimes called application software. Restrictions are placed on its use.

- **Freeware (free software)**

This is software that is distributed for free of charge over networks such as the Internet. Since the developer holds the copyright (i.e., the intellectual property rights relating to a created work), restrictions are placed on modification, redistribution, and so on.

- **Shareware**

This is software that can be used free of charge for a trial period. However if the user wishes to continue using the software after this period, a usage fee must be paid. It is the same as freeware except for the fact that a fee must be paid.

- **PDS (Public Domain Software)**

This is originally referred to software that was researched and developed by a public institution or other organizations for which the copyright is waived and can be used free of charge. However, in recent years, it sometimes refers to software for which the copyright has not been waived, and the classification has become vague.

Furthermore, there exists a classification called **OSS (Open Source Software (OSS))** which is based on the **OSD (The Open Source Definition)** that was released by a nonprofit organization **OSI (Open Source Initiative)**. A typical example of OSS is **Linux** whose kernel (**Linux kernel**), the core of the OS, has been published and distributed.

[Open source requirements specified by OSD]

1. Free redistribution is allowed.
2. Source code can be obtained.
3. Derived works is allowed and the same license is applied.

4. Integrity of the author's source code may be requested.
5. No discrimination against persons or groups is allowed.
6. No discrimination against fields of endeavor is allowed.
7. No additional license is requested at the time of re-distribution.
8. License must not be specific to a product.
9. License must not restrict other software.
10. License must be technology-neutral.

Other than Linux, the below are examples of OSS in UNIX-based OSs. **The Open Group** is a group that is aiming for the standardization of UNIX, and mainly promotes UNIX-based OSs to be released as open source software.

- **BSD-family OSs**

This type of OS is OSS that is a derivative of BSD (Berkeley Software Distribution), which was developed to be UNIX-compatible. The OSs are developed with a focus on the points below.

- **FreeBSD**: well-developed support for Intel processors
- **NetBSD**: running BSD-family UNIX on many types of computers
- **OpenBSD**: the same philosophy as NetBSD, but with well-developed security

- **IRIX**

This is an OS that is developed and provided by Silicon Graphics. It is developed on the basis of System V, which is a UNIX-based OS.

Typical OSSs other than UNIX-based OSs include the following. Furthermore, **open source libraries** that compile OSS components as a package include, the Perl library **CPAN**, the PHP library **PEAR**, and the JavaScript library **jQuery**.

Software name	Purpose
Apache HTTP Server	Web server
BIND	DNS server
sendmail	Mail server
Firefox	Web browser
Chrome	Web browser
MySQL	Database
PostgreSQL	Database
PHP	General-purpose script language
Perl	Script language

Python	Object-oriented script language
Ruby	Object-oriented script language

[LAMP/LAPP]

This is a typical OSS combination that is used for developing database-driven web applications that run over the Internet.

- **LAMP**: Linux + Apache HTTP Server + MySQL + PHP
- **LAPP**: Linux + Apache HTTP Server + PostgreSQL + PHP

Note: In some cases, the script language may be Perl or Python instead of PHP.

The typical **OSS licenses** regarding the use of OSS are described below. For OSS licenses, in many cases, a principle called **copyleft** is adopted where reproduction, redistribution, modification, and such other activities are not restricted, and in return, the same license must be applied to derivative works. Therefore, a **dual license** in which this is combined with another license that places restrictions on reproduction, redistribution, modification, and such other activities, is sometimes used for distribution (or sale) of a derivative work.

- **GPL (GNU General Public License) / LGPL (GNU Lesser GPL)**

This is a license that is created by the FSF (Free Software Foundation), and is applied to software that can be used freely. This is mainly applied to the results (e.g., **GCC (GNU Compile Collection)**) of UNIX-compatible software development projects (**GNU/GNU projects**) that are promoted by the FSF. It is a typical copyleft license.

- **MPL (Mozilla Public License)**

This is a copyleft license that was created by Netscape and the Mozilla Organization for the web browser that was made open source by Netscape.

- **BSD (BSD License)**

This is a license that allows unrestricted reproduction, distribution, and modification if the copyright is displayed and a disclaimer (e.g., no guarantee) is included. It contains some slightly unique terms and conditions, such as redistribution being permitted without publishing the modified software's source code if certain conditions are met.

- **Apache license**

This is a BSD style license from the Apache Software Foundation. However, depending on the version, it is compatible with a GPL. Typical examples of this include **Tomcat**, which is server software that processes Java servlets/JSP.

In order to use OSS, as well as the scope of the licenses above, it is necessary to pay attention to the points below.

- The reliability and security of OSS is not always guaranteed (general rule of no guarantee).
- OSS does not always run on every OS or open source OS.
- OSS developer's rights (i.e., copyright) are not always relinquished.
- Most OSS is distributed free of charge, but it is not always free of charge.
- If there is a mechanism to obtain the source code, it might not be published on the Internet. In the same way, even if the OSS is improved and is not to be redistributed, the source code of the improved part does not have to be published.

Research and development of OSS is being advanced through many **open source communities** (i.e., groups that aim for OSS development, improvement, and information exchange) by volunteers (e.g., engineers, users) around the world across the network. The number of companies that are proactively adopting OSS is increasing, so it is a field in which development is expected to continue.

2 OS (Operating System)

An OS is software that manages resources such as hardware and enables effective use of such resources.

2 - 1 Functions and Configurations of OS

The three points below are the main aims of usage of an OS.

1. Effective use of resources

It enables the effective use of resources related to computers (hardware resources, software resources, and human resources). Typical methods for the effective use of resources include **multiprogramming** which shares resources among multiple programs and appears to run the programs concurrently, and this can improve throughput and shorten response time.

2. RASIS improvement

It improves the computer system reliability and ensures security. It enables the increase of MTBF by retrying an instruction that has caused a fault, and the reduction of MTTR by recording the fault status.

3. Providing usability

It improves the operability to decrease the burden on the user. It implements serial processing of jobs (i.e., programs) because it is time consuming for an operator to operate the computer every time a program is run.

In order to achieve these aims, the OS provides the management functions described below.

Management function	Role
Job management	To manage the execution of work (jobs) from the perspective of a user
Task management	To manage the execution of work (tasks) from the perspective of a machine
Memory management	To effectively use memory area
Data management	To manage access to data on auxiliary storage devices
Input/output management	To manage access to input/output devices
Network management	To manage network control
Operations management	To manage simplicity and flexibility concerning operations

User management	To manage information and controls relating to users
Security management	To manage information and controls relating to security
Fault management	To perform management relating to the detection, recording, and handling of a fault

2 - 2 Management Functions of OS

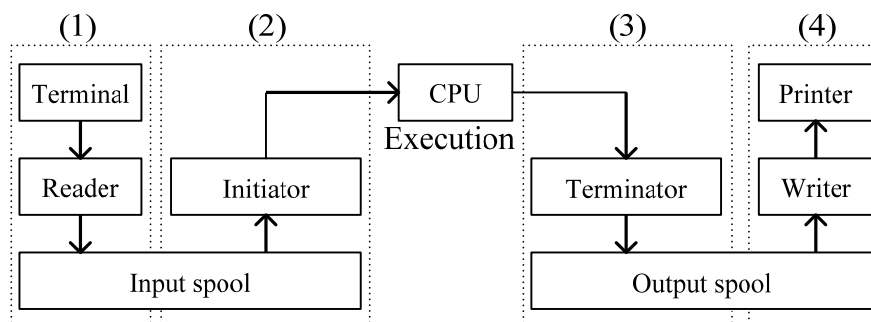
Among the various functions that are provided by an OS, job management, task management, and memory management are sometimes called the three main functions of an OS. On the basis of these three main functions, this section explains the various management functions of an OS.

2-2-1 Job Management

Job management is a function that manages the execution of **jobs**, which are the unit of activity from the perspective of a user. Generally, a job is defined as a set of **job steps**. A job step is a single unit of a divided work. Job management allocates resources to each job (i.e., job step), and it performs things such as consecutive execution of jobs.

The general flow (i.e., procedure) for executing jobs with job management is shown below.

[General job execution procedure]



- (1) The user enters a job (i.e., a program or an execution instruction) from an input device (e.g., terminal) by using a **JCL (Job Control Language)**. The entered job is read by a **reader**, and is temporarily stored in the input spool.
- (2) The job to be executed is selected from the jobs that wait for execution in the input spool and an **initiator** prepares for execution (e.g., resource allocation). After that, the job is executed.
- (3) After the execution of the job, a **terminator** performs post-processing (e.g., release of resources). If there are output results, they are stored in the output spool.
- (4) A result necessary for output is selected from the output results that are stored in the

output spool, and then, a **writer** sends the result to an output device (e.g., a printer).

Job management uses functions (1) to (3) below in order to implement this procedure for job execution.

(1) Spooling

Spooling is a method that enables a peripheral device (i.e., an input/output device) to operate in parallel and independently of a CPU. Specifically, it is a method that performs data transmission between a main memory unit and a low speed input/output device via a high-speed auxiliary storage device (**spool**). This releases the CPU from transmission waiting time and improves the throughput of the system overall. The term **buffering** is sometimes used to generally refer to a method such as spooling that uses an area (i.e., buffer) of memory to store data and transmits the data between devices that have different access speeds.

(2) Job scheduler

A **job scheduler** is a program that automatically executes jobs. It manages the control flow for jobs and also performs **job scheduling**, which selects a job to be executed from waiting jobs.

[Job scheduling methods]

- **Priority scheduling**

This method executes jobs in order of the priority assigned to each job, with high priority first. It is useful for improving the response performance in interactive processing by assigning a higher priority to interactive processing in systems that perform both interactive processing and batch processing. However, in some cases, if a higher priority is given to a job (i.e., CPU-bound job) that frequently uses the CPU, the CPU is monopolized and the waiting time of the other jobs increases.

- **Time slice**

This is a method that allocates the right to use the CPU to a job and executes it at a set time interval. The OS forcefully switches jobs and executes them consecutively, which means the CPU has no idle time and the throughput improves.

- **FCFS (First-Come First-Served)**

This is a method that processes jobs in order of arrival. It allocates the CPU equally between jobs, but if a job with a long processing time is started first, jobs that arrive later are made to wait and the throughput and response time deteriorate. This method is also called FIFO (First-In First-Out).

(3) Master scheduler

A **master scheduler** is a program that mainly manages the interface between the user and the computer. It transmits directions (e.g., JCL) relating to job execution and various types of messages between the user and the computer.

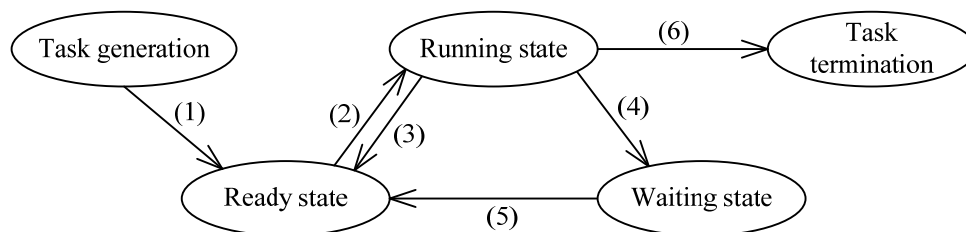
2-2-2 Task Management

Task management is a function that manages the execution of **tasks**, which are the units of work from the perspective of a machine (i.e., computer). Some OSs refer to a task as a **process**, so this is sometimes called **process management**.

A task becomes executable when the CPU is allocated to it. Therefore, the flow (i.e., procedure) of task execution with task management means a procedure for allocation of the CPU to tasks without waste. The management (i.e., execution) of multiple tasks in parallel is called **multitasking**. **Multiprogramming**, in which multiple processes appear to a user to be running simultaneously, is implemented with multitasking.

The general flow of task execution with task management can be summarized with a **state transition diagram** (i.e., a diagram displaying the state of a system that is changed according to time and events) as shown below.

[State transition diagram of tasks]



- (1) Tasks are generated from a job (i.e., job step) that is to be executed. Resources other than the CPU are allocated to the task, and when the CPU is allocated, the task enters the executable **ready state**.
- (2) When the CPU is allocated to the task, the task changes to the **running state**.
- (3) If a task with a higher priority than the currently running task changes to the ready state or the running task uses up the CPU time (i.e., time quantum) allocated to it, the running task is temporarily halted and returned to the ready state and has to wait for CPU allocation.
- (4) If the running task requests a resource (for example, execution of an input/output instruction), the task is halted and enters the **waiting state** until the requested resource is available.

- (5) As soon as the requested resource for the waiting task is available, the task changes to the ready state.
- (6) When the task is completed after the CPU is allocated several times, it is terminated.

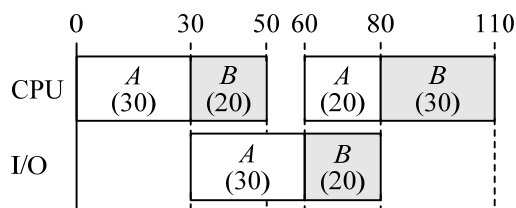
In order to halt a task that is being executed and run the management program, an **interrupt** occurs.

Example: If the two tasks *A* and *B* are executed concurrently, how many milliseconds does it take before both tasks are completed? Here, task *A* has the higher priority.

Task <i>A</i>	CPU (30)	I/O (30)	CPU (20)
Task <i>B</i>	CPU (20)	I/O (20)	CPU (30)

Each number in parentheses is the corresponding processing time (in milliseconds)

<Task scheduling results>



- (1) Tasks *A* and *B* are generated and enter the ready state.
- (2) The CPU is allocated to task *A* with the higher priority, and task *A* changes to the running state.
- (3) When CPU processing for task *A* is completed and task *A* requests I/O processing, an SVC interrupt occurs and task *A* enters the waiting state. I/O is not being used, so I/O processing for task *A* starts.
- (4) The CPU is allocated to task *B*, and task *B* changes to the running state.
- (5) When CPU processing for task *B* is completed and task *B* requests I/O processing, an SVC interrupt occurs and task *B* enters the waiting state. Task *A* is using I/O, so task *B* must wait for I/O to become available.
- (6) When I/O processing for task *A* is completed, an input/output (completion) interrupt occurs and task *A* changes to the ready state. As a result, I/O becomes available, and then I/O processing for task *B* starts.
- (7) The CPU is allocated to task *A*, and task *A* changes to the running state.
- (8) When CPU processing for task *A* finishes, task *A* is completed and is thus terminated.

- (9) When I/O processing for task *B* is completed, an input/output (completion) interrupt occurs and task *B* changes to the ready state.
- (10) The CPU is allocated to task *B*, and task *B* changes to the running state.
- (11) When CPU processing for task *B* finishes, task *B* is completed and is thus terminated.

Task management uses functions (1) to (3) below in order to implement this procedure for task execution (i.e., multitasking).

(1) Dispatcher

A **dispatcher** is a task management program that selects (**dispatches**) a task to be executed from multiple tasks in the ready state. The criteria for selecting a task are decided depending on task scheduling.

[Task scheduling]

- **Preemptive**

This is a method where a task being executed is temporarily halted under the control of the OS (or a preemption occurs).

- **Priority scheduling**

This is a method where the CPU is allocated to the task with the highest priority first according to the priority assigned to each task. It includes **static priority scheduling** in which the assigned priority cannot be changed, and **dynamic priority scheduling** which enables the priority to be raised (**aging**) when a process with a low priority is waiting for a long time without being executed (**starvation**). Methods, such as priority scheduling, in which the occurrence of an event is used as a trigger are called **event-driven** methods.

- **Shortest processing time first**

This is a method that assigns a high priority to tasks that have a short expected processing time and processes them earlier. This method increases the potential for tasks with a long expected processing time to wait continuously for CPU allocation.

- **Round robin**

This is a method that manages ready-state tasks in a queue in order of arrival, and at a given time interval (i.e., **time quantum**) allocates the CPU to the task at the head of the queue. Tasks that are not finished but have used up the allotted time are sent to the back of the queue. Methods that allocate a very small set amount of time to a task,

such as round robin, are called **time slice** methods.

- **Non-preemptive**

This is a method where a task that is being executed is not halted under the control of the OS.

- **First-come first-served**

This is a method that manages ready-state tasks in a queue in order of arrival. When a task that is being executed finishes, the CPU is allocated to the task at the head of the queue.

(2) Semaphore

Semaphore is used in mutual control (i.e., exclusive control), which synchronizes tasks and manages shared resources, when resources are allocated to a task. In actual terms, it performs control between tasks with the P operation when a resource is requested and the V operation when a resource is released.

[P operation / V operation]

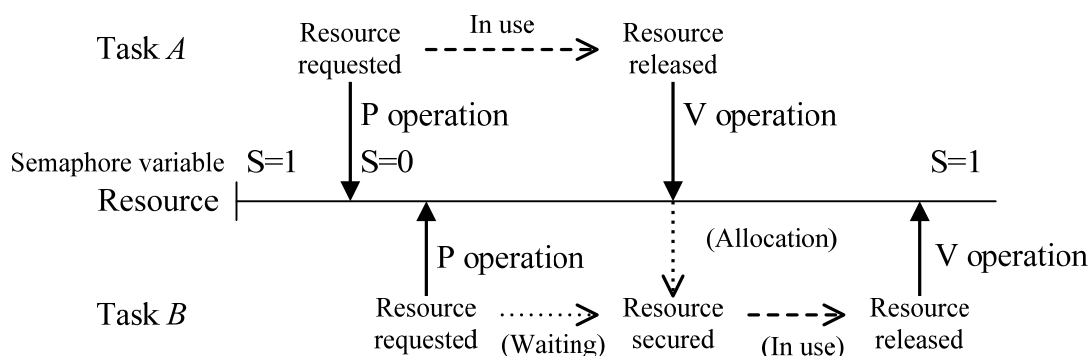
These are operations that secure and release resources on the basis of the semaphore variable of each resource. When semaphore variables can only be either 0 or 1 because only a single resource can be allocated, it is known as **binary semaphore**.

- P operation:

This is an operation that is performed when a resource is requested. If the semaphore variable is “1” or more, the resource is secured and “1” is subtracted from the semaphore variable. If the semaphore variable is “0”, the waiting state is entered until the resource is released.

- V operation:

This is an operation that is performed when a resource is to be released. If a task is waiting for the same resource, the resource is allocated and “1” is added to the semaphore variable.



(2) Threads

A **thread** is a subdivision of a task, and can only be allocated CPU resources. It is also called a lightweight process (lightweight task). For memory and other resources, it uses the resources that are allocated before subdivision. Not every OS has a thread function.

2-2-3 Memory Management

Memory management is a management function for the effective use of a computer's memory areas. Memory management is classified as either **real memory management** or **virtual memory management**.

(1) Real memory management

Real memory management is a management function for the effective use of main memory areas. A program to be executed must be recorded in main memory but there is a limit to the storage capacity of memory. Therefore, this method uses an auxiliary storage device.

(1) Partitioning method

This is a method that divides main memory into partitions for management. It is classified as either **fixed partitioning method** or **variable partitioning method**.

A: Fixed partitioning method

This is a method that divides the usable area of main memory into several partitions of a fixed size. This method reads (**roll-in**) a program or data from an auxiliary storage device when it is necessary, and writes (**roll-out**) the program or data back to the auxiliary storage device when it is no longer necessary.

- **Single fixed partitioning method**

This is a method that uses the usable memory area of main memory as a single fixed partition. Although the control is simple, main memory is used by only one program, and therefore, the unused area is large.

- **Multiple fixed partitioning method**

This is a method that uses the usable memory area of main memory as multiple fixed partitions. A program can be stored in each partition, but if a program is larger than the capacity of a partition, it cannot be stored.

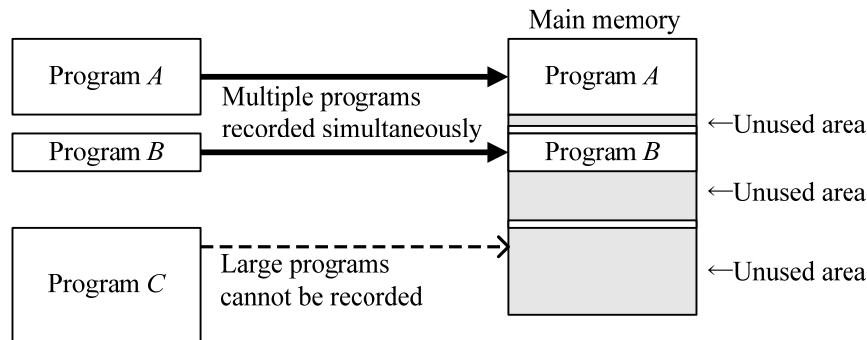


Figure 3-3 Fixed partitioning method (multiple partitioning method)

B: Variable partitioning method

This is a method that partitions the usable memory area of main memory according to the capacity required by a program.

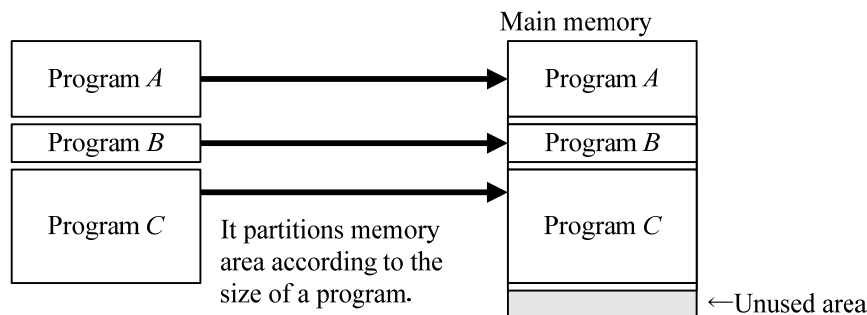


Figure 3-4 Variable partitioning method

When the variable partitioning method is applied, many small areas of garbage (i.e., unused areas) are generated because a partition remains unchanged after the execution of a program is completed. This is called **fragmentation**. In this case, **garbage collection** (or **memory compaction**) is executed to combine these unused areas into one large area.

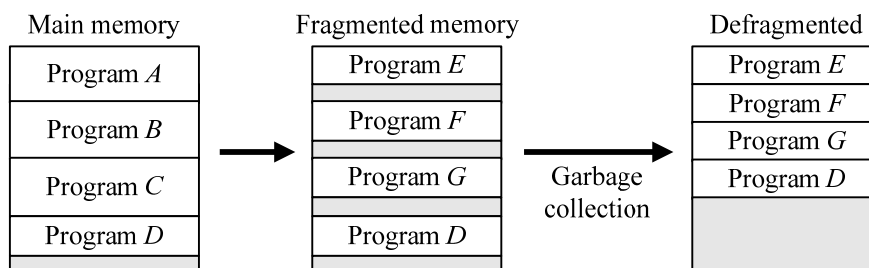


Figure 3-5 Fragmentation and garbage collection

Garbage collection can also be expected to have an effect whereby the potential for **memory leak** to occur is reduced. A memory leak is caused by a bug or such other problem in a program (e.g., an OS or an application) that uses memory (i.e., main memory), and areas that

were secured are not released during operation. If a memory leak occurs, usable memory areas in main memory decrease, and serious problems may be caused.

[Memory pool]

In this method, a program (e.g., an OS or an application) collectively secures memory areas in advance, and then allocates the secured memory areas as needed. Memory pool management also uses a fixed length method that allocates memory blocks of the same size and a variable length method that allocates memory blocks of the required size.

(2) Overlay method (Segment method)

This is a method that is used in order to execute a program that is larger than the usable area of main memory. It divides the program into multiple segments that are not executed concurrently and transfers segments to and from an auxiliary storage device upon execution.

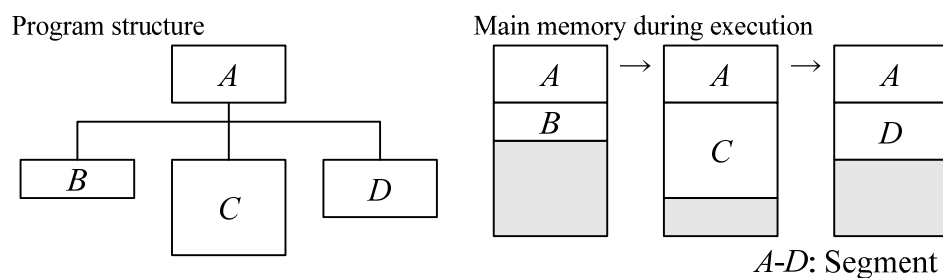


Figure 3-6 Overlay method

(3) Swapping

In multiprogramming and such other environment, swapping refers to interrupting execution of the current program and transferring it to an auxiliary storage device in order to execute programs or such other processes of a higher priority. As soon as main memory has available space (or when execution is restarted), the program that was transferred to the auxiliary storage device is retrieved and processing is restarted from where it was interrupted. The activity to transfer from main memory to auxiliary storage is called **swap-out**, and the retrieval activity is called **swap-in**. If swapping occurs frequently, much time (i.e., overhead) that is not related to a program is required, and processing efficiency is drastically decreased.

(2) Virtual memory management

Virtual memory management is a virtual memory system that implements a concept called “virtual memory” by establishing a logical memory space (a virtual memory space or a logical

address space) that is larger than main memory and executing programs that are stored in it. (In reality, only programs in main memory can be executed, so the section to be executed is loaded (i.e., transferred) from auxiliary storage to main memory).

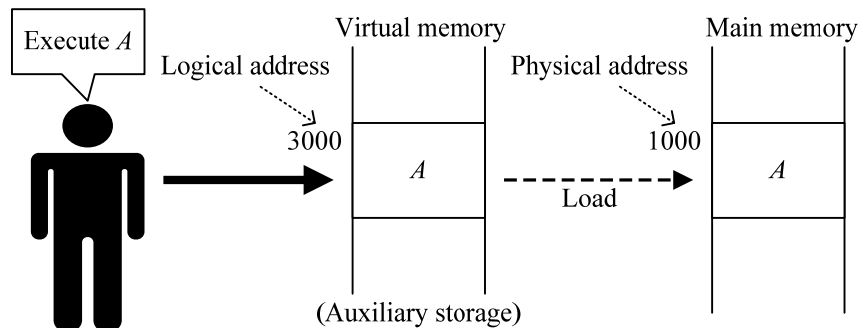


Figure 3-7 Image of virtual memory system

Virtual memory management is classified as shown below according to the difference in the management method of memory areas.

- **Paging**

This is a method that manages the memory areas of main memory and auxiliary storage by dividing them into units (i.e., pages) of a set size. Since transactions between main memory and auxiliary storage are performed in units of a fixed length, waste is less likely to occur in their memory areas. The loading of a page from auxiliary storage to main memory is called **page-in**, and the reverse of this is called **page-out**.

- **Segment**

This is a method that divides a program into meaningful units (i.e., segments), and performs transactions between main memory and auxiliary storage in units of segments.

- **Paged segment**

This is a method that handles pages (i.e., memory that has been divided into units of a set size) by compiling them into segments that have a logical relation. There are also another approach that handles multiple pages as a segment even if there is no logical relation.

In a virtual memory system, it is necessary to convert between the virtual memory address (logical address) and the main memory address (physical address). This conversion is called address conversion and is performed by hardware called a **DAT (Dynamic Address Translator)**.

In the segment method, address conversion is performed using the **base address method**, which references a segment address table in which the starting points (i.e., physical addresses) of segments are recorded. On the other hand, in the paging method, addresses conversion is performed by a DAT referencing a page address table in which the physical address of each page is stored.

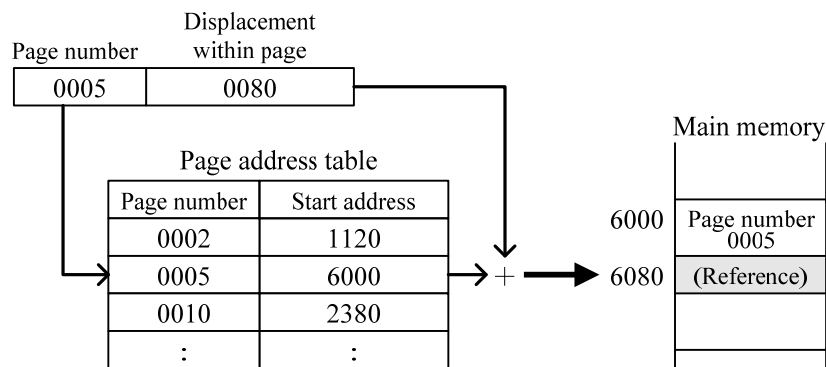


Figure 3-8 Address conversion in paging

In this case, when a page that is not recorded in the page address table (or not recorded in main memory) is referenced, an interrupt called a **page fault** occurs from the DAT. When a page fault occurs, the page for reference is loaded into main memory, and the page address table is updated. If main memory has no free space at this time, a page (i.e., page for replacement) from main memory is selected, a page-out is performed, and then a page-in is performed in order to put the referenced page into the free space.

If page faults frequently occur, page-in and page-out increase, and a state called **thrashing** occurs in which processing efficiency decreases. Therefore, in order to minimize the occurrence of page faults, it is necessary to perform a page-out for pages that have a low probability of referral after this point. The method that is used to select such pages is called a **page replacement algorithm**, and there are three typical ones.

- **FIFO** (First-In First-Out)

This performs a page-out for the page that has been in main memory the longest.

- **LRU** (Least Recently Used)

This performs a page-out for the page for which the longest time has passed since the last reference.

- **LFU** (Least Frequently Used)

This performs a page-out for the page that has been referenced the least.

Example: If the page capacity of main memory is three pages and the pages are referenced in order of {1, 2, 3, 2, 1, 4, 1, 2}, how is “page-in” performed with FIFO or LRU?

Referenced page	FIFO	LRU
1	(1)	(1)
2	1 (2)	1 (2)
3	1 2 (3)	1 2 (3)
2	1 (2) 3	1 (2) 3
1	(1) 2 3	(1) 2 3
4	(4) 2 3	1 2 (4)
1	4 (1) 3	(1) 2 4
2	4 1 (2)	1 (2) 4

() : Referenced page, : Page-in

The apparent memory capacity of main memory can be increased with the virtual memory system. However, if the capacity of main memory is too small, thrashing occurs and processing efficiency decreases. This means that for running applications that will not run because of insufficient memory, increasing the size of virtual memory is only a temporary solution. In such cases, it is necessary to consider increasing the usable memory areas by adding memory or stopping the use of unnecessary applications.

2-2-4 Other Management Functions

(1) Data management

Data management is a function that provides a method to access data stored on an auxiliary storage device through an interface that does not depend on the auxiliary storage device. In order to use data, it is necessary to consider physical aspects, such as the characteristics of the auxiliary storage device and the data recording method. However, a user cannot use a computer easily if it is necessary to think about this. Hence, the role of data management is to allow the user to manipulate data without the need to be aware of such matters. Data management manages **file organization format** and **access methods**. The details of file organization format and access methods are explained in section 4.

(2) Input/output management

Input/output management performs tasks such as entering data to be processed by a computer and controlling input/output devices in order to obtain output of processing results. Upon a request for input/output (**SVC interrupt**), processing according to the **input/output control method** (e.g., in the case of channel control, sending a channel program to the input/output channel) is performed. Upon completion of input/output, notification of completion of input/output is performed with an **input/output interrupt**. Furthermore, in some cases, the role of input/output management includes a function that expands the concept of input/output devices and provides interfaces with peripheral devices that use device drivers or such other programs.

(3) Network management

Network management is a function that performs network control in relation to communication. It uses a **network control program** to provide an interface to connect to a **LAN (Local Area Network)** or a **WAN (Wide Area Network)**. It also performs **protocol control** that uses **communications protocols** (e.g., **OSI basic reference model**, **TCP/IP**) and **message control**, which manages the order of data transactions and such other functions. “Network” is explained in detail in Chapter 5.

(4) Operations management

Operations management is a function that performs management of a computer system’s operational status from system startup processing (i.e., OS initialization) to system shutdown processing. It provides support for the effective use of a system, such as operation **scheduling** and creation of records through the **monitoring** of various information (e.g., usage status of system resources, billing information), and achieves simplicity and flexibility in operation. In some cases, the functions for **user management** described later are included in operations management.

(5) User management

User management is a function that issues a user ID for each user, and provides functions, such as **profile function** that saves user information and system environments and **account function**.

Accounts in account functions refer to “rights.” Many rights are required in order to use a computer (or a system) such as the **right to use a system**, **access rights** for using data (of

note, the rights regarding files are called **file access rights**) and **rights to use a terminal**. The setting/changing/deletion of such rights for each user is called account management. The user ID (and password) that is used to identify a user for performing such actions is sometimes called a **user account** (i.e., rights granted to users). In addition to general users, types of user accounts include a **superuser** (also known as an **administrator** in Windows-family OSs and a **root** in Unix-family OSs) that has administrator privileges (i.e., full rights) for a computer or a system, and a **guest** that signifies a temporary user. Furthermore, in some cases a **directory service** is used to share a user account across all terminals connected to a system. A directory service is a database service for the centralized management of resource information over a network, and the **LDAP (Lightweight Directory Access Protocol)** is commonly used for access.

(6) Security management

Security management is a function that maintains the confidentiality, integrity, and availability of information in order to maintain the security of a computer. Controls that apply to this include: **access control** which prohibits unauthorized access to resources with the account functions of user management, **encryption control** which encrypts recorded data to prevent information leakage, and controls to detect and prevent unauthorized intrusion from external locations. In order to confirm that these security controls are functioning effectively, it is verified as a **reliability process** that **accountability** is secured, through functions such as a **logging function** which records the history (i.e., log) of usage, and an **audit function** which inspects (i.e., audits) events such as login and logout. Chapter 6 explains the details of security.

(7) Fault management

Fault management is a function that monitors hardware and software, and takes measures at the occurrence of a fault. It covers both **hardware faults** that are caused by hardware problems and **software faults** that are caused by software problems. Depending on the fault, the appropriate measure, such as isolation of the area where the fault occurred and reconstruction of the system, or restart of the system, is implemented. Furthermore, by recording in advance the fault status, cause, and measures when a malfunction occurs, this can be used to reduce the recovery time (i.e., MTTR) when the same type of fault occurs.

3 Programming Languages and Language Processors

A programming language is a language for creating a program that instructs a computer to do work. A language processor translates a program that is written in a programming language into another language that a computer can understand.

3 - 1 Classification of Programming Languages

In a computer, all information is recorded digitally. In other words, a computer is only able to understand the states 0 and 1. For this reason, in the early stages, **machine languages** were used to give instructions to a computer for work with combinations of the values 0 and 1.

However, it is very difficult for humans to write programs in a machine language (using 0 and 1) and mistakes are prone to occurring. Hence, programming languages were invented to make symbols correspond to combinations of 0s and 1s. These are called **assembly languages** or **symbolic languages**.

However, assembly languages still create symbols from combinations of 0s and 1s. Therefore, in order to create a program with assembly languages, it is necessary to have a detailed understanding of the mechanisms of the computer (i.e., hardware). In order to solve this, programming languages that enabled human thought processes to be written in a notation that is easy for humans to understand were developed. These were classified as **high-level languages**, and the machine languages and assembly languages that were used until this point were classified as **low-level languages** to distinguish them.

[Classification of high-level languages]

- **Procedural language**

This is a language that describes processing procedures (i.e., algorithms) on the basis of the execution (or procedure) of instructions, such as operation instructions and data manipulation instructions.

- **Functional language**

This is a language that describes functional programs to obtain output data by applying functions to input data.

- **Logic language**

This is a language that describes the relationships and conditions that exist within a problem, by using logical expressions, and resolve the problem.

- **Object-oriented language**

This is a language that describes object-oriented programs on the basis of the units called objects that unify data and processes (i.e., procedures).

- **Script language**

This is a language that describes simple programs (called scripts).

The programming languages that are generally used at the present day are explained below.

3-1-1 Assembly Language

Assembly languages are the closest language to machine languages. One assembly language instruction corresponds to one machine language instruction. The structural organization and grammar of an assembly language differs depending on the computer.

- **CASL II**

This is a programming language that is used in the Fundamental Information Technology Engineer Examination, and it is an assembly language that runs on the COMET II virtual computer.

3-1-2 Procedural Language

Procedural languages are programming languages that use a procedure as the basic unit and describes process procedures (i.e., algorithms). Almost all procedural languages use a **compilation method** to translate a whole written program (i.e., source program) into a machine language program (i.e., object program) before it is executed.

- **Fortran**

This is a programming language for scientific and technological calculation. It is the first high-level language that was widely used.

- **COBOL**

This is a programming language for business processing. It is not suitable for the development of programs that use graphics.

- **PL/I**

This is a programming language that combines the characteristics of Fortran and COBOL. Since it can handle both scientific, technological calculation and business processing, it is also called a general purpose programming language.

- **Pascal**

This is a programming language that was developed for programming education. It also had an effect on programming languages that were developed later.

- **BASIC**

This is a conversational programming language for beginners. It also supports the **interpreter method**, which interprets and executes a program on a line-by-line basis.

- **C**

This is a programming language that was developed by AT&T Bell Laboratories to describe programs for the UNIX OS. It has high portability and is widely used for system development on a workstation or a PC.

3-1-3 Object-oriented Language

Object-oriented languages treat data and data processing (i.e., procedures or methods) as a unified unit called an object, and programs are created using objects. **Object orientation** is an approach that does not focus on processes (i.e., procedures), but on the data that is subject to processing.

- **C++**

This is an object-oriented language that includes the concepts and functions of object-orientation. It is used for more general purpose than C and has a high level of reusability.

- **Java**

This is an object-oriented language that was developed by Sun Microsystems, Inc. in the United States on the basis of C++. The programs (i.e., **Java applications**) developed with Java support **multi-platform** in which a program can be executed on different hardware or a different OS as long as in the environment in which **JVM (Java Virtual Machine)** is installed. Since it does not depend on specific types of computers or OSs, it is used in web applications that run on the Internet and also in distributed system environments. Currently, a compilation method that uses a **runtime compiler (JIT compiler)** is prevalent.

- **Java Servlet**

This is a Java program that is executed on a web server in response to a request from a client. A **servlet** is a program that is executed on the server side.

- **Java Applet**

This is a Java program that is downloaded by a client from a web server and is executed on the client. An **applet** is a program that is downloaded to the client-side to be executed.

- **JavaBeans**

This is a specification for handling a program that is developed with Java as an application component.

- **VB (Visual BASIC)**

This is an object-oriented language that was developed by Microsoft. It enables the creation of a program in a visual environment that uses windows and toolboxes. A

macro language called **VBA (VB for Application)** which enables linking with other applications based on VB is supported by products, such as Excel, Word, and Access.

- **Macro**

This is a mechanism that automatically executes a series of specific operations in an application. There are two methods to create a macro: one is to record the operation procedure, and the other is to describe the operation procedure with the macro language.

3-1-4 Script Language

Script languages are simple programming languages that enable the end user to develop a simple program (called script). Script is an easily executable series of instructions. In script languages, the end user creates programs through various specifications in a GUI environment by using a mouse or other tools.

- **JavaScript**

This is an object-oriented script language to write programs that run in a web browser. Some parts of it are similar to Java, but it is completely different. It enables programming by directly writing instructions into the codes of a web page. It is used in the creation of web pages.

- **ECMAScript**

This is a specification for JavaScript that was standardized by Ecma International (former ECMA, European Computer Manufacturers Association).

- **VBScript**

This is an object-oriented script language to describe programs that run in a web browser. Some of the VB functions can be used on a web page.

- **PostScript**

This is a page description language that was developed by Adobe Systems. It describes text data, graphics, and other elements in units of pages. It is mainly used in DTP software for printing.

- **Perl/PHP/Python/Ruby**

The **CGI (Common Gateway Interface)** is the mechanism in which a web server launches an application program in response to processing requests from a browser. These are languages that describe applications (i.e., CGI programs) that are launched with this mechanism. Each language has its own characteristics. However, they all share excellent text processing capabilities.

[CLI (Common Language Infrastructure)]

CLI (Common Language Infrastructure) is an execution environment in which applications written in multiple high-level languages can be executed in different system environments without the need to rewrite those applications to take into consideration the unique characteristics of those environments. Originally this was a specification for virtual machines for Microsoft's .NET Framework, but it is now standardized by **ISO/IEC 23271 (JIS X 3016)** as an environment (or specification) that is able to execute programs that are written in various high-level languages on a common platform.

3-1-5 Markup Language

Markup languages are a type of programming language for printing and screen display. Text (i.e. a character string) is enclosed in marks that are called **tags**, and attribute information is attached. Structural elements of a document (e.g., texts, paragraphs, titles, tables and diagrams) and their relationships can be easily defined.

(1) **SGML (Standard Generalized Markup Language)**

This is a general purpose markup language that was standardized by ISO. It enables description of the logical structure, semantic structure, and attributes of a document. So, document management and data conversion can be performed easily. In SGML, the structure of a document is defined with a **DTD (Document Type Definition)**.

(2) **HTML (HyperText Markup Language)**

This is a hypertext description language that extends SGML to enable it to handle not just text but also graphics, images, movies, and so on. Hypertext interconnects content with other content (i.e., web page) through a link comprising character information.

HTML encloses text with a **start tag** and an **end tag** to specify character size and color, image files to be included, web pages to be linked, and so on. Although the specifications are standardized, companies that create web browsers add their own extended specifications. Therefore, some tags can only be used in browsers created by a specific manufacturer.

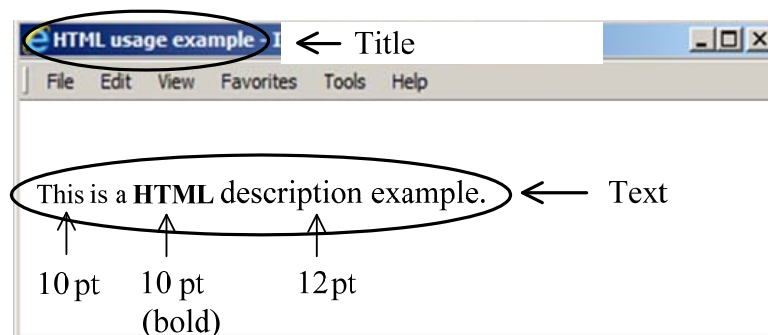
Furthermore, **CSS (Cascading Style Sheets)** can be used to define only the layout of a web page that is completely separated from HTML.

Example: How is the following HTML document displayed? The standard font size for the text is 10 pt.

```
<TITLE>HTML usage example</TITLE>  
<BODY>This is a <B>HTML</B><FONT = "+2">description example.</FONT></BODY>
```

- <TITLE> is a tag that indicates the title of the web page.
- <BODY> is a tag that indicates the text of a web page.
- is a tag that emphasizes characters within the tags by using a bold typeface.
- is a tag that alters the font (size) of the characters within the tags.

<Display example>



(3) DHTML (Dynamic HTML)

This is an extended HTML specification. By incorporating a script written in JavaScript or VBScript into an HTML page, it enables the dynamic creation of a web page according to user operations, and the checking of user input data.

(4) XML (Extensible Markup Language)

This is a derivative of SGML and HTML, and is a hypertext description language that enables bidirectional linking. It is highly suitable for the Internet environment and is often used for corporate transactions via a network. By using a **DTD (Document Type Definition)**, users can define their own tags. Currently, a dedicated XML schema language called **XML Schema** is often used instead of DTD. In HTML, the end tag can be omitted in some cases, but in XML there must be a start tag and an end tag even for a null element.

XSL (Extensible Stylesheet Language) can be used for style sheets in XML.

[Technologies related to XML]

- **XML parser**

This is software for the use of the structural elements of an XML document from an application. It allows use of APIs such as **DOM (Document Object Model)**

which manages XML documents in a tree, and **SAX (Simple API for XML)** which reads and interprets an XML document in order from the top.

- **SOAP (Simple Object Access Protocol)**

This is a communication protocol for message exchange that calls data or services based on XML.

- **SVG (Scalable Vector Graphics)**

This is a format for two-dimensional vector graphics based on XML.

- **SDL (Service Description Language)**

This is an XML-based language that defines functions for web services. It has been absorbed by **WSDL (Web SDL)**.

- **Ajax (Asynchronous JavaScript + XML)**

This is a mechanism that transmits XML documents without synchronization between a browser and a web server, and dynamically redraws the screen.

(5) **XHTML (EXtensible HyperText Markup Language)**

This is a markup language that has redefined HTML, which extends SGML, by using XML. The specifications are stricter than HTML; for example, “it is not possible to leave out a tag.” **XHTML Basic** is intended for use on cell phones and personal digital assistants and is a subset of combined modules defined on the basis of the specifications of the modularization of XHTML.

3 - 2 Language Processor

A **language processor** is software that translates programs that are written in a programming language (other than machine language) into a machine language or an **intermediate language** (a language between a machine language and a high-level language). Even though the program before translation and the program after translation can process the same work, their content differs and they are distinguished as described below.

- **Source program**

This is a program that is written in a programming language other than machine language.

- **Object program**

This is a program that has been translated into machine language or an intermediate language.

3-2-1 Types of Language Processors

(1) Assembler

An **assembler** is a language processor that translates a source program written in an assembly language into an object program. Translating with an assembler is called “**assembling**.”

(2) Compiler

A **compiler** is a language processor that translates a source program written in a high-level language into an object program. Translating with a compiler is called “**compiling**.” A high-level language that is translated by a compiler is called a compiler language. Typical examples include COBOL and C. Since instructions of high-level languages are translated into multiple machine languages, a dedicated compiler for each language is required.

[Translation procedure of a compiler]

(1) **Lexical analysis**

This divides a source program which is a sequence of character strings in units of tokens.

(2) **Syntax analysis**

This analyzes the character strings according to the grammar of the programming language, and generates a syntax tree.

(3) **Semantic analysis**

This analyzes the meaning of a syntax tree and generates an intermediate code.

(4) **Optimization**

In order to increase processing efficiency upon execution and reduce execution time, this simplifies expressions in the intermediate code and modifies the structure.

(5) **Code generation**

This converts the intermediate code into an object code of the machine language (or an intermediate language).

(1) **Lexical analysis**

A general programming language is defined with **formal language** based on **context-free grammar**. Therefore, the program is divided into units of character strings (i.e., tokens) with **finite automaton** or other method which is used to interpret **regular expressions**. In general programming languages, identifiers, constants, keywords,

operators, and such other elements are divided as tokens.

- **Context-free grammar**

This is a grammar that enables symbols to be replaced without relying on the preceding or subsequent context. For example, with context-free grammar, if “A” is defined as “xy” or “yz”, “xyz” can be replaced by either “Az” or “xA”.

- **Formal languages**

These are languages, such as programming languages, for which a strict grammar is defined. In contrast, the language that is used in everyday conversation is sometimes called natural language.

- **Regular expression**

This is a notation method that is used in the definition of a string format (i.e., pattern).

[Example of notation in regular expression]

[A-Z]*[0-9]+

[A-Z]: Represents a single alphabetical character

[0-9]: Represents a single number

*: Represents the repetition of zero or more times of the immediately preceding regular expression

+: Represents the repetition of one or more times of the immediately preceding regular expression

Note: In the strings defined in this example, after an alphabetical character is repeated zero or more times, a number is repeated one or more time. In other words, “AB1” and “123” are applicable, but “ABC” and “12A” are not applicable.

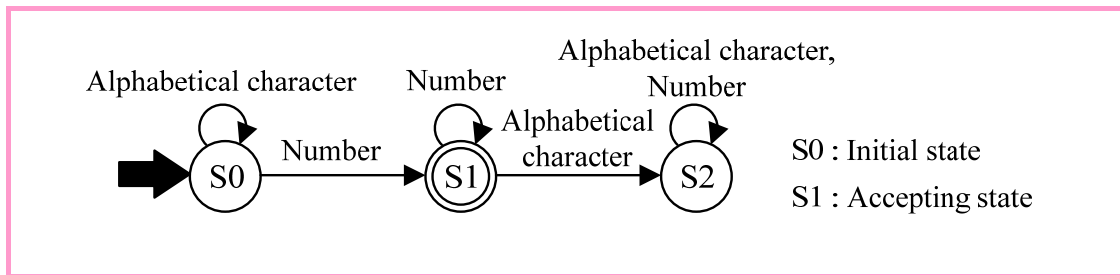
- **Finite automaton**

An **automaton** is a notation method that models the difference in processing of input depending on the internal state. A finite automaton is composed of a finite number of states and transitions, and is represented with a **state transition table**, a **state transition diagram**, or such other method.

[Representation with a state transition table]

	S0	S1	S2	
Alphabetical character (A - Z)	S0	S2	S2	S0: Initial state S1: Accepting state
Number (0 - 9)	S1	S1	S2	

[Representation with a state transition diagram]



(2) Syntax analysis

This interprets the token structure according to the grammar of the programming language, and generates a **syntax tree**. In context-free grammar, syntax is often defined with **BNF notation** or a **syntax diagram**, so this can be called an activity to confirm that the tokens in the program constitute the correct meaning.

• BNF (Backus Naur Form) notation

This is a **metalanguage** (i.e., a language to define a language) that defines the grammar of a programming language or such other aspects. It is also used for the definition of XML, and so on.

[Example of BNF notation]

`<expression> ::= <numeric value> | <expression><operator><numeric value>`

`<numeric value> ::= <number> | <sign><number> | <numeric value><number>`

`<operator> ::= <sign> | * | /`

`<sign> ::= + | -`

`<number> ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9`

Note: In this example, “+4-12*6” is interpreted as below.

+4-12*6

→ <code><number><sign><number><number><operator><number>

→ <numeric value><sign><numeric value><number><operator>
<numeric value>

→ <numeric value><sign><numeric value><operator><numeric value>

→ <numeric value><operator><numeric value><operator><numeric value>

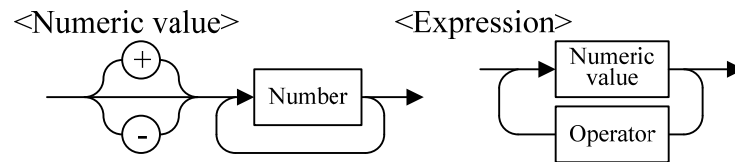
→ <expression><operator><numeric value><operator><numeric value>

→ <expression><operator><numeric value>

→ <expression> ... interpreted as correct <expression>

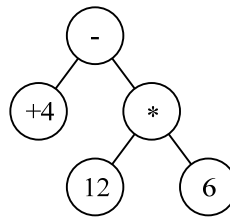
• Syntax diagram

This is a diagram of a definition in BNF notation that is visually easy to understand.



- **Syntax tree**

This is a diagram of the results of syntactic analysis in a tree structure. An **abstract tree structure** especially refers to a tree structure that is composed of only the information that is required for semantic analysis.



(3) **Semantic analysis**

This analyzes a tree structure generated in syntax analysis for meaning in terms of a programming language, and generates an intermediate code. Semantic analysis includes interpretation of sections that cannot be analyzed with the grammar of the programming language alone (e.g., validity of variable names, calling of external functions). In the intermediate code, the **quadruple format** or **reverse Polish notation** is sometimes used to represent formulas.

- **Quadruple format (three-address statement)**

This is a method to represent an expression with four elements (i.e., operator, operand 1, operand 2, and result, including three variable addresses) in the form.

For example, expression “s=4-12*6” is represented as below.

(*, 12, 6, T1)

(-, 4, T1, s)

- **Reverse Polish notation**

This is a notation method for formulas in which an operator is written after variables (or expressions) or numeric values as the subject of the operation. Since the order in which the operators appear is order of execution, it is used to represent formulas in a computer. (In some cases it is also used instead of a syntax tree.) There is also a notation method in which an operator is first written, which is called **Polish notation**.

For example, an expression “s=4-12*6” is expressed as follows:

Reverse Polish notation: (s 4 12 6 * - =)

Polish notation: (= s - 4 * 12 6)

(4) Optimization

This analyzes intermediate code, modifies the program structure (e.g., expression simplification, composition modification), and optimizes the code so that it can be executed at a higher efficiency level.

Optimization method	Details
Inline function expansion	This method expands the function to be called, at the location where the function is called.
Common subexpression elimination	This method calculates the value of an expression that exists in multiple locations as a variable for use in operations, and replaces the expressions with this variable.
Constant folding	This method replaces expressions consisting of constants with the results of the expressions.
Constant propagation	This method replaces a variable with a constant.
Dead code elimination	This method eliminates statements that have no effect on the execution results.
Loop-invariant code motion	This method shifts an expression whose value does not change in a loop to outside of the loop.
Loop unrolling	This method expands the iteration process in a loop.

(5) Code generation

This converts the optimized intermediate code into the target object code. Basically, the intermediate code is converted into machine language. However these days, it is temporarily converted to an intermediate code (e.g., byte code using **Java byte code** or **CIL (Common Intermediate Language)**), and then, upon execution, converted into an environment-dependent machine language.

Furthermore, depending on the intended use, the types of compilers below are available.

- **Precompiler (preprocessor)**

This is a compiler that interprets additional (or supplemental) functions contained in a high-level language program, and then converts to a high-level language program before actual compilation.

- **Optimization compiler**

This is a compiler that focuses on the optimization of processing, and reduces the

size and execution time of the object program after compiling as much as possible.

- **Cross compiler**

This is a compiler that generates an object program on one computer for another computer that has a different machine language. An assembler that is used for the same purpose is called a **cross assembler**.

- **Runtime compiler (JIT compiler (Just-In-Time compiler))**

This is a compiler that translates a source program (i.e., source code) or intermediate code that is written in an intermediate language into a machine language just before its execution. It takes a somewhat long time at the start of execution, but it has the advantage of being able to execute a program without depending on the execution environment.

(3) Interpreter

An **interpreter** is a language processor that translates and executes a source program written in a high-level language one instruction on an instruction-by-instruction basis. The differences between a compiler and an interpreter are as below.

	Compiler	Interpreter
Object program	Generated	Not generated
Translation units	All statements at once (translation not possible if a program is not completed)	One instruction at a time (translation and execution possible even if the program is not completed)
Execution speed	Fast	Slow
Suitable application	Batch processing	Interactive processing

(4) Generator

A **generator** is a language processor that generates object programs by providing parameters, such as input data/output data specifications and process conditions. Since it can generate an object program without the need to be aware of the processing procedure, it is sometimes referred to as a **non-procedural language**.

(5) Translator

In the case of a language processor, a **translator** is software that converts a program into a

different programming language. A compiler or an assembler can be called a translator that converts (i.e., translates) a source program into an object program. This does not apply to interpreters.

3-2-2 Service Programs

An object program created by a language processor cannot be executed as is. In order to execute an object program, various **service programs** (i.e., utility programs) are required. There are also service programs that play an auxiliary role in program creation and execution but have no direct relationship with program execution.

This subsection explains typical service programs.

(1) Linker (linkage editor)

A **Linker** (i.e., **linkage editor**) is a service program that assembles (i.e., links) multiple object programs (i.e., object modules) into a single executable program (i.e., **load module**). In many cases, a source program that is translated into machine language by a compiler is originally a module into which a single program was divided. In such cases, the compiler processes other modules (e.g., external functions) as an unresolved address. The role of the linker is to resolve this unresolved address and integrate the divided and translated object program. When the linker does this, it also has a function to retrieve components (i.e., **library modules**) that are registered in the **program library** (i.e., **load library**) and incorporate them.

Linking performed by a linker is called **static linking** in order to distinguish it from **dynamic linking** which incorporates components (i.e., modules) from a **DLL (Dynamic Link Library)** by the OS just prior to execution.

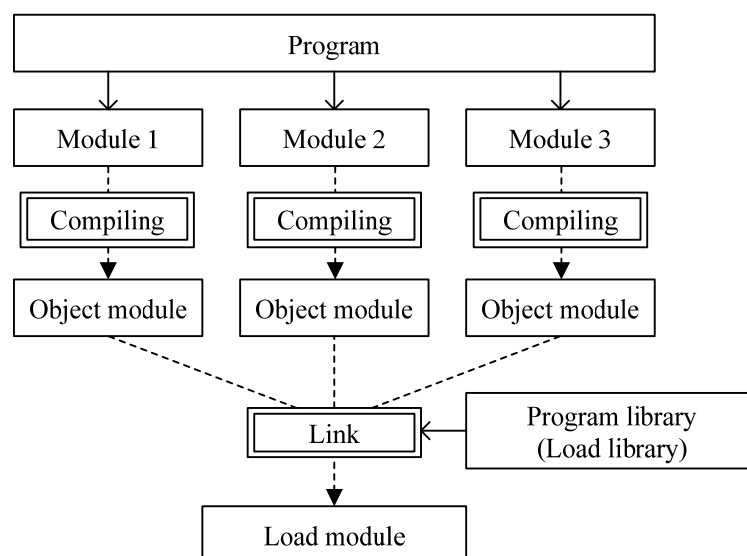


Figure 3-9 Linker (linkage editor) operation

(2) Loader

A **loader** is a program that stores load modules in main memory and executes them. There is also a program called a **linking loader** which makes an object module into an executable load module and stores it in main memory.

(3) Editor

An **editor** is software that performs an auxiliary role concerning program input, modification, and other programming activities. Generally, this refers to a **text editor** that is used for character input, but there are also other editors such as **graphical editors** that are used for inserting diagrams or images and **structure editors** that increase the input efficiency according to the structure of a specific programming language.

(4) Emulator

An **emulator** is a microprogram that creates a pseudo environment including a different OS and other programs, and executes the programs of another type of machine.

- **ICE (In-Circuit Emulator)**

This is an emulator that imitates the functions of a given microprocessor by using hardware in the development of a microprogram or such other activity. It is used as a testing tool (i.e., debugging tool) in development.

(5) Simulator

A **simulator** is a program or system that performs simulation concerning the complex phenomena of natural science and such other fields. In the case of a language processor, it is sometimes used as a program development tool to simulate the behavior of a program.

(6) Sort/merge programs

A **sort program** is a program that performs sort processing on data. A **merge program** is a program that performs merge processing to combine multiple sorted data into one. Since there is a close relationship between sorting and merging, they are often provided as a common service program (i.e., sort/merge program).

3 - 3 Program Attributes

Programs that are executed by a computer have several properties. Depending on the property of the program that is executed, the method of operation is different. It is necessary to pay attention to this point.

(1) Relocatable program

A **relocatable program** is a program that can be read into memory by a loader and executed at any address. It is sometimes called a **relocation program**. The program is made executable by actions including correction of the address information corresponding to the load position. Furthermore, programs for which the address can be changed even during execution are called **dynamic relocatable programs**.

(2) Reusable program

A **reusable program** can be used any number of times. Normally, a program must be reloaded into main memory for each execution. (This is called a reload.) However, a reusable program does not require any reloads and can be used repeatedly once it is loaded. When the execution of a reusable program starts, actions including the initialization of variables are performed to maintain process consistency. Therefore, it cannot be used for multiple tasks at the same time.

(3) Reentrant program

A **reentrant program** allows the correct result to be obtained even if it is used in multiple tasks simultaneously. A reentrant program is divided into a variable part and a procedure part, and implements concurrent processing by allocating a variable part to each task while only the procedure part is shared. In order to be used in multiple tasks simultaneously, a reentrant program requires the properties of **reusable programs** that can be used repeatedly without reloading.

(4) Recursive program

A **recursive program** is a program that can call itself during program execution. It can return to the state immediately prior to execution by recording the states during processing with the LIFO (Last-In First Out) method. However, even if the program is able to call itself, there is no guarantee that it can also be called from another task, so it is not necessarily a reentrant program.

4 Files

A file is a unit that a computer uses to manage information. The data management function of an OS can be described as a function to manage files. This section introduces file organization formats and access methods.

4 - 1 Files and Records

Below is a list of the units of information in a computer.

- **Bit**
The smallest unit of memory in a computer. It records either 0 or 1.
- **Byte**
This is a collection of multiple bits (i.e., eight bits). Since a character code is usually represented in 8-bit form, it is a unit that represents a single character.
- **Item**
This is a unit that is composed of several characters and has a single meaning.
- **Record**
This is a unit that includes multiple related items and is processed as data.
- **File**
This is a unit that includes multiple records of the same format and manages data.

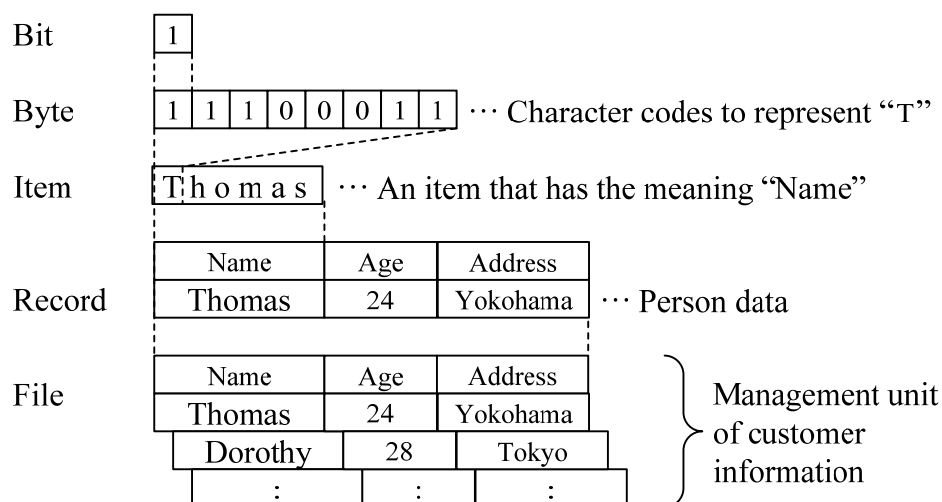


Figure 3-10 Units of information in a computer

4-1-1 Classification of Records

(1) Physical records and logical records

In a computer, data is processed in units of records, which have meaning. However, to handle records that are actually stored on an auxiliary storage device, it is not efficient to perform input and output on a record basis, so this is performed with a unit that groups multiple records called a **block**. This block, which is an input/output unit, is referred to as a **physical record** as it is a record that is handled physically. This is distinguished from a record that forms a processing unit, which is called a **logical record**.

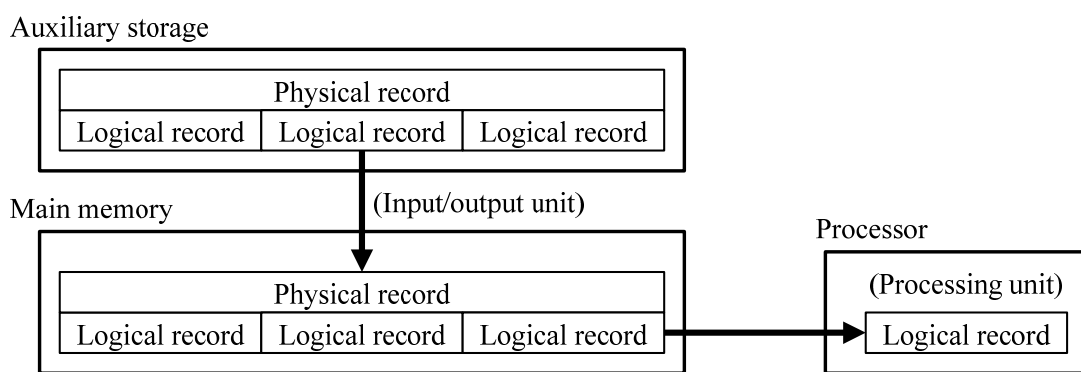


Figure 3-11 Physical records and logical records

(2) Record formats

Below is the classification of record formats.

- **Fixed length record**

A record format in which the size (i.e., number of bytes) of all records is fixed. These records can be made into blocks, and are also easy to handle.

- **Variable length record**

A record format in which the length of each record is different. Even though the record size varies, these records can be made into blocks because the record size information is stored at the beginning of each record.

- **Undefined length record**

A record format in which the length of each record is different like variable length records. Undefined length records, however, cannot be made into blocks, because they contain no information concerning the size of each record.

4-1-2 Classification of Files

File classifications include classifications depending on the purpose of usage, storage (or usage) period, users, or such other factor. These classifications are separate from the file access methods and organization format that are described later.

(1) Classification based on purpose of usage

- **Master file**

This is a file that records the information that forms the core of business (e.g., product information, customer information).

- **Transaction file**

This is a file that records information, such as updates for a master file.

- **Historical file**

This is a file that records the history of a process. It is also called a journal file or a log file, and is mainly used as a measure against faults.

- **Backup file**

This is a file that is an exact duplicate of the content of an important file. It is mainly used as a measure against faults.

- **Working file** (work file)

This is an intermediate file that is created temporarily during the implementation of a process.

(2) Classification by storage (or usage) period

- **Archived file** (saved file)

This is a file that is to be continually used over a long period, and is also called a permanent file. This classification includes files such as master files.

- **Temporary file**

This is a file that is used temporarily as needed and is generally erased after it is used. This classification includes files such as transaction files and working files.

(3) Classification based on users

- **System file**

This is a file for system operations and management, such as programs that constitute an OS.

- **User file**

This is a file that records programs or data used by a user.

4 - 2 File Access Methods

An **access method** is a method for reading the records stored in a file and writing records to a file.

(1) Sequential access

Sequential access is a method for accessing records in order from the start of a file.

It can only read records from a file from the beginning of the file in order. Therefore, it is a good method for reading and processing all records in order from the top of the file, but it is not suitable for processing from a specific record in the middle of a file.

When it writes records to a file, it appends the records after the last record in order (if there are no records already present, it writes in order from the top), so there is no wasted area. This means that it is efficient for recording, but to insert a record in the middle of a file, it is necessary to remake the file.

(2) Direct access (random access)

Direct access (**random access**) is a method that directly accesses a specific record independent of the order of the records in a file.

It is suitable for processing (i.e., reading or writing) on a specific record, but is not suitable for the processing of all records.

(3) Dynamic access

Dynamic access is an access method that combines direct access and sequential access.

This is explained here with the example of the creation of a list of names beginning with “s.” The method for reading and writing records will be to first perform direct access to records with names beginning with “s”, and then perform sequential access for the subsequent records. Dynamic access enables the simultaneous use of these two access methods.

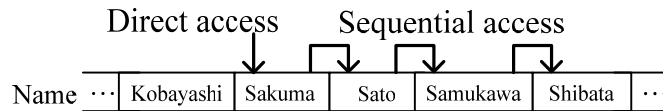


Figure 3-12 Dynamic access

4 - 3 File Organization Formats

A file **organization format** defines the order of records to be recorded in a file and the structure. The user must consider specifically how the file is to be used and consider which organization format is appropriate.

4-3-1 Sequential Organization

Sequential organization file has a dedicated organization format for sequential access, which writes records sequentially on storage media. Its usage of storage media is efficient because the records are written sequentially, but it cannot insert records in the middle of a file. Since records are always processed in order, it is effective to read records in advance by using a buffer. This organization can be implemented on all storage media including DASD (Direct Access Storage Devices). However, on magnetic tape, only sequential organization can be implemented.

4-3-2 Relative Organization

Relative organization file has a format that writes records in order as in a sequential organization file, and then assigns a relative number (i.e., relative address) to each record from the top of records. Since the records are written sequentially, it uses storage media efficiently, and direct access with the relative number of a record is also possible.

4-3-3 Indexed Sequential Organization

An **indexed sequential organization file** (or indexed sequential file) is composed of two areas: an area for data, and an area in which information (i.e., index) for direct access to a specific record is stored. Its usage of storage media is efficient, and it can create large files.

- **Prime area**

This is an area in which records are stored. Records in this area are stored in logically ascending order, and sequential access is possible.

- **Index area**

This is an area in which index records that match record keys with record addresses are stored. Types of indexes include a master index, a cylinder index and a track index.

- **Overflow area**

This is an area in which records ejected from the prime area are stored. If many records are stored in this area, the efficiency of access decreases and reorganization is required. Types of overflow areas include a cylinder overflow area and an independent overflow area.

4-3-4 Partitioned Organization

In a **partitioned organization file**, a file is divided into multiple units called **members** to store it. The information from each member is managed with a **directory**.

Members are subfiles into which a sequential organization is divided, and update management can be performed in units of members. However, deletion of members only removes the members from the directory and the corresponding members in the memory area remain as is. As a result, the number of waste areas in which unused members are stored increases, and storage efficiency decreases, so it is necessary to periodically reorganize memory areas.

Partitioned organization files are generally used as a program library.

4-3-5 Direct Organization

A **direct organization file** has a dedicated format for direct access that enables access to a specific record by the specification of an address. The location of records is not sequential, so it does not use storage media efficiently. It can only be implemented on DASD (Direct Access Storage Devices) such as hard disks.

In order to specify a specific record, direct addressing and indirect addressing are available as methods to calculate the address by using the key item of a record.

(1) Direct addressing

Direct addressing uses the value of a record key as the record address. No processing is required to calculate the address, but depending on the record key, there may be many variations of addresses, so it is not practical.

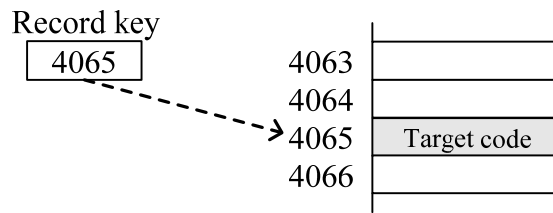


Figure 3-13 Direct addressing

There are also methods that use a correspondence table for record keys and record addresses. However this requires a large number of elements equal to the records to be stored, so its usage is limited to a small number of records.

(2) Indirect addressing

Indirect addressing changes the value of a record key in accordance with a rule and calculates the record address. This conversion is called **hashing**, and the used rule (i.e., function) is called a **hash function**. Therefore, a direct organization file that uses indirect addressing is sometimes called a **hash organization file**.

[Methods for calculating a record address from a record key with indirect addressing]

- **Division**

This is a method that divides a record key by a given value (e.g., generally the closest prime number to the total number of records) and uses the remainder as the address.

Example: When the record key is 123456 and the number of records is 100,

$$123456 \div 97 \text{ (closest prime number to 100)} = 1272 \dots 72$$

=> 72 is used as the record address.

- **Superposition**

This splits a record key into parts in accordance with a rule, and uses the sum of these parts as the address.

Example: When the record key is 123456, it is split into parts comprising the three digits in the first half and the three digits in the second half.

$$123 + 456 = 579$$

=> 579 is used as the record address.

- **Radix conversion**

Record keys are normally handled as decimal values, but this uses the result of conversion to a radix other than decimal (e.g., ternary) as the address.

Example: When the record key is 123456, ternary radix conversion is performed

$$1 \times 3^5 + 2 \times 3^4 + 3 \times 3^3 + 4 \times 3^2 + 5 \times 3^1 + 6 \times 3^0 = 543$$

=> 543 is used as the record address.

Note: In the case of ternary, the value of each digit normally does not exceed 2, but it is not necessary to consider this here.

With indirect addressing, different keys may be converted to the same record address and this phenomenon is called the occurrence of a **synonym**. When this happens, the record that can be used with the converted address is called the **home record**, and the one that cannot be used is called the **synonym record**.

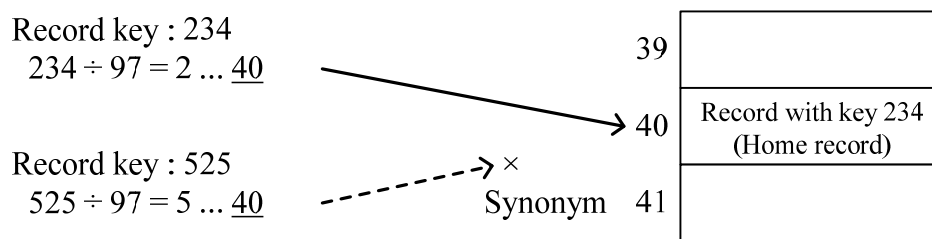


Figure 3-14 Occurrence of a synonym (with division by 97)

There are two methods to deal with a synonym. These are described below. In both methods, the subject record cannot be accessed with a record address calculated from the key, so the access efficiency is reduced; in other words, access time increases. Therefore, it is preferable that there be no bias in the distribution of key values, and that the record addresses calculated from key values be close to **uniform distribution** (i.e., a distribution in which the occurrence of a phenomenon has equal probability).

- **Sequential method**

This stores the synonym record in an available area that is close to the calculated record address. Synonyms are prone to occurring in some cases.

- **Chain method**

This secures another memory area, and stores the synonym record in it. In this case, a pointer is required to indicate the address in which the synonym record is stored.

4-3-6 VSAM Organization

VSAM organization files can be used with an OS that uses a virtual memory system. This can also be described as a way of implementing conventional file organization format virtually.

In VSAM organization, a file is called a data set. The table and diagram below show the correspondence between data sets and file organizations, and data set structure, respectively.

Data set name	Corresponding file organization
ESDS (Entry Sequence Data Set)	Sequential organization file
KSDS (Key Sequence Data Set)	Indexed sequential organization file
RRDS (Relative Record Data Set)	Direct organization file/relative organization file

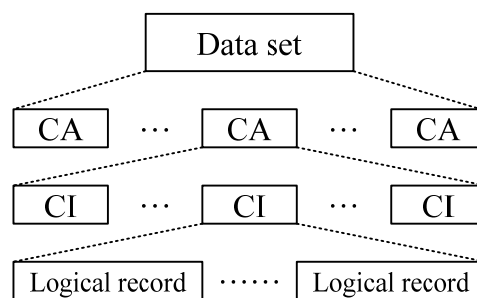
[Data set structure]

- **CA (Control Area)**

This is a logical control area that manages partitioned memory areas.

- **CI (Control Interval)**

This is a logical input/output unit that groups multiple logical records.



4 - 4 File Management in Small Computers

The explanation so far mainly covered files on large general-purpose computers. From here, the explanation will cover file management in the OS (e.g., Windows, UNIX) of a small computer (e.g., PC).

4-4-1 File Systems

A **file system** is a mechanism that manages files that are stored on auxiliary storage media such as a hard disk drive. A different file system is provided by each OS, and a typical example is the **FAT (File Allocation Tables) file system** that is used in Windows-family OSs. In a FAT file system, a hard disk is called a **volume**, and a volume is divided into units called **clusters** (or hard disk sectors are grouped) for management. File input/output is performed as a batch in units of clusters, so the role of software is to identify these as records or items. **FAT32** is a typical file system, which uses a cluster identifier with 32 bits to divide a volume into 2^{32} clusters. But currently, NTFS (NT File System) is widely used, because the size of HDD has become large and the maximum volume size of NTFS is designed to be large. Other typical file systems except for the Windows OS are shown below.

File system name	Supported OSs	Notes
UFS (Unix File System)	UNIX-family	Divided into partitions
HFS (Hierarchical File System)	Mac OS	HFS+ is also used on iPods.

4-4-2 Directory Management

The OS of a small computer manages files hierarchically in a tree structure. In UNIX, the whole file system is managed as a single tree. In this system, a **directory** is used to group multiple files in the higher hierarchy.

In order to create a hierarchical structure, a directory at the highest level is necessary. This is called a **root directory**. Some files are stored in the root directory, but handling all files at the same level leads to complex management. Therefore, directories (called **subdirectories**) that group related files into a directory are created beneath the root directory. Beneath a subdirectory, files or other subdirectories can be created. Figure 3-15 is a hierarchical structure diagram that shows an example of the hierarchical management of files.

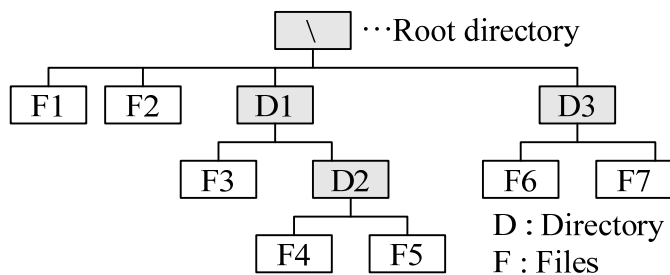


Figure 3-15 Hierarchical structure diagram for files

In a hierarchy, a user can only see beneath the directory (called the **current directory**) that they are currently using. In this example, if the current directory is the root directory, the user can see only F1, F2, D1, and D3. Therefore, in order to manage the files beneath a subdirectory, the user must change the current directory. For example, if the user wishes to use F3, the current directory must be changed to D1.

However, changing the current directory each time requires excessive actions, so files are referenced with the two methods below that specify a **path** (i.e., route). A path that references a file is sometimes called **reference information**.

- **Absolute path**: specifies the route from the root directory.
- **Relative path**: specifies the route from the current directory.

Example: In Figure 3-15, what is the absolute path and the relative path to specify F3 when the current directory is D2? Here, “..” indicates the parent directory. “\” indicates the root directory when it is at the top of a path and a separation when it is in the middle.

Absolute path: \D1\F3 (root directory → D1 → F3)

Relative path: ../F3 (current directory D2 → parent directory D1 → F3)

Furthermore, in the case of a multi-user computer, file management for each individual user is required. As a directory for this purpose, a **home directory** and a **desktop directory** are used. The current directory after login when the OS is started is called the home directory. This is the highest level in which general users can perform actions, such as saving files. When a user logs in, a dedicated desktop for that user is displayed. The management information for this is stored in the desktop directory.

4-4-3 File Sharing

File sharing is a mechanism to enable multiple users to use a single file. By setting the **right to access a file** (i.e., file reference permission) for files stored on a file server or NAS (Network Attached Storage), it is possible to make a file available as shared file. In NAS storage, file systems that support multiple protocols are embedded, so file sharing between different OSs or between different types of servers can be easily performed. If a **file search function** provided by an OS or other software is used, it is possible to search for desired files including shared files, just by specifying part of a filename or the date of the last update.

4-4-4 Symbolic Links

A **symbolic link** is a function that assigns a different name to a file or directory in the file system of a Unix-family OS. It allows a user or application to handle it in the same way as the actual file. From the perspective of the user or application, it is the same as handling the original file. Through a symbolic link, access to the file or directory can be performed. The same type of function is provided as a **shortcut** in Windows, and an **alias** in Mac OS.

4-4-5 Folders

In a Windows-family OS, file manipulation is also performed in a GUI environment. In other words, a file to be used is not specified as a path but is specified by selecting an icon.

In this case, the name **folder** is used instead of directory. Since it is possible to create a folder

inside a folder, the user must select folders until the desired file is displayed. In other words, by selecting folders to switch windows, the user changes the current directory. Folders can also be shared in the same way as files.

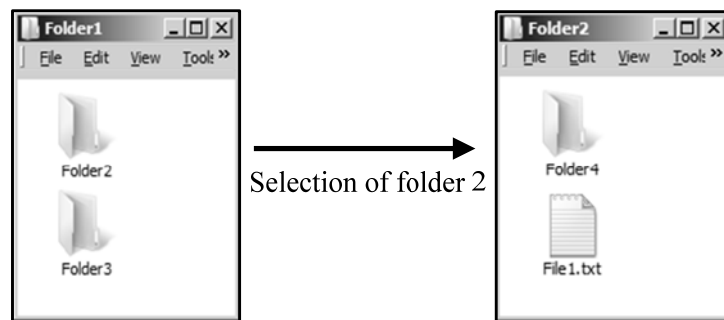


Figure 3-16 GUI environment file management (folder)

4 - 5 Backup

Backup refers to saving data, such as files, on an auxiliary storage device in case of a fault or such other situation. Since the data recorded in a file is very critical, it is necessary to make efforts to improve reliability and availability through backups. In many cases, as the storage media for a backup, a streamer, such as a **DDS (Digital Data Streamer)**, is used.

[Types of backups]

- **Full backup**

Copies all data. The time taken for the backup is long, but a restore can be performed only with the latest full backup data.

- **Differential backup**

Copies all data that has been updated since the previous full backup. The restore for this uses only the latest full backup data and the latest differential backup, so it involves less work than an incremental backup.

- **Incremental backup**

Copies only the data that has been updated since the previous backup (full backup and incremental backup). The time taken for the backup is shorter than a differential backup, but the restore uses the latest full backup and all incremental backup data since the full backup.

[Points for attention in backup operations]

- Backups should be performed periodically after a schedule is created. Especially in cases where there are multiple people involved, a plan should be created so that backup can be implemented outside of working hours.
- In order to prevent operational errors or neglecting to perform backup, backup processing should be automated as much as possible, which also makes it less laborious.
- Storage media that has plenty of capacity should be selected. However, if files are backed up onto the same HDD they cannot be read if a fault occurs, so a separate storage media should be used.
- Thorough consideration should be given to the importance of the files in terms of the storage location of the backup files. If the backup files are stored in the same room as the originals, they may also be lost if a fire or other disaster occurs. If it is a room for which entry/exit control is not performed, then important information may be obtained by a third party.
- In order to protect data from disasters, accidents, or such other situation, a **multiple backup** that copies twice, the main and the secondary, should be taken and stored in a location that is geographically separate.
- Backup files should have generation information (e.g., version numbers), and management should be performed for as many generations as possible.

Chapter 3 Exercises

Q1

The sentences below are descriptions concerning a language processor, a service program, and a control program. Which of the following is the appropriate combination of *A*, *B*, and *C*?

- A*: It provides an efficient operating environment through the allocation of hardware resources, or such other function.
- B*: It performs program translation or its related function.
- C*: It provides programs that are generally deemed to be necessary, such as a data management utility and a disk management utility.

	<i>A</i>	<i>B</i>	<i>C</i>
a)	Language processor	Service program	Control program
b)	Service program	Language processor	Control program
c)	Control program	Language processor	Service program
d)	Control program	Service program	Language processor

Q2

Which of the following is software that can be used free of charge for a trial period, but if the user wishes to continue using the software after this period, a usage fee must be paid?

- a) Shareware
- b) Package software
- c) Public domain software
- d) Freeware

Q3

Which of the following is an appropriate description concerning open source software?

- a) Open source software is a concept that only exists for application software, and no open source operating systems exist.
- b) Open source software can be freely modified, and copyright must always be relinquished.
- c) Open source software can be freely redistributed, and redistribution must not only be for a specific group or individual.

- d) Open source software is distributed free of charge, and derivative works must also be distributed free of charge.

Q4

Which of the following is an appropriate description concerning a spooling function?

- a) When a task is being executed, the CPU is allocated to another task if the CPU switches to the idle state as a result of the execution of an input/output instruction.
- b) By grouping small areas in main memory into one, a large usable area is secured.
- c) The program being executed is temporarily halted, and control is transferred to the control program.
- d) By performing data transmission between a main memory unit and a low-speed I/O device via an auxiliary storage device, the throughput of the entire system is improved.

Q5

A process is executed through a state transition among three states: a ready state, a running state, and a waiting state. Which of the following is an appropriate description concerning the state transition of a process?

- a) If multiple processes are executed simultaneously in which CPU processing and I/O processing appear alternately, each process makes a state transition between the two states of the running state and the waiting state only.
- b) Ready state refers to a state in which a process is waiting for CPU allocation. Generally, there are multiple processes in the ready state, and they are formed into a queue.
- c) In a multiprogramming system, even with only one CPU, multiple processes in the running state exist.
- d) In systems that perform time-sharing processing with the round robin method, a process in the running state makes a state transition to the waiting state after a fixed time passes.

Q6

Which of the following is an appropriate description concerning fragmentation?

- a) In the variable partitioning method, fragmentation does not occur even if memory areas of various sizes are acquired and released.
- b) In the fixed partitioning method, the acquisition and release of memory areas are performed faster than in the variable partitioning method, but fragmentation tends to occur.
- c) Although the total available memory is sufficient, the required memory area cannot be acquired in some cases, because of fragmentation.
- d) In a system with a high frequency of acquisition and release of memory, garbage collection must be performed whenever a memory area is released.

Q7

Among the descriptions concerning segments and paging in virtual memory, which of the following is a characteristic of paging?

- a) The size of an area that forms a unit of management in a virtual address space can be dynamically changed.
- b) The usage of actual memory areas is efficient, and area management is also simple.
- c) Access can be performed in a logical unit that is seen by a program.
- d) During the execution of a program, other programs can be loaded in units of programs.

Q8

The page replacement algorithms that are used in a virtual memory system include FIFO and LRU. Which of the following is an appropriate description of the basic concept of these page replacement algorithms?

- a) They predict the page that is most likely to be deleted after that point in time.
- b) They predict the page that is least likely to be deleted after that point in time.
- c) They predict which page will be referenced in the nearest future after that point in time.
- d) They predict which page will not be referenced in the farthest future after that point in time.

Q9

Which of the following is an appropriate explanation of a Java servlet?

- a) It is a program that is developed with Java and executed after the download from a web server.
- b) It is a program that is developed with Java and executed on an application server in response to a request from a client.
- c) It is a set of rules for handling a program that is developed with Java as an application component.
- d) It is an interpreter (i.e., execution environment) that executes programs that are developed with Java, and has a function to execute a sort of intermediate code called bytecode.

Q10

Which of the following is a standard document description language that defines the elements of a document, such as document title, chapter title, section title, diagram, and table, and the relationships between these elements?

- a) CSS
- b) HTML
- c) SGML
- d) SOAP

Q11

Which of the following is an appropriate explanation of optimization in a compiler?

- a) Analyzing the meaning of a syntax tree in terms of a programming language, and generating intermediate code
- b) Generating an object code which runs on a computer whose architecture is different from the computer which performs compilation
- c) Performing actions such as changing the program structure, and generating code that increases processing efficiency during execution
- d) When a program is executed, translating a source code or an intermediate code to generate an object code

Q12

Which of the following is an appropriate description concerning a DLL (Dynamic Link Library)?

- a) It is embedded by a compiler during compilation.
- b) It is generated by a pre-compiler before compilation is performed.
- c) It is linked to by the OS during the time of execution.
- d) It is linked to by a linkage editor when a load module is created.

Q13

Which of the following is an appropriate description concerning the program attributes?

- a) In order to implement recursive processing, the status of a running program must be recorded and controlled with an FIFO method.
- b) Reusable programs can also be reentrant.
- c) In order to implement a reentrant program, the program must be divided into a procedure section and a variable section, and a variable section must exist for each task.
- d) A program that can be executed by multiple processes concurrently is recursive.

Q14

Which of the following is a file organization format that is composed of multiple members and is most appropriate for a program library?

- a) Partitioned organization file
- b) Indexed sequential organization file
- c) Sequential organization file
- d) Direct organization file

Q15

Which of the following is an appropriate description concerning a file system?

- a) As the highest directory in a hierarchy, the current directory is created first.
- b) In order to manage files by using directories, the FAT file system must be used.
- c) In order to specify a certain file in a file system, a path that goes through directories is specified.
- d) It is necessary to separately create a directory in which files are registered and a directory in which lower-level directories are registered.

Q16

Which of the following is an appropriate description concerning a file backup?

- a) In a system such as RAID0 that performs striping, a file backup is completely unnecessary.
- b) In order to efficiently perform backup operations, backup files are created on the same physical device.
- c) A backup can be effective for restoring files that are lost due to a hardware fault or a human error.
- d) It is inexpensive to use a magnetic tape for the backup of file contents, but it cannot be automated.

Chapter 4

Database

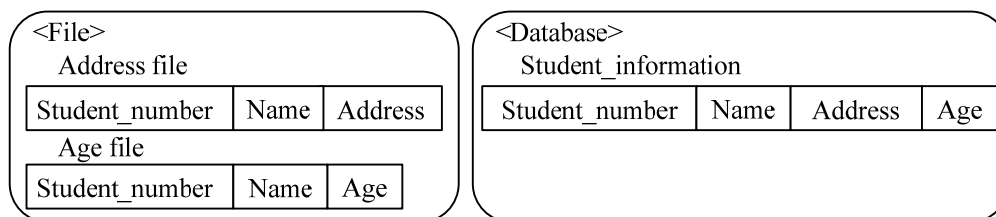
1 Outline of Database

Methods of handling data in computers include the method that uses files and the method that uses a database. A database is a method where data is consolidated and managed in an integrated manner. This section describes the concepts of a database and its operations.

1 - 1 Difference Between Database and File

Since the **file** is created for each program, there is a risk that the problem occurs – such as overlapping of data items and data inconsistency because of different contents of overlapping data items.

Therefore, as the integrated management of data, the concept of **database** becomes popular because it prevents overlapping of data and inconsistency of data.



- **Overlapping of data items:** Student number and name of the same student are recorded as overlapping in two files (address file and age file).
- **Data inconsistency:** When the name is modified in the address file but not in the age file, the student with the same student number has a different name.

Figure 4-1 File and database

The following functions are required for a database.

- **Data sharing function**
It allows identical data to be provided to multiple users.
- **Data independence function**
It ensures that data modification does not affect the program.
- **Data maintenance function**
It ensures that data is always in the correct status (or there is no contradiction).
- **Data fault countermeasures function**
It allows any fault of a database to be quickly addressed.
- **Data security protection function**

It ensures security of data by setting access rights, and so on.

1 - 2 Database Design

For using a database, it is necessary to analyze the data to be handled and design the database.

1-2-1 Data Model

Data model refers to modeling of various relations of data in the real world for handling them in computers. Data models are classified from the viewpoint of modeling.

- **Conceptual data model**

It models relations of data without considering any specific database. It defines what kind of data is handled.

- **Logical data model (external model)**

It logically models the relations of data considering a specific database. It defines interrelations of data in the database.

- **Physical data model (internal model)**

It physically models the relations of data considering a specific database product. It defines the physical internal structure of a database such as data type.

Among these models, the following three data models are used as the logical data models considering a specific database.

- **Hierarchical model**

It is a data model that represents the relations of data as a hierarchical structure where parent and child are in a one-to-many relationship. It is implemented as **HDB (Hierarchical DataBase)**.

- **Network model**

It is a data model that represents the relations of data as a hierarchical structure where parent and child are in a many-to-many relationship. It is implemented as **NDB (Network DataBase)**.

- **Relational model**

It is a data model that represents the relations of data in a two-dimensional table. It is implemented as **RDB (Relational DataBase)**.

The database, such as HDB (Hierarchical DataBase) or NDB (Network DataBase), is referred

to as **structured database**. This is a type of database where the user recognizes logical layout of data.

In this textbook, the subsequent explanation is provided focusing on the RDB (Relational DataBase) that implements the relational model.

1-2-2 Relational Model

Relational model is a data model that represents the relations of data in a two-dimensional tabular form. The entire table is called “**relation**,” rows and columns that constitute the table are called “**tuples**” and “**attributes (fields)**” respectively.

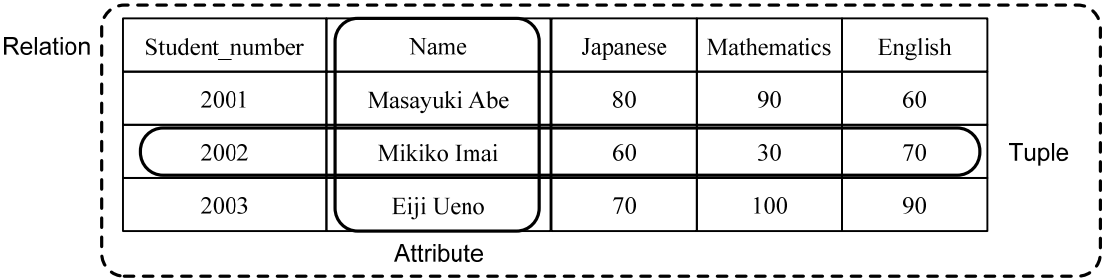


Figure 4-2 Relational model

In Figure 4-2, values that are actually recorded, such as “2001”, “Masayuki Abe”, “80”, “90”, and “60”, are referred to as **occurrence** (or instance). In addition, a set of occurrences that an attribute can take is referred to as **domain**. (Domain may include data types and constraint conditions of attributes.) Therefore, relation is defined as a subset of the direct product of domains. (**Direct product**: a set operation that obtains all combinations, **Subset**: a set that is included in a certain set)

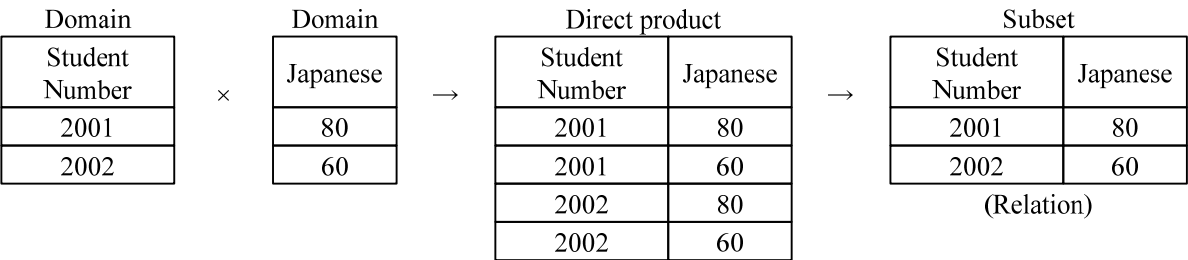


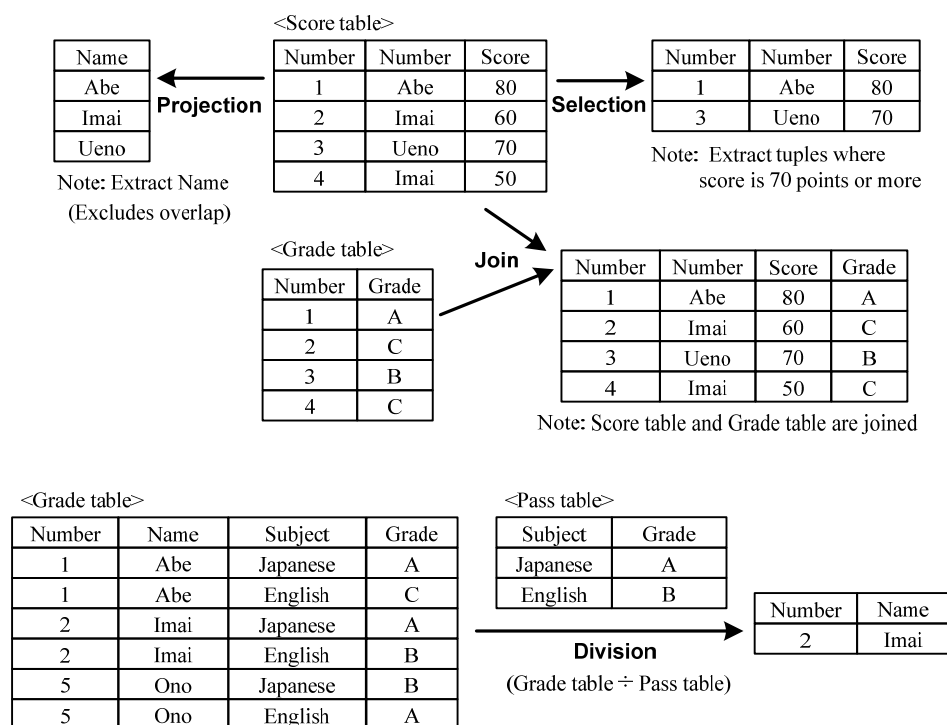
Figure 4-3 Definition of relations

The relational model has a close relationship with the set. Therefore, in the relational database that implements a relational model, the database is operated in combination with **relational operations (relational algebra)** that are specific to relational database and **set operations** that are based on the concept of set.

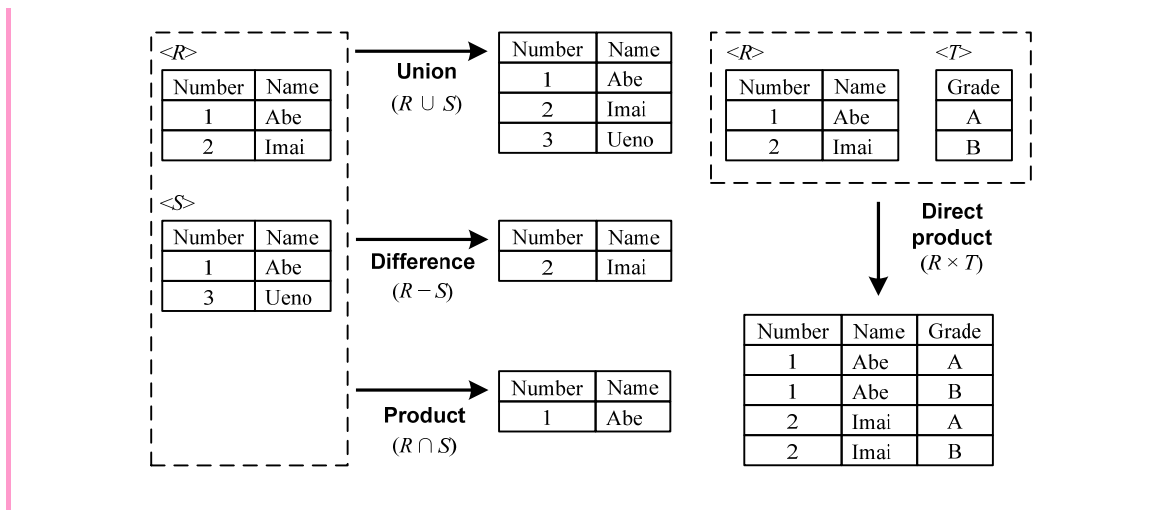
	Operation name	Operation contents
Relational operation	Selection	Extracts the tuples that satisfy the specified conditions.
	Projection	Extracts the specified attributes.
	Join	Combines multiple tables into one.
	Division	Extracts the tuples that are identical to all tuples in another table.
Set operation	Union	Extracts all tuples from two tables excluding any overlap.
	Difference	Extracts all tuples excluding the tuples that belong to another table.
	Product	Extracts the tuples that appear commonly in two tables.
	Direct product	Combines all tuples in two tables.

[Examples of relational operations/set operations]

• Relational operations



• Set operations



In addition, there are following manipulations of the relational database.

- **Insert:** Adding (i.e., inserting) new data (i.e., tuples) in a relation (i.e., table)
- **Update:** Changing (i.e., updating) the data (i.e., tuples) recorded in a relation (i.e., table)
- **Delete:** Deleting the data (i.e., tuples) recorded in a relation (i.e., table)

1-2-3 Conceptual Design of Databases

Conceptual design of databases is the process of creating conceptual data models. For that, it is necessary to conduct **data analysis** at the beginning. The data analysis identifies the data that is required for the targeted operations where database is implemented, and then the meaning and relation of this data are analyzed and summarized. In addition, data items are standardized by using rules such as naming convention in order to eliminate data duplication. **Metadata** such as names, meanings, or attributes of data items are recorded in the **data dictionary** for users in **DD/D (Data Dictionary/Directory)** to ensure that synonyms/homonyms do not occur. The conceptual data model is created on the basis of the results of data analysis. For the conceptual data model, it is common to use **E-R model** (i.e., **E-R diagram**) that represents the target content with two concepts of entity and relationship.

[Constituent elements of E-R model (E-R diagram)]

- **Entity**
It is an object that is managed and represented with a rectangle.
- **Relationship**
It is a relation between entity and entity, and it is represented with a rhombus.
- **Attribute**
It is a characteristic or property of an entity and a relationship, and it is represented

with an ellipse.

In E-R model, it is possible to represent **cardinality** (i.e., multiplicity) of the relationship. There are three types of cardinality: “one-to-one”, “one-to-many”, and “many-to-many”. For example, Figure 4-4 in the E-R model shows a “many-to-many” relation where one instructor delivers a lecture to multiple students, and one student attends lectures of many instructors.

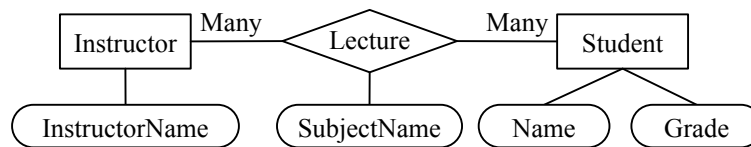


Figure 4-4 Example of E-R model (E-R diagram) (1)

When cardinality in E-R model is represented, “Many” is also represented as “*”, “M”, or “N”. Another notation method is to directly connect two entities with a line as shown in Figure 4-5. In this case, an arrow on the line means “Many.” Figure 4-5 shows that multiple types of products are purchased from a wholesaler, and products are always purchased from the same wholesaler.

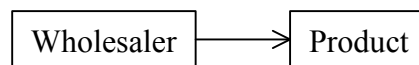


Figure 4-5 Example of E-R model (E-R diagram) (2)

1-2-4 Logical Design of Databases

Logical design of databases is the process of creating logical data models in consideration of the databases to be implemented. (Files/records/fields (items) are defined in a structured database, while tables/tuples/attributes are defined in a relational database.)

In the case of a relational database, **table design** is performed to decide which attributes of the table should be used for managing the data items represented in conceptual design. At that time, **primary key** and **foreign key** of each table should also be considered.

- **Primary key**

It is a field (or a combination of fields) that uniquely identifies each tuple (i.e., record) in a table. (**Composite key** made of multiple fields is also acceptable.) For primary key, there are constraints, such as **non-NULL constraint** that does not allow **NULL** (null value) and **unique constraint** that does not allow duplication of values in a table.

- **Foreign key**

It is a field (or a combination of fields) for referencing a tuple (i.e., record) whose values match the primary key of another table. For a foreign key, it is possible to define **referential constraint**, which requires that the primary key of the table to be referenced must have a tuple (i.e., record) having the same value.

Note: Non-NULL constraint, unique constraint, and referential constraint are also called **consistency constraint** for always maintaining the data in the correct status.

In a general table design, required fields are listed in a table in the **un-normalized form** (UNF). Next, **data normalization** is performed in order to design efficient table. **Functional dependency** is the property where if a certain attribute is decided, other attributes is uniquely decided.

[Data normalization procedure]

- 1) **First normalization** (Deliverable: Table in the first normal form)
Iterative fields or derived fields (i.e., fields determined through calculation) are eliminated.
- 2) **Second normalization** (Deliverable: Table in the second normal form)
When the primary key is a composite key, fields that are dependent on some of the fields constituting the primary key are split into a separate table. (This dependency is called **partial functional dependency**.)
- 3) **Third normalization** (Deliverable: Table in the third normal form)
Fields dependent on the fields other than the primary key are split into a separate table. (This dependency is called **transitive functional dependency**.)

By proceeding up to the third normalization as per the procedure, it is possible to design a table where all fields are dependent on the primary key. (This is called **full functional dependency**.) However, there are cases where the third normal form is not the best option. (For example, it may be more efficient to retain frequently used derived fields.) Therefore, it is necessary to consider including the operational aspects.

Example: Provide the third normal form obtained by normalizing the following table in un-normalized form. Underlined fields are primary keys, and { } indicates repetitive fields.

Voucher (VoucherNumber, Date, CustomerNumber, CustomerName, TotalAmount,
{ProductNumber, ProductName, UnitPrice, Quantity, Amount})

- 1) First normalization: Eliminate repetitive fields (ProductNumber,

ProductName, UnitPrice, Quantity, Amount) and derived fields (TotalAmount, Amount)

Voucher (VoucherNumber, Date, CustomerNumber, CustomerName, ProductNumber, ProductName, UnitPrice, Quantity)

- 2) Second normalization: Split partial functional dependency ({Date, CustomerNumber, CustomerName} dependent on VoucherNumber, {ProductName, UnitPrice} dependent on ProductNumber)

Voucher (VoucherNumber, Date, CustomerNumber, CustomerName)

Details (VoucherNumber, ProductNumber, Quantity)

Product (ProductNumber, ProductName, UnitPrice)

- 3) Third normalization: Split transitive functional dependency (CustomerName dependent on CustomerNumber)

Voucher (VoucherNumber, Date, CustomerNumber)

Details (VoucherNumber, ProductNumber, Quantity)

Product (ProductNumber, ProductName, UnitPrice)

Customer (CustomerNumber, CustomerName)

1-2-5 Physical Design of Databases

Physical design of databases is the process of creating physical data models in consideration of the database product (e.g., database software) to be implemented. Here, the data type that is most appropriate for each attribute is selected from the available data types in the database product, and memory format or mapping on hard disk is reviewed. In addition, required disk capacity is estimated from the anticipated data volume, and performance evaluation, such as processing efficiency or access efficiency, is conducted. At that time, it is also necessary to consider defining **index** (i.e., search key) for improving access efficiency. When the size of the database is large, the search efficiency can be significantly increased by defining index for the attributes that are frequently used in search operation. However, it should be noted that disk capacity for recording index is required and the process of updating the attributes of index requires more time than the usual processing of attributes.

1 - 3 DBMS (DataBase Management System)

DBMS (DataBase Management System) is software that manages databases for effectively using the databases. The person who manages the databases by using software, such as DBMS,

is called the DBA (DataBase Administrator).

1-3-1 Database Definition Function

Database definition function is a function that defines **schema** (definition and description concerning logical structure, storage structure, and physical structure of databases). Schema is mostly defined by using the following **three-schema architecture**.

Schema name	Meaning	Language used
Conceptual schema (Schema)	Defines the logical structure and name of the overall database.	DDL (Data Definition Language)
External schema (Subschema)	Defines the database as seen from the user's viewpoint (only the required part of database)	DDL (Data Definition Language) DML (Data Manipulation Language)
Internal schema (Storage schema)	Defines the physical structure of databases (storage area and organization method)	DSDL (Data Storage Definition Language)

1-3-2 Database Manipulation Function

Database manipulation function is a function that offers the usage environment of database language (e.g., **SQL** in the case of a relational database) that is used in definition and operation of data. The following are the execution methods of database manipulation language (e.g., **SQL**) provided by DBMS.

- **Host language system**

This is a method of executing SQL statements by using a high level language. It includes **embedded SQLs** that describe SQL statements in a program and **module language system** that creates SQL statements as external modules and calls them from the program.

- **Independent language system**

This method independently executes SQL statements. It includes **interactive SQL method** where SQL statements are entered from the command line and **command driven method** where the registered SQL statements are called and executed with commands having parameters. In addition, some DBMS for PCs allow easy operation

of databases where, by selecting from the predetermined forms and entering conditions, the process is executed as a query (i.e., a command to be converted into SQL statements for operating database).

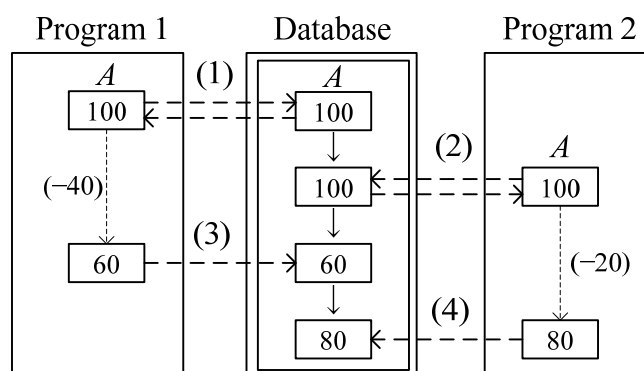
1-3-3 Database Control Function

Database control function is a function for improving reliability and security of the data recorded in the databases.

(1) Maintenance function

Maintenance function is a function for maintaining integrity of data (i.e., data integrity). The following **double update** is the typical example where integrity of data is lost. (As a result, inconsistency occurs.)

[Example of double update]



- 1) Program 1 references Data A (100) in the database.
- 2) Program 2 references Data A (100) in the database.
- 3) Data A is updated with the content (60) that is modified in Program 1.
- 4) Data A is updated with the content (80) that is modified in Program 2.
→ Updated content is overwritten, and update of Program 1 becomes invalid.

Exclusive control is designed for resolving such a problem. There are several methods in exclusive control, and the typical method is **lock system**.

The lock system locks the database record that is referenced and restricts further referencing until the process is completed. When this method is used, there is no risk of data being referenced by another program during the update, therefore, the integrity of data increases.

The following two types of locks are used separately in the lock system.

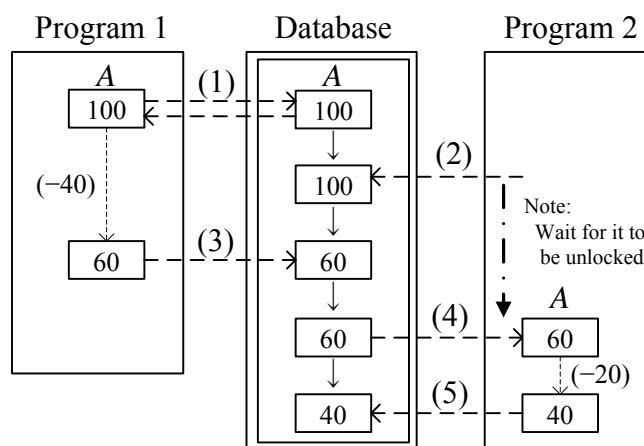
- **Shared lock**

This lock is mainly used when data is read. In shared lock, only reading is permitted for the users other than the user who applied the lock.

- **Exclusive lock**

This lock is mainly used when data is updated. Neither reading nor writing is permitted for the users other than the user who applied the lock.

[Example of exclusive control (lock system)]



- 1) Program 1 references Data A (100) in the database.
→ Apply exclusive lock to Data A.
- 2) Program 2 tries to reference Data A in the database. However, since Data A is locked, Program 2 waits for it to be unlocked.
- 3) Data A is updated with the content (60) that is modified in Program 1.
→ Release the lock of Data A. (Data A is unlocked.)
- 4) After the release of the lock is confirmed, Program 2 references Data A (60).
→ Apply exclusive lock to Data A.
- 5) Data A is updated with the content (40) that is modified in Program 2.
→ Release the lock of Data A. (Data A is unlocked.)

However, the problem called **deadlock** may occur in the lock system. Deadlock is the phenomenon where multiple programs are simultaneously in the waiting state because of the lock and their executions completely stop. It is difficult to fully avoid the deadlock. But, by reducing the unit (granularity) of applying the lock, the possibility of the occurrence of deadlock can be reduced.

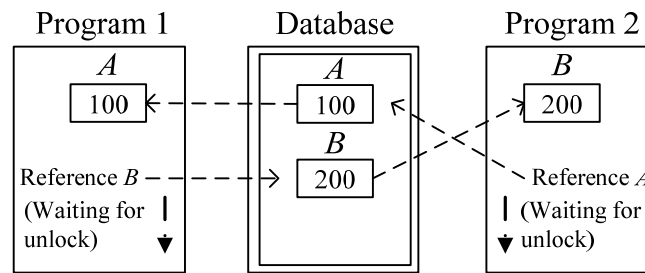


Figure 4-6 Image of deadlock

Other methods of exclusive control include **semaphore system**, which manages databases as resources by means of semaphore.

(2) Security protection function

Security protection function is a function for maintaining security of data (i.e., data security). Critical data and highly confidential data are stored in databases. Therefore, data security protection becomes necessary. The following security protection functions are provided by DBMS.

- **Encryption**

In this method, contents that are recorded in a database are encrypted. Even if the information of the database is leaked to a third party, that party does not know the meaning of the information. By distributing the software for decryption to only valid users, the security protection function can be increased in a relatively easy manner. However, problems, such as “decryption is not absolutely impossible” and “there is a risk of leakage of encryption/decryption software,” may occur. Therefore, this method is mostly used with other methods.

- **Access rights setting**

In this method, processes (i.e., accesses) are allowed for the applicable database is defined in advance. Detailed access rights can be set for each user, such as “User A can refer to data, but cannot update it” and “User B can refer to and update data, but cannot delete it.” The following is the summary of various permissions with respect to database and access rights that can be set in general.

	“Read” right	“Insert” right	“Delete” right	“Update” right
Right to “connect”	Yes	Yes	Yes	Yes
Right to “search”	Yes	Yes	Yes	Yes
Right to make a “new registration”		Yes		
Right to “delete”			Yes	

Right to “update”				Yes
-------------------	--	--	--	-----

- **Password setting**

This method uses a user ID and a password to check whether the user of the database has the right to use the database or not. It is useful for preventing unauthorized use from outside. Even when the access right is set, it is mandatory to set a password in order to prevent the unauthorized use of user ID.

- **Registration in journal file (log file)**

This method records the status of use (date, account, details of use) of the database and checks whether there has been any unauthorized use or not. While the journal file does not offer immediate security protection, it is useful because unauthorized users can be identified at a later date. It is mostly used in a supplementary sense.

(3) Failure recovery function

Failure recovery function is a function that restores databases at the occurrence of a failure. Failure recovery uses files, such as **backup file** where a database at a particular time is copied as it is, and **journal file** (i.e., **log file**) where an update process performed on the database is recorded. Journal records are created in the memory and are written in a file at the timing of **commit** (i.e., process of finalizing the update process) which is performed when the database update process for each transaction is completed. A log is prepared by setting a **checkpoint** for a certain time interval (or a certain processing amount). The database being updated in the memory is recorded in the **checkpoint file**. With this, it is possible to recover from the checkpoint time when a failure has occurred.

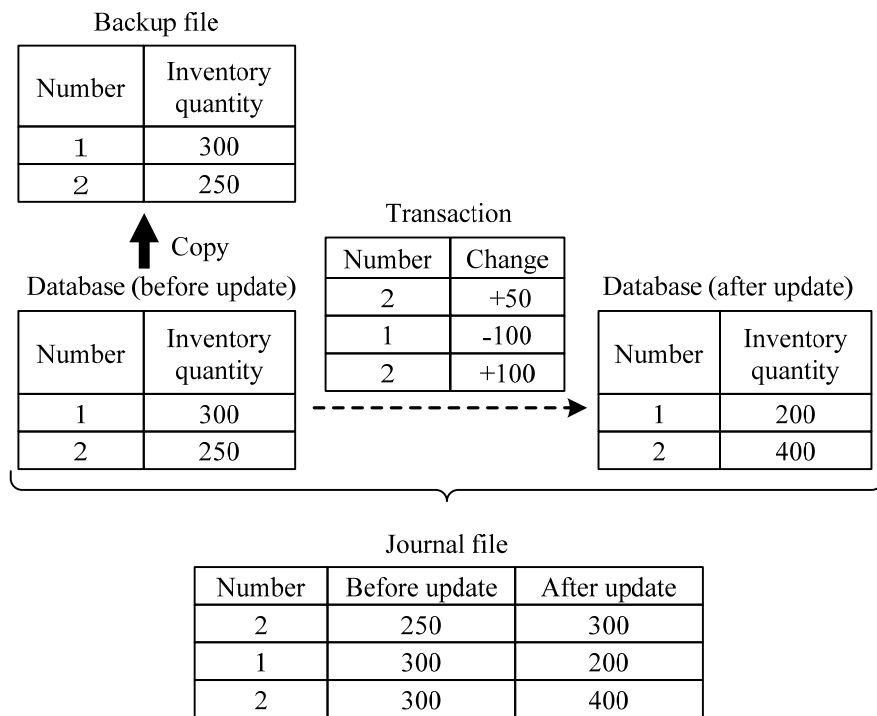


Figure 4-7 Backup file and journal file

The following are two types of recovery processes performed for failure recovery.

- **Rollforward**

This recovery technique is mainly used for physical failure of storage media. Contents of the backup file (or the checkpoint file) are updated on the basis of information (also known as “redo” information) after the update of the journal file, in order to restore the contents as the database at the time the failure occurred.

- **Rollback**

This recovery technique is mainly used for logical faults, such as transaction error. The database just after the disabling process is restored to the status before the disabling process on the basis of the information (also known as “undo” information) before the update of the journal file. Rollback is also used in the sense of canceling the update process that has not been committed yet.

The recovery process is also performed for each transaction. In this case, transactions committed after a checkpoint are recovered to the status after completion of commit with rollforward. On the other hand, for the transactions that are running at the time the failure occurred, processes that are not committed are canceled (i.e., rolled back) and transactions are executed again after the database is reverted to the status before the process started.

In the case of Figure 4-8, T1 and T2 committed after the checkpoint are recovered with rollforward from the checkpoint time. Meanwhile, T3, which was being updated at the time the

failure occurred, is executed again when the database is reverted to the status before the process started with rollback from the checkpoint time.

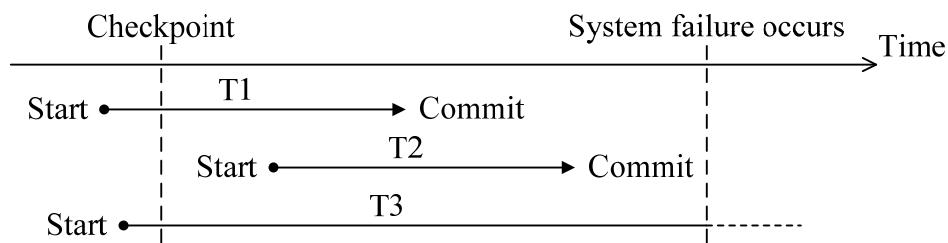


Figure 4-8 Failure recovery for each transaction

The following two methods are available for restarting the database.

- **Warm start method**

This method restarts the database, while the status of memory is maintained, by using software reset where the power supply is not turned off. Without initializing the database, it is restored to the status just before it was restarted by using the checkpoint file or journal file.

- **Cold start method**

This method restarts the database, after the status of memory is completely cleared, by using hardware reset where the power supply is turned off. After the database is restarted, it is restored to the status just before it was restarted by using the latest backup file and journal file.

A repetitive update (i.e., addition or deletion) of the database may result in a wasteful storage area that is not reused, which may decrease the access efficiency. In such a case, **reorganization** that optimizes the database is also one of the failure recovery functions.

[Transaction management of database]

Transaction processing with respect to the database is required to have **ACID characteristics**.

- **Atomicity**: The process completes by commit or rollback.
- **Consistency**: Contradiction (i.e., inconsistency of data) does not occur before and after the process.
- **Isolation**: Multiple transactions do not mutually interfere.
- **Durability**: Results after commit are continuously retained even after the process.

2 SQL

SQL (Structured Query Language) is a database language for using a relational database. SQL consists of DDL (Data Definition Language) that defines data, DML (Data Manipulation Language) that operates data, and others. The basic method of using SQL is explained with the following tables.

<Student>

<u>StudentNumber</u>	Name	Gender
6724	Kazuki Yamamoto	Male
6725	Jyugo Motoyama	Male
6816	Mone Yamada	Female
6817	Chiyo Yamamoto	Female

Alphanumeric
characters

Alphabetical
characters

Alphabetical
characters

<Subject>

<u>SubjectNumber</u>	SubjectName
K11	English I
K12	English II
K21	Mathematics

Alphanumeric
characters

Alphabetical
characters

<Score>

<u>StudentNumber</u>	<u>SubjectNumber</u>	Score	ExaminationDate
6724	K11	65	20XX-10-20
6724	K21	85	20XX-10-21
6725	K21	60	20XX-10-21
6817	K11	85	20XX-10-20
6817	K12	90	20XX-10-20
6817	K21	95	20XX-10-21

Alphanumeric
characters

Alphanumeric
characters

Integer

Date

Note:

Fields underlined
in each table show
the primary key.

2 - 1 Data Definition

Data definition refers to defining the database, table (i.e., base table), view (i.e., virtual table), and so on.

2-1-1 Definition of Database

CREATE DATABASE statement is used for defining database.

```
CREATE DATABASE name
```

ex01: Define the database “PerformanceManagementDB”.

```
CREATE DATABASE PerformanceManagementDB
```

2-1-2 Definition of Tables

CREATE TABLE statement is used for defining a table.

```
CREATE TABLE name
```

```
(Column name 1, Data type 1, Column name 2, Data type 2, ... )
```

- The following are the main data types that can be defined.

Data type	Definition name	Meaning
Character type	CHAR	Character of the specified length (1 byte)
	NCHAR	Character of the specified length (2 bytes)
Numeric type	INT	Integer having precision defined by the processing system.
	DEC	Numerical values having integer part and decimal part of the specified number of digits.
Date type	DATE	Year-Month-Date

- **PRIMARY KEY** : Declaration of primary key (**Primary key constraint**: non-NULL constraint + unique constraint)
- **FOREIGN KEY** : Declaration of foreign key (**referential constraint**)
- **NOT NULL** : **non-NULL constraint** (NULL is not allowed as occurrence)
- **UNIQUE** : **unique constraint** (Duplication of occurrence in the table is not allowed)
- **CHECK** : **check constraint** (specifies the conditions of occurrence)

Note: It is called **column constraint** when it is simultaneously performed with column definition and **table constraint** when it is performed at the end.

ex02: Define table “Score”.

```
CREATE TABLE Score
( StudentNumber    CHAR(4),
  SubjectNumber    CHAR(3),
  Score            INT  CHECK (Score >= 0),
  ExaminationDate  DATE NOT NULL,
  PRIMARY KEY (StudentNumber, SubjectNumber),
  FOREIGN KEY (StudentNumber) REFERENCES Student(StudentNumber),
  FOREIGN KEY (SubjectNumber) REFERENCES Subject(SubjectNumber)
)
```

Notes:

- * A check constraint (as column constraint) is set for the score to be 0 and above.
- * Non-NULL constraint (as column constraint) is set for ExaminationDate.
- * Primary key constraint (as table constraint) is set for (StudentNumber, SubjectNumber).
- * For StudentNumber, a referential constraint (as table constraint) is set so that StudentNumber in Student is referenced as a foreign key.
- * For SubjectNumber, a referential constraint (as table constraint) is set so that SubjectNumber in Subject is referenced as a foreign key.

Moreover, use the following ALTER TABLE statement when the configuration of the defined tables (i.e., database) is changed (or redefined).

ALTER TABLE name Details of redefinition

Note: In the underlined part, write **ADD** when attribute is added, write **MODIFY** when data type is changed, and so on.

2-1-3 Definition of View

View refers to the table that is virtually set from the real tables and corresponds to external schema. (View is also referred to as **virtual table**, and table is also referred to as **base table**.) View can record attributes of a single table and new attributes that are created from multiple tables. Users can operate the defined views in the same way as normal tables, and therefore, restrict the scope of use of a database, which can help in data security protection and maintenance.

CREATE VIEW statement is used for defining a view.

CREATE VIEW Name **AS** **SELECT ...**

Note: In the underlined part, write the SELECT statement for extracting data. (Details are explained in 2-2.)

ex03: Extract StudentNumber and Name from the “Student” table and define the view “Name”.

CREATE VIEW Name **AS** **SELECT StudentNumber, Name** **FROM** Student

Note: Underlined part is the SELECT statement that extracts StudentNumber and Name from the Student table.

View defines only a virtual table and does not create any new tables. (It can simply be considered that the shaded portion in Figure 4-9 is made invisible.) Therefore, updating a view would also reflect the updated results in the original table. Because of that, there is a constraint that the views where tuples and the original table are not in a 1:1 relationship cannot be updated. Such a view (e.g. view that uses the set function) is described later.

StudentNumber	Name	Gender
6724	Kazuki Yamamoto	Male
6725	Jyugo Motoyama	Male
6816	Mone Yamada	Female
6817	Chiyo Yamamoto	Female

Figure 4-9 Concept of view

2-1-4 Definition of Access Right

Access right is the right (i.e., **processing privilege**) for each user to use a database. Setting access right is useful for data security protection.

GRANT statement is used for defining access right.

GRANT Privilege 1, Privilege 2, ... **ON** Table Name **TO** Identifier

- The following access rights can be defined.

Type of Privilege	Meaning
SELECT	Permission to refer to the database
INSERT	Permission to add (i.e., insert) data in the database
DELETE	Permission to delete data from the database
UPDATE	Permission to update data in the database
REFERENCES	Permission to redefine the database

ALL PRIVILEGES	All permission related to the database
-----------------------	--

Note: Identifier uniquely specifies the intended person who is granted the permission.

ex04: Set the permission to refer and update the “Student” table in the identifier “EducationAffairsOffice”.

```
GRANT SELECT, UPDATE ON Student TO EducationAffairsOffice
```

REVOKE statement is used for canceling the defined (i.e., granted) permission.

```
REVOKE Privilege 1, Privilege 2, ... ON Table name TO Identifier
```

2-1-5 Data Storage

Data storage is the process of inserting data in a table. (While it should be included in data manipulation, it is explained in data definition as preparation for using the database.) There are two methods of storing data in a table. Generally, method 1) is suitable for small amounts of data while method 2) is more efficient for large amounts of data.

- 1) Storing data in units of tuples (i.e., records)

INSERT statement is used for storing data in units of tuples (i.e., records).

```
INSERT INTO Table name VALUES (Data 1, Data 2, ...)
```

ex05: Store tuple (K11, English I) in the “Subject” table.

```
INSERT INTO Subject VALUES ('K11', 'English I')
```

- 2) Using data storage program

A data storage program is prepared in the host language system beforehand, or the data storage program that is available in the utility program is used.

2 - 2 Data Manipulation

Data reference (i.e., extraction) from the database is the most widely-used data manipulation. In SQL, use the SELECT statement for referring to data. The explanation provided here is mainly focused on the method of using the SELECT statement.

2-2-1 Reference Without Specifying Conditions

Reference without specifying conditions corresponds to the relational operation “Projection” that extracts the specified attributes. Use the following SELECT statement in reference without specifying conditions.

```
SELECT Column name 1, Column name 2, ... FROM Table name
```

Notes:

1. If “*” is specified as column name, all columns are extracted.
2. When **DISTINCT** is specified in a column name, a duplication is eliminated. (The same results are not be displayed.)

ex06: Extract all columns from the “Student” table.

```
SELECT * FROM Student
```

<Execution results of ex06>

StudentNumber	Name	Gender
6724	Kazuki Yamamoto	Male
6725	Jyugo Motoyama	Male
6816	Mone Yamada	Female
6817	Chiyo Yamamoto	Female

ex07: Extract Gender from the “Student” table. (Projection).

```
SELECT Gender FROM Student
```

ex08: Extract Gender from the “Student” table after duplication is eliminated. (Projection)

```
SELECT DISTINCT Gender FROM Student
```

<Execution results of ex07>

Gender
Male
Male
Female
Female

<Execution results of ex08>

Gender
Male
Female

2-2-2 Reference With Specifying Conditions

Reference with specifying conditions corresponds to the relational operation “Selection” that extracts the tuples that satisfy the specified conditions. Use the following SELECT statement in reference with specifying conditions.

SELECT Column name 1, Column name 2, ... **FROM** Table name **WHERE** Extraction conditions

Notes: 1. Tuples that satisfy the conditions that are described in the WHERE clause are extracted.

2. The following are the main operators and notations used in extraction conditions.

<Comparison operators>

Notation	Example of use	Meaning
=	$A = B$	A is equal to B .
<>	$A <> B$	A is not equal to B . ($A \neq B$)
<	$A < B$	A is smaller than B .
<=	$A <= B$	A is equal to or smaller than B . ($A \leq B$)
>	$A > B$	A is greater than B .
>=	$A >= B$	A is equal to or greater than B . ($A \geq B$)

<Logical operators>

Notation	Meaning
AND	True if both the conditions are satisfied
OR	True if either of the conditions is satisfied
NOT	False if the original condition is true, and true if it is false (negation)

<Others (specifying special conditions)>

Notation	Meaning
BETWEEN A AND B	Specifies the range of values (A or more, and B or less)
LIKE ...	Specifies the pattern of string (wild card specification) (%: Characters above 0 characters, _: 1 character)
IS NULL	Judgment of NULL

ex09: From the “Student” table, extract tuples where Gender is 'Female' (Selection).

```
SELECT * FROM Student WHERE Gender = 'Female'
```

ex10: From the “Subject” table, extract subject names other than the SubjectName 'Mathematics' (Selection, Projection).

```
SELECT SubjectName FROM Subject WHERE SubjectName <> 'Mathematics'
```

<Execution results of ex09>

StudentNumber	Name	Gender
6816	Mone Yamada	Female
6817	Chiyo Yamamoto	Female

<Execution results of ex10>

SubjectName
English I
English II

ex11: From the “Score” table, extract StudentNumber, SubjectNumber, and Score where SubjectNumber contains '2' and Score is between 80 and 90.

```
SELECT StudentNumber, SubjectNumber, Score FROM Score
WHERE (SubjectNumber LIKE '%2%')
AND (Score BETWEEN 80 AND 90)
```

ex12: From the “Score” table, extract StudentNumber, SubjectNumber, and Score where SubjectNumber contains '2' or Score is between 80 and 90.

```
SELECT StudentNumber, SubjectNumber, Score FROM Score
WHERE (SubjectNumber LIKE '%2%')
OR (Score BETWEEN 80 AND 90)
```

<Execution results of ex11>

Student Number	Subject Number	Score
6724	K21	85
6817	K12	90

<Execution results of ex12>

Student Number	Subject Number	Score
6724	K21	85
6725	K21	60
6817	K11	85
6817	K12	90
6817	K21	95

ex13: From the “Score” table, among the tuples where the score is greater than 60, extract the tuples where the 2nd digit as counted from the left edge of StudentNumber is '7' or ExaminationDate is '20XX-10-21'.

```
SELECT * FROM Score
WHERE Score > 60
AND (StudentNumber LIKE '_7_ _' OR ExaminationDate = '20XX-10-21')
```

<Execution results of ex13>

StudentNumber	SubjectNumber	Score	ExaminationDate
6724	K11	65	20XX-10-20
6724	K21	85	20XX-10-21
6817	K21	95	20XX-10-21

Note: If () is not included, AND is evaluated first, and incorrect results are extracted as shown below.

StudentNumber	SubjectNumber	Score	ExaminationDate
6724	K11	65	20XX-10-20
6724	K21	85	20XX-10-21
6725	K21	60	20XX-10-21
6817	K21	95	20XX-10-21

2-2-3 Grouping of Data

Grouping of data refers to handling the tuples where a certain attribute has the same value in a consolidated manner, and functions used for this are referred to as **set functions** (or **aggregate functions**).

GROUP BY clause is used for grouping of data.

```
SELECT Extraction items FROM Table name GROUP BY Grouping column name
```

- In extraction items, grouping column names, set functions, and constants can be described.
- The following are the typical set functions.

Set function	Meaning
SUM	Determine the sum of group.
AVG	Determine the average of group.
MIN	Determine the minimum value of group.
MAX	Determine the maximum value of group.

COUNT	Count the number of records (i.e., number of tuples) in the group. * ... Count all tuples. DISTINCT Column name ... Count the tuples where the value of the column does not overlap.
--------------	---

- **AS** : A new column name can be specified in the column determined by using set functions.
- **HAVING** : The extraction condition that uses set functions can be written.

ex14: From the “Score” table, determine and extract average score of each subject.

```
SELECT SubjectNumber, AVG(Score) FROM Score
GROUP BY SubjectNumber
```

ex15: From the “Score” table, determine and extract the number of subjects for which the respective student took an examination.

```
SELECT StudentNumber, COUNT(*) AS NumberOfSubjects FROM Score
GROUP BY StudentNumber
```

ex16: From the “Score” table, determine and extract the number of days for which the respective student took an examination.

```
SELECT StudentNumber, COUNT(DISTINCT ExaminationDate) AS NumberOfDays
FROM Score GROUP BY StudentNumber
```

< Execution results of ex14 >

SubjectNumber	AVG (Score)
K11	75
K12	90
K21	80

< Execution results of ex15 >

StudentNumber	NumberOfSubjects
6724	2
6725	1
6817	3

< Execution results of ex16 >

StudentNumber	NumberOfDays
6724	2
6725	1
6817	2

ex17: From the “Score” table, extract students having TotalScore (total of score) of 150 or more.

```
SELECT StudentNumber, SUM(Score) AS TotalScore FROM Score
GROUP BY StudentNumber HAVING SUM(Score) >= 150
```

<Execution results of ex17>

StudentNumber	TotalScore
6724	150
6817	270

2-2-4 Sorting of Data

Sorting of data refers to rearranging the extraction results in the specified order of a particular attribute. (When there are no sorting instructions, data is extracted in the order in which it is recorded in the original table.)

ORDER BY clause is used for sorting data.

```
SELECT Column name 1, Column name 2, ... FROM Table name
ORDER BY Column name to be sorted Sorting order
```

- The following two types of sorting orders can be specified.

Sorting order	Meaning
ASC	Sort in the ascending order. (This is the default value when sorting order is omitted.)
DESC	Sort in the descending order.

- For records having the same column value, original recording order is generally retained.
- The column name to be sorted can also be specified by indicating how many columns from the left it is.

ex18: Sort and extract the “Score” table in the descending order of score.

```
SELECT * FROM Score ORDER BY Score DESC
```

ex19: From the “Score” table, determine the highest score for each student and extract in the ascending order.

```
SELECT StudentNumber, MAX(Score) FROM Score
GROUP BY StudentNumber ORDER BY 2
```

< Execution results of ex18 >

Student Number	Subject Number	Score	ExaminationDate
6817	K21	95	20XX-10-21
6817	K12	90	20XX-10-20
6724	K21	85	20XX-10-21
6817	K11	85	20XX-10-20
6724	K11	65	20XX-10-20
6725	K21	60	20XX-10-21

< Execution results of ex19 >

Student Number	MAX(Score)
6725	60
6724	85
6817	95

2-2-5 Joining the Tables

Joining the tables means combining two or more tables into one table. There are a few methods of joining. However, the easiest method is explained by using the following two tables.

< α >

X	Y
X1	Y2
X2	Y1

< β >

Y	Z
Y1	Z1
Y2	Z2

When two or more tables are used, list all tables used in the FROM clause. With this, the combinations (i.e., **direct product**) of all tuples of the specified tables are created. This method of joining, which takes the direct product of two or more tables, is called **cross join**.

```
[Cross join] SELECT * FROM  $\alpha$ ,  $\beta$ 
```

When two or more tables are joined, values of the columns having common meaning mostly combine the same tuples. Therefore, with the condition of the WHERE clause, extract the tuples having the same values of column from the results of cross join. (Joining the tuples having the same value of the corresponding column is called **equijoin**.)

```
[Equijoin] SELECT * FROM  $\alpha$ ,  $\beta$  WHERE  $\alpha.Y = \beta.Y$ 
```

<Execution results of cross join>

$\alpha.X$	$\alpha.Y$	$\beta.Y$	$\beta.Z$
X1	Y2	Y1	Z1
X1	Y2	Y2	Z2
X2	Y1	Y1	Z1
X2	Y1	Y2	Z2

<Execution results of equijoin>

$\alpha.X$	$\alpha.Y$	$\beta.Y$	$\beta.Z$
X1	Y2	Y2	Z2
X2	Y1	Y1	Z1

ex20: From the “Score” table and the “Subject” table, extract StudentNumber, SubjectName for which the students took an examination, and Score.

```
SELECT StudentNumber, SubjectName, Score FROM Score, Subject
WHERE Score.SubjectNumber = Subject.SubjectNumber
```

<Execution results of ex20>

Student Number	SubjectName	Score
6724	English I	65
6724	Mathematics	85
6725	Mathematics	60
6817	English I	85
6817	English II	90
6817	Mathematics	95

Note:

When columns that are common in two tables are extracted, it is necessary to clearly identify the table to which the rows belong.

Example: SELECT Score. SubjectNumber, ...

When the table with FROM clause is specified, by adding **correlation name**, SQL statement in ex20 can be simplified as follows:

```
SELECT StudentNumber, SubjectName, Score FROM Score X, Subject Y
WHERE X.SubjectNumber = Y.SubjectNumber
```

One of the methods of joining tables uses JOIN specification in the FROM clause. With JOIN specification, the following three joining operations can be achieved.

- **CROSS JOIN (Cross join)**

It specifies direct product of multiple tables. (The result is the same as listing the tables.)

- **INNER JOIN (Inner join)**

It specifies equijoin of tables with joining conditions.

```
SELECT StudentNumber, SubjectName, Score
FROM Score X INNER JOIN Subject Y
```

ON X.SubjectNumber = Y.SubjectNumber

- **OUTER JOIN (Outer join)**

It extracts all tuples of the priority table even if the joining conditions are not met.

LEFT OUTER JOIN : Table on the left is the priority table.

RIGHT OUTER JOIN: Table on the right is the priority table.

2-2-6 Sub Reference (Subquery)

Sub reference (subquery) refers to searching for another table on the basis of the search results of a certain table. SELECT statement that determines the conditions is called subquery statement, while the SELECT statement that searches for the table by using the results of subquery statement is called main query statement. Subquery is broadly classified into two types.

(1) Subquery statement that can be independently executed

This subquery statement is independently executed for getting the conditions of the main query statement. In this format, subquery statement inside the parentheses () is executed first, and main query statement is executed on the basis of its results.

SELECT Column name, ... **FROM** Table name

WHERE Column name Comparison term (subquery statement)

- Comparison term includes IN predicate, and ANY predicate and ALL predicate combined with comparison operator.

Predicate	Meaning
IN	Either of the results of subquery statement matches. (Same as “= ANY”)
ANY	Comparison operator holds true for either of the results of subquery statement.
ALL	Comparison operator holds true for all results of subquery statement.

Note: If the result of subquery statement is always one, a single comparison operation can also be used.

ex21: From the “Student” table and the “Score” table, extract the names where ExaminationDate is '20XX-10-20'.

```
SELECT Name FROM Student
WHERE StudentNumber IN ( SELECT StudentNumber FROM Score
                        WHERE ExaminationDate = '20XX-10-20' )
```

< Execution results of ex21 >

Name
Kazuki Yamamoto
Chiyo Yamamoto

Note: Results of the subquery statement have the same meaning as the following SQL statement.

```
SELECT Name FROM Student
WHERE StudentNumber IN ('6724', '6817',
                        '6817')
```

(2) Correlation subquery (subquery statement that cannot be executed independently)

Correlation subquery uses items of the table in the main query statement as a condition, and therefore, this subquery statement cannot be executed independently. In this case, tuples are fetched one by one from the table in the main query statement and the subquery statement is executed. On the basis of whether the result of the subquery statement is one row or more, whether to extract the concerned tuple or not is decided. EXISTS predicate is used in this decision.

ex22: From the “Student” table and the “Score” table, extract the names where ExaminationDate is '20XX-10-20'.

```
SELECT Name FROM Student X
WHERE EXISTS ( SELECT * FROM Score Y
              WHERE X.StudentNumber = Y.StudentNumber
                AND ExaminationDate = '20XX-10-20' )
```

In the SQL statement of ex22, the subquery statement inside the parentheses () cannot be executed independently. (Correlation name X cannot be interpreted.) Therefore, tuples of the “Student” table are fetched one at a time, and the subquery statement is executed.

- Execute the subquery statement for the tuple of StudentNumber '6724'.

```
SELECT * FROM Score Y
WHERE '6724' = Y.StudentNumber AND ExaminationDate = '20XX-10-20'
```

< Execution results >

StudentNumber	SubjectNumber	Score	ExaminationDate
6724	K11	65	20XX-10-20

=> The result is one row or more, and therefore, the tuple of StudentNumber '6724' is extracted.

- Execute the subquery statement for the tuple of StudentNumber '6725'.

```
SELECT * FROM Score Y
WHERE '6725' = Y.StudentNumber AND ExaminationDate = '20XX-10-20'
```

<Execution results>

StudentNumber	SubjectNumber	Score	ExaminationDate
---------------	---------------	-------	-----------------

=> Since there is no single row in the result, the tuple of StudentNumber '6725' is not extracted.

In this manner, by executing the subquery statement by using StudentNumber of all tuples, tuples to be extracted are decided.

When this example is represented without using subquery, the SQL statement is described as below. Here, note that the DISTINCT specification is omitted and (Chiyo Yamamoto) is extracted twice.

```
SELECT DISTINCT Name FROM Student X, Score Y
WHERE X.StudentNumber = Y.StudentNumber
      AND ExaminationDate = '20XX-10-20'
```

In addition, IN predicate and EXISTS predicate can be used in combination with the logical operator NOT.

- **NOT IN:** True if only unequal subquery rows are found
- **NOT EXISTS:** True if tuple does not exist as the result of subquery

2-2-7 Other Methods of Using SQL

(1) Other data manipulation languages

- **INSERT statement:** This statement is used when a tuple is inserted (i.e., added) into the table.

```
INSERT INTO Table name VALUES ( Data 1, Data 2, ... )
```

- **UPDATE statement:** This statement is used when the data in the table is updated (i.e., changed).

```
UPDATE Table name SET Column name = Updated value
WHERE Update specification condition
```

- DELETE statement: This statement is used when a tuple is deleted from the table.

```
DELETE FROM Table name WHERE Delete specification condition
```

(2) Cursor

Cursor is used when data in the database is used in the host language system (i.e., embedded SQL). Usually, host language (i.e., programming language) handles data in units of records, while SQL handles data in units of sets. Therefore, the cursor acts as a bridge by fetching and transferring one row at a time.

- DECLARE CURSOR statement: This statement defines the derived table to be processed and assigns the cursor.

```
DECLARE Cursor name CURSOR FOR SELECT ...
```

Note: **Dynamic SQL** is also available that prepares the SELECT statement at the time of execution by using the PREPARE statement.

- OPEN statement: This statement starts the process of cursor.

```
OPEN Cursor name
```

- FETCH statement: This statement reads data from the position of cursor.

```
FETCH Cursor name INTO Variable name
```

- CLOSE statement: This statement terminates the process of cursor.

```
CLOSE Cursor name
```

3 Various Databases

In present day society, database technology is applied in various areas. In companies, databases are used in many systems, such as corporate accounting systems, inventory control systems, document management systems, and sales support systems. In this section, we will learn about the database technology used in these systems.

3 - 1 Distributed Database

Distributed database is a technology that handles databases distributed at multiple sites as if they were one database. Databases are, by nature, used for the purpose of consolidating data and managing it in an integrated manner. However, there are following problems concerning databases consolidated over a network: the fault of a database affects the entire system, the network becomes over crowded because of concentration of processes, and maintenance/management of the database becomes difficult. Therefore, the concept of distributed database was developed where databases are dispersed and constructed at multiple sites.

In the distributed database, by using the **RDA (Remote Database Access)** system, users can use the database without becoming aware of which database they are accessing. (This is called **transparency** of distributed database.) However, just like usual distributed processing systems, it suffers from the problem that operations, such as security management, take a lot of effort. In addition, it is possible that data may duplicated. Therefore, there is a risk that consistency of data is lost. In the distributed database, these problems are resolved with the following techniques.

- **Two-phase commitment** (Two-phase commit)

This is a commitment control method that executes the update process in the distributed database after the process is divided into two phases. It is suitable when an instantaneous update is required. The control method that instructs COMMIT or ROLLBACK without checking whether the update process can be performed or not is called **one-phase commitment**.

- 1) 1st phase

Check the possibility of update process for all databases.

- 2) 2nd phase

Finalize (commit) if all databases can be updated (i.e., if all databases respond with a positive responses), and cancel (i.e., rollback) if even only one database cannot be updated (i.e., if even only one database responds with a negative

response.).

- **Replication**

This method places a copy (i.e., replica) of the original database as the distributed database and reflects the update done in the original database into the replicated database at regular intervals (or specified times). It is suitable when an instantaneous update is not required.

3 - 2 Data Warehouse

Data warehouse is a company-wide integrated information system (multi-dimensional database) that expands the functions of databases and extracts the required decision-making information for the business strategy of the company. It is used in **OLAP (OnLine Analytical Processing)**, which extracts the data that is accumulated through **OLTP (OnLine Transaction Processing)** in the mission critical systems, builds a massive database for information analysis, and conducts multifactor analysis and forecasts of data,.

- **ETL (Extract/Transform/Load)**

It refers to extracting the raw data stored in the mission critical system, processing the raw data into an easy-to-use format with tasks such as name identification and standardization, and exporting it to a data warehouse. It is also software that supports this series of processes.

- **Data cleansing (Data cleaning)**

It refers to removing duplication and inconsistency of notations from the database. It is used in database optimization and ETL processing.

Data mining is the method to analyze data or regularity required for the management by using mathematical and statistical techniques such as OLAP. Mining, which is literally described as digging up data from databases, is useful in marketing strategies by analyzing a large amount of accumulated data and finding the potential needs of customers. For example, from a large amount of sales data, it extracts rules, such as “People who buy bread also buy milk simultaneously.”

- **Star schema**

It is a schema of a database for analysis. It places analysis values in a radial manner focusing on the target of analysis. Creating index for implementing star schema is one of the preparations required for data mining.

- **Cluster analysis**

It is an analytical technique that the similarity of data to be analyzed is quantitatively

(e.g., by using distance or extent of similarity) determined in order to group data. **Dendrogram** is used for showing the results of analysis.

This series of data management/analysis work from building a data warehouse to data mining is also called **data administration**.

In recent years, the amount of accumulated data has become so large that it is also referred to as **big data**. It has become difficult to handle so much data with the commercially available database software (i.e., DBMS). Therefore, in some cases, a small-scale data warehouse (i.e., **data mart**) is used, which is prepared by extracting only data required for a specific department (or business operations) from the data warehouse. Data mart is easy to use because data is refined as per the requirement and level of end user. However, there is a risk that it may overlook unforeseen regularity.

3 - 3 Other Related Techniques

(1) IRDS (Information Resource Dictionary System)

IRDS (Information Resource Dictionary System) is the system that registers and manages **metadata**, such as attribute, meaning, and storage location of data, in **DD/D (Data Dictionary)**. It is also used as a **repository** that manages source code or such other data, in an integrated manner in software development and maintenance.

(2) Special databases

- **Commercial database**

It is a database that contains independently collected information and that is offered to third parties on a chargeable basis for generating profits. In addition, in some commercial databases, by using a thesaurus (i.e., a dictionary where synonyms and equivalent words are systematically classified and arranged), **thesaurus search** can search with minimum oversight with respect to various representations.

- **OODB (Object-Oriented DataBase)**

It is a database that is managed with objects where data and processing (i.e., procedures) are integrated (i.e., encapsulated). Users can process the data by simply giving specific instructions (i.e., messages).

- **Multimedia database**

It is a database that can handle information (e.g., images, audio) in addition to characters and numeric values. It is very hard for users to use the database by being aware of various pieces of information such as images and audio, and therefore, the

concept of OODB (Object-Oriented DataBase) is used in most of the cases.

- **Hypertext database**

It is a network structured database that manages hypertext (e.g., a document that can freely search/display the relevant information by using keywords) used in the Internet. These days, a hypermedia database that can handle images and audio is also available.

- **XML database**

It is a database that can manage the document structure (e.g., tags) of XML as it is. This database uses the features of XML and has excellent flexibility and expandability.

(3) Database integration techniques

- **Data mapping**

It refers to creating a table (i.e., data map) that correlates data between different applications. It is used for correlating a database of different DBMSs.

- **JDBC (Java DataBase Connectivity)**

This API is used for accessing databases from Java. It allows the development of a highly general-purpose program that does not depend on DBMS.

Chapter 4 Exercises

Q1

Which of the following is a benefit that can be expected by installing a database system?

- a) Mitigation of code design activity
- b) Reduction of duplicate data
- c) High-speed data transfer
- d) Dynamic access to data

Q2

Which of the following is an appropriate explanation of the relational database?

- a) Data is viewed as a two-dimensional table to users. Relation between records is associated by using the values of items in each record.
- b) Relation between records is represented by using a one-to-many relationship of parent and child.
- c) Relation between records is represented with the network structure.
- d) A user recognizes and represents the logical layout of records.

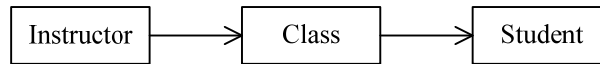
Q3

Which of the following is an appropriate explanation of “projection” that is one of relational operations?

- a) From the table, it selects tuples that satisfy the given conditions and creates a new table.
- b) From the table, it extracts only the specified attributes and creates a new table.
- c) It extracts tuples that commonly exist in two tables and creates a new table.
- d) From tuples in two tables, it joins the pairs of tuples that meet the conditions and creates a new table.

Q4

Which of the following is the appropriate interpretation of the E-R model shown below?



- a) One student is looked after by one instructor.
- b) One student belongs to multiple classes.
- c) One instructor looks after only one class.
- d) One class is looked after by two instructors, namely the homeroom teacher and deputy homeroom teacher.

Q5

Which of the following is the second normal form of the “SkillRecord” table? Here, the underlined parts show primary key, while { } shows iterative items.

SkillRecord

(EmployeeNumber, EmployeeName, {SkillCode, SkillName, SkillExperienceYears })

- a)

<u>EmployeeNumber</u>	EmployeeName	<u>SkillCode</u>	SkillName	SkillExperienceYears
-----------------------	--------------	------------------	-----------	----------------------
- b)

<u>EmployeeNumber</u>	EmployeeName	<u>SkillCode</u>	SkillExperienceYears
-----------------------	--------------	------------------	----------------------

<u>SkillCode</u>	SkillName
------------------	-----------
- c)

<u>EmployeeNumber</u>	<u>SkillCode</u>	SkillExperienceYears
-----------------------	------------------	----------------------

<u>EmployeeNumber</u>	EmployeeName
-----------------------	--------------

<u>SkillCode</u>	SkillName
------------------	-----------
- d)

<u>EmployeeNumber</u>	<u>SkillCode</u>
-----------------------	------------------

<u>EmployeeNumber</u>	EmployeeName	SkillExperienceYears
-----------------------	--------------	----------------------

<u>SkillCode</u>	SkillName
------------------	-----------

Q6

Which of the following is an appropriate description of the exclusive control function of a DBMS?

- a) It should be used during online updates, and needs not be used during updates with batch processing.
- b) It can only be performed for each relation (i.e., table).
- c) It prevents the loss of data integrity because of a lost update or such other problem.
- d) The purpose of the lock method which is one of the exclusive controls is to prevent deadlock.

Q7

When a failure occurs in database operation, which of the following is the failure recovery process that returns the status of database to the status before the start of the transaction?

- a) Reorganization
- b) Checkpoint
- c) Rollback
- d) Rollforward

Q8

Which of the following is the SQL statement that returns the largest value when it runs on the 'DeliveryRecord' table?

DeliveryRecord

ProductNumber	Date	Quantity
NP200	20XX-10-10	3
FP233	20XX-10-10	2
NP200	20XX-10-11	1
FP233	20XX-10-11	2

- a) `SELECT AVG(Quantity) FROM DeliveryRecord WHERE ProductNumber = 'NP200'`
- b) `SELECT COUNT(*) FROM DeliveryRecord`
- c) `SELECT MAX(Quantity) FROM DeliveryRecord`
- d) `SELECT SUM(Quantity) FROM DeliveryRecord WHERE Date = '20XX-10-11'`

Q9

Which of the following is the appropriate result of executing the SQL statement that is given below on the “Student” table and the “Department” table?

```
SELECT Name FROM Student, Department
WHERE Affiliation = DepartmentName AND Department.Address = 'Shinjuku'
```

Student

Name	Affiliation	Address
Tomoko Goda	Science	Shinjuku
Shunsuke Aoki	Engineering	Shibuya
Satoshi Kawauchi	Humanities	Shibuya
Yuko Sakaguchi	Economics	Shinjuku

Department

DepartmentName	Address
Science	Shinjuku
Engineering	Shinjuku
Humanities	Shibuya
Economics	Shibuya

- a)

Name
Tomoko Goda
- b)

Name
Tomoko Goda
Shunsuke Aoki
- c)

Name
Tomoko Goda
Yuko Sakaguchi
- d)

Name
Tomoko Goda
Shunsuke Aoki
Yuko Sakaguchi

Q10

Which of the following is the appropriate SQL statement for extracting Product information (ProductNumber, ProductName) of the products where InventoryQuantity is less than 100 units from the “Product” table and the “Inventory” table?

Product

ProductNumber	ProductName	UnitPrice
---------------	-------------	-----------

Inventory

ProductNumber	InventoryQuantity
---------------	-------------------

- a) `SELECT X.ProductNumber, ProductName FROM Product X, Inventory Y
WHERE X.ProductNumber = Y.ProductNumber OR InventoryQuantity < 100`
- b) `SELECT X.ProductNumber, ProductName FROM Product X, Inventory Y
WHERE InventoryQuantity < 100`
- c) `SELECT ProductNumber, ProductName FROM Product
WHERE EXISTS (SELECT * FROM Inventory WHERE InventoryQuantity < 100)`
- d) `SELECT ProductNumber, ProductName FROM Product
WHERE ProductNumber IN
(SELECT ProductNumber FROM Inventory
WHERE InventoryQuantity < 100)`

Q11

Which of the following is the SQL statement for defining the derived table when the table of the relational database is accessed from the program by using embedded SQL statements?

- a) CLOSE statement
- b) DECLARE CURSOR statement
- c) FETCH statement
- d) OPEN statement

Q12

In the distributed database system, which of the following is a method that updates the database after inquiry is made for multiple sites that perform a series of transaction processing whether they can be updated or not and if it is confirmed that all sites can be updated?

- a) Two-phase commitment
- b) Data cleansing
- c) Data mining
- d) Replication



Chapter 5

Network



1 Network Mechanism

In an information-driven society, the network supports the exchange of information (data communication). By using a network, it is possible to exchange information between computers. This section describes the basic mechanism of a network.

1 - 1 Types and Characteristics of Networks

Currently, the networks in use are broadly divided into two types.

- **LAN (Local Area Network)**

This is a network that is constructed by connecting the computers and peripheral equipment in a relatively narrow range (site), such as within a company or premise, with a private line.

- **WAN (Wide Area Network)**

This is a network that is laid down in a relatively broad region, such as the Internet and public telephone network.

Moreover, networks such as LAN and WAN may be either **wired** (i.e., connected through a cable) or **wireless** (i.e., connected through electromagnetic waves). Common examples of a wired connection include the use of a leased line and a public telephone network for data transmission, but **PLC (Power Line Communications)**, which makes use of a power line, is also included. On the other hand, examples of a wireless network include a **sensor network** in which data is collected by connecting several terminals (i.e., devices) on which a sensor is installed. A sensor network is a basic technique for implementing **ubiquitous computing** in which anyone can communicate anytime and anywhere.

When a WAN is to be constructed, it is common to use a communication service that is provided by a **telecommunications carrier**. For example, if a user uses an Internet connection service that is provided by an **ISP (Internet Service Provider)**, the user can easily construct a WAN to be connected to the Internet. At this time, if there are several telecommunications carriers who provide the same type of communication service, it becomes necessary to select the telecommunications carrier to sign a contract with depending on the usage charge and quality.

In most cases, any one of the following billing systems is used for the usage charge of the communication service.

- **Metered-rate system**

This is a system in which billing is decided on the basis of the usage time of the

communication service and the amount of transmission data.

- **Flat-rate system**

This is a system in which a fixed usage charge (e.g., a flat monthly fee) is set initially.

- **Two-part tariff system**

This is a system in which the basic charge is applicable up to a fixed usage, and beyond that, billing is decided according to the metered-rate system.

On the other hand, the indexes or indicators shown below are used for evaluating the quality of the communication service.

- **Line speed/line capacity**

Line speed refers to the transmission capacity of the line, and uses **bps (bit per second)** as the unit. The line speed, the amount of data to be transferred, and the transfer time have the relationship shown below.

$$\text{Transfer time (second)} = \text{Amount of data (bit)} \div \text{Line speed (bit/second)}$$

However, since it is not realistic to use 100% of the line transmission capacity, the **line utilization rate (%)** is generally used. In such a case, the line speed that can actually be used (effective line speed or **transfer speed**) is calculated as shown below.

$$\text{Effective line speed} = \text{Line speed} \times \text{Line utilization rate}$$

The transfer speed may be used in the same sense as line speed, and the line utilization rate may be used in the sense of what percentage of the line capacity is being used.

On the other hand, the **line capacity** is the overall acquired transmission capacity. For example, it may be assumed that in order to obtain a circuit capacity of 500 Mbps in a network, five lines with an effective line speed of 100 Mbps each are needed. (This method of examining the required number of lines is called **capacity planning**.)

- **Bit error rate** (Transmission error rate)

This is the probability that represents how many bit errors occurred in the number of total transmitted bits. For example, in the case of a line with a bit error rate of 10^{-4} (1/10,000), one bit error occurs when 10,000 bits are transmitted.

- **Traffic density/Lost-call rate**

The traffic density is the value that is obtained by dividing the total usage time of a call (request for network resources) by the unit time, and its unit is **erlang**. For example, when 30 calls are made in one hour and the average usage time per one call is 60 seconds, the traffic density is calculated as shown below.

$$\text{Total usage time of call} = 30 \text{ times} \times 60 \text{ seconds/one time} = 1,800 \text{ seconds}$$

$$\text{Traffic density} = 1,800 \text{ seconds} \div 3,600 \text{ seconds} = 0.5 \text{ erlang}$$

As the traffic density increases, the lost-call rate (i.e., the possibility that the calls that are made cannot use the network) increases. In order to keep the lost-call rate below a

fixed value, finding out the required number of lines and the upper limit of traffic density is also an example of capacity planning.

QoS (Quality of Service) is a quality assurance technique of such a communication service. The literal translation of QoS is **service quality**, but it is often used as a term that represents the technique of guaranteeing network quality, such as the line speed, for a specific communication service. In terms of QoS, the minimum line speed (i.e., communication speed) is secured through bandwidth control that secures an exclusive bandwidth (i.e., communication channel) beforehand, and priority control that adds a priority level to the transmission data. In contrast to a **guarantee-type** communication service that guarantees the minimum line speed by using QoS, the communication service that guarantees the maximum level of efforts for providing the upper limit of the line speed is called the **best effort-type** communication service.

1 - 2 Basic Configuration of a Network

In a system like an OLTP system, the basic configuration of the network when the host computer at the center is used from a remote terminal is as described below. Data communication can be broadly divided into the data transmission system that is used for transmitting data and the data processing system that is used for processing the received data.

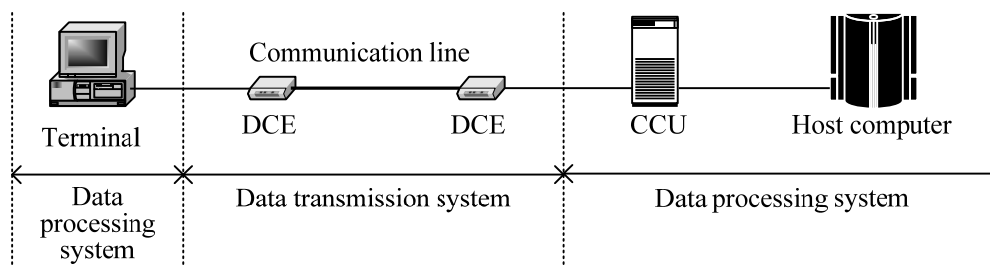


Figure 5-1 Basic configuration of a network

(1) DTE (Data Terminal Equipment)

This is the collective term for the terminal and host computer that constitute the data processing system.

- **CCU (Communication Control Unit)**

This is a unit to perform serial-parallel conversion (assembling/disassembling characters) of data during transmission, error control of data, and control of several lines. This unit isn't used as an independent device nowadays, but the function is implemented by a server or a FEP.

- **PBX (Private Branch eXchange)**

This is a unit that is used for connecting the terminal and telephones with a public line such as a telephone line. It is also used for connecting extensions within a company. This is also used in **CTI (Computer Telephony Integration)**, in which a telephone and a facsimile are integrated with the computer system to facilitate automated response, or to dispatch (or transfer) to optimum recipients.

- **MDF (Main Distribution Frame)**

This is a line concentrator that collectively manages the telephone lines and communication lines. Generally, it is a device that is used to connect to an external provider's line, and a line concentrator that is used internally on a per floor basis is called an **IDF (Intermediate Distribution Frame)**.

(2) **Communication line**

This is a physical transmission path for actually transmitting data.

- **Analog line**

This is a communication line for transmitting analog signals. An analog signal refers to a waveform signal, and a telephone line that transmits voice signals is regarded as an example of analog lines.

- **Digital line**

This is a communication line for transmitting digital signals. A digital signal is represented by either 0 or 1 and is suitable for data communication.

(3) **DCE (Data Circuit terminating Equipment)**

This is a collective term for the equipment that is used to connect the data terminal and communication line. It refers to the equipment for converting/transforming a data signal to be transmitted into a signal that is suitable for transmission.

- **MODEM (MOdulator/DEModulator)**

This is a type of DCE that is used when data is transmitted on an analog line. It is used to modulate digital signals into analog signals, or demodulate analog signals into digital signals.

- **DSU (Digital Service Unit)**

This is a type of DCE that is used when data is transmitted on a digital line. It converts the digital signals in a computer to a format that is suitable for transmission.

- **NCU (Network Control Unit)**

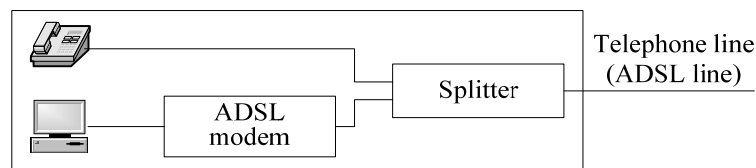
This is a type of DCE that is used when data is transmitted through a public telephone network, and has a dial function for connecting a line to the partner. Usually, it is provided as a built-in feature of the modem.

- **TA (Terminal Adapter)**

This is a type of DCE that is used when data is transmitted on an ISDN line. (ISDN is described later.) Currently, ISDN lines are hardly used.

- **ADSL modem**

This is a type of DCE that is used when data is transmitted on an ADSL line. (ADSL is described later.) When a telephone line is used as an ADSL line, a **splitter** that separates or combines the voice signals and data signals flowing through the same lines is required.



[Terminal interface]

A **terminal interface** refers to the standards of connectors (i.e., connecting parts) and the regulations concerning signals that are developed to transmit data between terminals.

The following series are ITU-T recommendations for the terminal interfaces defined by the **ITU-T (ITU-Telecommunication Standardization Sector)**, which is a subordinate organization of **ITU (International Telecommunication Union)**.

- V series: Interfaces between DTE and DCE for analog line (e.g., V.24)
- X series: Interfaces between DTE and DCE for digital line (e.g., X.25)
- I series: User/network interfaces for ISDN line (e.g., I.430)

1 - 3 Basic Techniques of a Network

1-3-1 Modulation Method

Modulation refers to the conversion of digital signals to analog signals, or analog signals to digital signals. (Returning converted signals to its original form is called **demodulation**.)

The types of methods used to modulate digital signals to analog signals with a modem in order to transmit data on an analog line are as shown below.

- **AM (Amplitude Modulation)**

1 and 0 of the digital signal are made to correspond to ON/OFF (presence or absence of waves) of the analog signal. Although this method is easy to implement, its disadvantage is that it is sensitive to noise.

- **FM (Frequency Modulation)**

0 and 1 of the digital signal are made to correspond to the low frequency and high frequency of the analog signal. This method is the next easiest to implement after the amplitude modulation method, and is also highly resistant to noise.

- **PM (Phase Modulation)**

The digital signal is represented on the basis of shifting of the waveform. It includes the binary phase shift keying method, in which 0 of the digital signal is represented by the standard waveform and 1 is represented by a 180-degree shift, and the quadrature phase shift keying method in which '00' is represented by the standard waveform, '01' is represented by a 90-degree shift, '10' by a 180-degree shift, and '11' by a 270-degree shift. The advantage of this method is that several bits can be transmitted through a single waveform by finely and minutely shifting the waveform. As a result, the **modulation speed** (unit: **baud**) that represents the number of times that modulation is performed in one second is not necessarily the same as the **signal speed** (or line speed) that represents the number of bits that can be transmitted in one second. The differential binary phase shift keying method, in which the waveform is reversed when 1 is transmitted, is also included.

On the other hand, the modulation method by which voice signals (i.e., analog signals) are encoded (i.e., converted to digital signals) includes **PCM (Pulse Code Modulation)**. In PCM, analog signals are modulated to digital signals through the procedures of sampling, quantization, and encoding.

[Encoding procedure according to PCM (excerpt from pp.170-171)]

- 1) **Sampling**

The analog signals to be encoded are sampled at regular intervals. This sampling interval is shown as the **sampling frequency**.

- 2) **Quantization**

The sampled sampling values are rounded to the nearest integer numbers.

- 3) **Encoding**

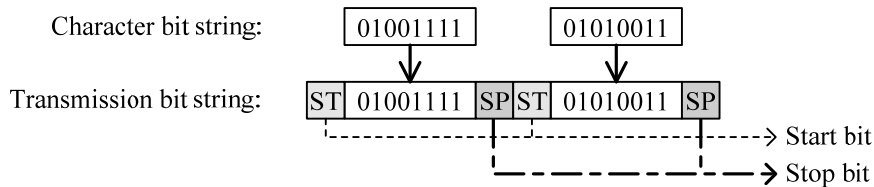
A quantized integer number is encoded by representing it as a binary number. (The number of bits used as a code is called the **quantization bit rate**.)

1-3-2 Synchronization Method

Synchronization refers to matching the timing at the transmitting side and the receiving side. There are three typical kinds of synchronization methods as shown below.

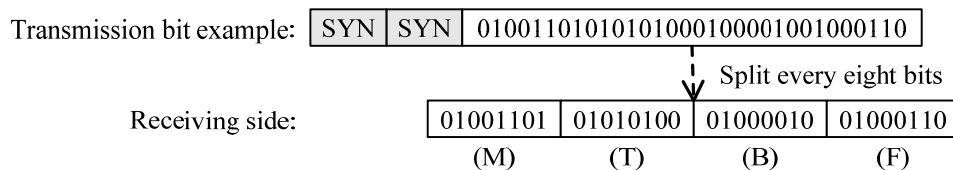
- **Start-stop synchronization method (Bit synchronization method)**

A start bit (one bit) and a stop bit (one to two bits) are appended before and after the character data to recognize the separation of characters. Since the timing of transmission does not match (in other words, a judgment is made in units of characters), it is also called the asynchronous method.



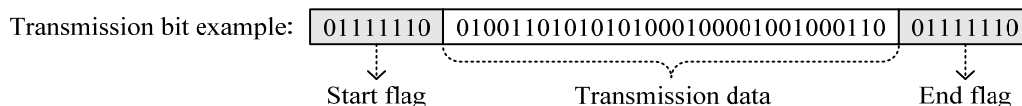
- **SYN synchronization method (Character synchronization method)**

Several (generally, two) synchronization characters (SYN code: 00010110) are appended before a block (i.e., set of character data) to recognize the beginning of the block. At the receiving side, the reception block is separated in units of eight bits and used as character data. Therefore, there is a constraint that the number of bits to be transmitted as a block must be an integral multiple of the eight bits that are separated in units of character data.



- **Flag synchronization method (Frame synchronization method)**

A flag (01111110) is appended before and after a frame (i.e., set of bits) to notify the beginning and end timing of the frame. Since the separation becomes clear with the help of the flag that is added before and after, the transmission data needs not necessarily be in units of character data (i.e., multiples of eight bits). Also, since 0 is inserted forcibly when there are five continuous 1s in the portion other than the flag, it is also called a synchronization method with no constraints in the transmission data.



1-3-3 Error Control Method

Error control refers to controlling the bit errors that occur on the communication channel when data is transmitted. Shannon recommends the two encodings that are described below as

the **coding theory** for improving reliability, efficiency, and safety in the course of information transmission.

(1) Source coding

Source coding refers to encoding (i.e., **data compression**) of the information source to reduce the amount of transmission data (i.e., number of bits). This is a concept according to which, if the bit error rate of the communication channel, etc. is a fixed value, the lower the number of transmission bits the more difficult it is for an error to occur.

There are two typical methods of source coding as described below.

- **Huffman coding**

This is an encoding method that assigns a short code to the information in the information source that has a high occurrence rate, and a long code to the information that has a low occurrence rate.

Information source	A	A	A	A	B	C	C	C	C	C	C	C
↓												
Encoding	10	10	10	10	11	0	0	0	0	0	0	0

- **Run length coding**

This is an encoding method according to which, if there is a run (i.e., range) in the information source in which the same information is iterated, the run is replaced by a combination of the iterated information and length.

Information source	A	A	A	A	B	C	C	C	C	C	C	C
↓												
Encoding	A	*	4	B	C	*	7					

(2) Channel Coding

Channel coding refers to appending a single or multiple redundant bits that are different from the information source to perform encoding. This is a concept that can detect an error that has occurred in the communication channel and can correct the error by using the appended redundant bit. It is generally called the error detection and correction method.

The typical kinds of the error detection and correction method are as described below.

	Name	Basic purpose
Error detection method	Parity check	Detect an odd number of bit errors.
	Checksum	Detect errors in units of blocks.
	CRC	Detect burst errors.
Error correction method	Hamming code	Correct 1-bit errors.

(1) Parity check

This is a method that detects an error by appending a redundant bit (i.e., parity bit) for checking to the bit string to be transmitted. This method can be classified into several techniques as shown below depending on the regularity of the redundant bit and the target to which the redundant bit is to be appended.

[Classification based on regularity]

- **Odd parity:** 0 or 1 is appended so that the number of ones becomes an odd number.
- **Even parity:** 0 or 1 is appended so that the number of ones becomes an even number.

[Classification based on the appending target]

- **Vertical parity:** The bit is appended in the vertical direction (i.e., in units of characters) with respect to the bit string.
- **Horizontal parity:** The bit is appended in the horizontal direction (i.e., in units of blocks) with respect to the bit string.
- **Horizontal/Vertical parity:** The bits are appended in both the horizontal and vertical directions.

	A	S	C	I	I	
b ₁	1	1	1	1	1	1
b ₂	0	1	1	0	0	0
b ₃	0	0	0	0	0	0
b ₄	0	0	0	1	1	0
b ₅	0	1	0	0	0	1
b ₆	0	0	0	0	0	0
b ₇	1	1	1	1	1	1
b ₈	0	0	1	1	1	1

Horizontal parity

Vertical parity

Figure 5-2 Example of horizontal/vertical parity (even parity)

In the parity check, while an odd number of bit errors can be detected, an even number of bit errors cannot be detected. By using the horizontal/vertical parity check technique, it is

possible to identify one bit error and its position (i.e., intersection point of the column and row in which the error has occurred). However, since the error may occur in several bits, the error correction function is generally not made available in the parity check.

(2) Checksum

This is a method that detects an error by appending redundant bits (e.g., a total value) that is determined through a simple calculation in units of blocks.

[Procedure of checksum]

- 1) Calculate the total value of each block of data to be transmitted at the transmitting side.
- 2) Transmit the data and the total value as redundant bits.
- 3) Calculate the total value of each block of received data at the receiving side.
- 4) Compare the calculated total value with the received total value, and make sure there is no error.

(3) CRC (Cyclic Redundancy Check)

This is a method in which transmission data is represented by a polynomial, the polynomial of the transmission data is divided by a predetermined generating polynomial, and then a checking code (CRC code) that is obtained by converting the remainder to a bit string is appended. At the receiving side, the reception data that is represented by a polynomial is divided by a generating polynomial, and the existence of an error is detected depending on the divisibility. This is a highly accurate error detection method by which continuous bit errors (i.e., **burst error**) can be detected.

Example: Calculate the CRC code for the “00110101” transmission data.

- (i) Represent the transmission data in polynomials of x

x^7	x^6	x^5	x^4	x^3	x^2	x^1	x^0
0	0	1	1	0	1	0	1

$$F(x) = \quad \quad \quad x^5 + x^4 \quad + \quad x^2 \quad + \quad 1$$

- (ii) Calculate $F'(x)$ by multiplying $F(x)$ by the highest generating polynomial.
Here, $G(x) = x^6 + x^2 + 1$ is used as the generating polynomial.

$$\begin{aligned} F'(x) &= F(x) \times x^6 \\ &= x^{11} + x^{10} + x^8 + x^6 \end{aligned}$$

- (iii) Calculate the remainder by dividing $F'(x)$ by the generating polynomial. At this time, use a special operation called “modulo 2”.

$$F'(x) \div G(x) = x^5 + x^4 + x^2 + x^1 \quad \text{Remainder} \quad x^5 + x^3 + x^2 + x^1$$

$$\begin{array}{r}
 x^6 + x^2 + 1 \quad \begin{array}{r} x^5 + x^4 \quad + x^2 + x^1 \\ \hline x^{11} + x^{10} \quad + x^8 \quad + x^6 \\ \hline x^{11} \quad \quad + x^7 \quad + x^5 \\ \hline x^{10} \quad + x^8 + x^7 + x^6 + x^5 \\ \hline x^{10} \quad \quad + x^6 \quad + x^4 \\ \hline x^8 + x^7 \quad + x^5 + x^4 \\ \hline x^8 \quad \quad + x^4 \quad + x^2 \\ \hline x^7 \quad + x^5 \quad + x^2 \\ \hline x^7 \quad \quad + x^3 \quad + x^1 \\ \hline x^5 \quad + x^3 + x^2 + x^1 \end{array} \\
 \hline
 \end{array}$$

(iv) Convert the remainder to a bit string, and use it as the CRC code that is appended to the transmission data.

Remainder = $x^5 + x^3 + x^2 + x^1$

x^5	x^4	x^3	x^2	x^1	x^0
1	0	1	1	1	0

... CRC code

Transmission bit string

0	0	1	1	0	1	0	1	1	0	1	1	1	0
Transmission data								CRC code					

(4) Hamming code

This is a method that enables the detection and correction of bit errors. A code like the Hamming code that has the purpose of correcting errors is called an **ECC (Error Correcting Code)**. The ECC is also used for correcting errors in memory.

Example 1: Generate the Hamming code of the transmission data $(b_4, b_3, b_2, b_1) = (0110)$.

- (i) Group the bits of the transmission data, and calculate check bits c_1 through c_3 by performing calculation on the basis of modulo 2 (or an exclusive OR operation) in each group.

$$c_1 = b_4 + b_3 + b_2 = 0 + 1 + 1 = 0$$

$$c_2 = b_4 + b_3 + b_1 = 0 + 1 + 0 = 1$$

$$c_3 = b_4 + b_2 + b_1 = 0 + 1 + 0 = 1$$

- (ii) Append the check bits to the transmission data, and generate the Hamming code.

$$\text{Hamming code} = (b_4, b_3, b_2, c_1, b_1, c_2, c_3)$$

$$= (0110011)$$

Example 2: Correct the error bit of the received Hamming code (0100011).

- (i) Disassemble the received Hamming code in units of bits.

$$\begin{aligned}\text{Hamming code} &= (d_7, d_6, d_5, d_4, d_3, d_2, d_1) \\ &= (b_4, b_3, b_2, c_1, b_1, c_2, c_3) \\ &= (0, 1, 0, 0, 0, 1, 1)\end{aligned}$$

- (ii) Group the bits, and use modulo 2 (or an exclusive OR operation) to calculate h_1 through h_3 .

$$\begin{aligned}h_1 &= b_4 + b_3 + b_2 + c_1 = 0 + 1 + 0 + 0 = 1 \\ h_2 &= b_4 + b_3 + b_1 + c_2 = 0 + 1 + 0 + 1 = 0 \\ h_3 &= b_4 + b_2 + b_1 + c_3 = 0 + 0 + 0 + 1 = 1\end{aligned}$$

- (iii) Identify the error bit d_n , and correct the bit error.

$$\begin{aligned}n &= h_1 \times 4 + h_2 \times 2 + h_3 = 1 \times 4 + 0 \times 2 + 1 = 5 \\ \text{Correct the error bit } d_5 &\rightarrow (01\mathbf{1}0011)\end{aligned}$$

The number of bits that can be detected or corrected by the Hamming code is decided by the **Hamming distance** (i.e., the number of different elements (i.e., bits) in the information bits of the same bit length).

- If the hamming distance is “ $m+1$ ” or more, m bit errors can be detected.
- If the hamming distance is “ $2n+1$ ” or more, n bit errors can be corrected.

In the Hamming code that is created in example 1, three redundant bits are added to the four-bit data. Therefore, it is called the (7, 4) Hamming code. Since the Hamming distance becomes three or more, it is possible to correct one bit error and detect two bit errors.

1-3-4 Switching Method

A **switching method** is used to determine how to transmit data to the other party through a line, such as a **public line**, for which the communication partner is not identified. Therefore, there is no need to consider the switching method in a **leased line service** that provides **leased lines** that are directly linked to the communication partner in a fixed manner.

(1) Circuit switching method

The **circuit switching method** is a method that has the similar configuration to a public telephone network, and transmits data by setting a physical communication channel with circuit switching equipment each time a data transmission request occurs. The network of the

circuit switching method is called a **circuit switched network**, and a **circuit switching service** refers to the service that provides a communication environment by connecting the users within the network with circuit switching equipment.

[Characteristics of circuit switching method]

- This is suitable for batch transmission of high-density/large amount of data.
- The communication equipment at the transmitting side and the receiving side must have the same communication speed.
- No transmission delay occurs within the network.

(2) Store-and-forward switching method

The **store-and-forward switching method** is used to transmit data via store-and-forward switching equipment on the basis of the address that is appended in each data unit, without setting a physical communication channel for the transmission destination. In the store-and-forward switching method, the optimum relay line is determined according to the congestion of the communication line. This is because when the traffic (i.e., data) that flows across the communication line becomes too high, the transmission efficiency declines, and a delay occurs easily.

[Characteristics of store and forward switching method]

- This is suitable for transmission of low-density/small amount of data.
- The communication equipment at the transmitting side and the receiving side need not necessarily have the same communication speed.
- Since the transmission data is accumulated in the store-and-forward switching equipment, a transmission delay occurs in the network. Moreover, the transmission order and reception order may be different.

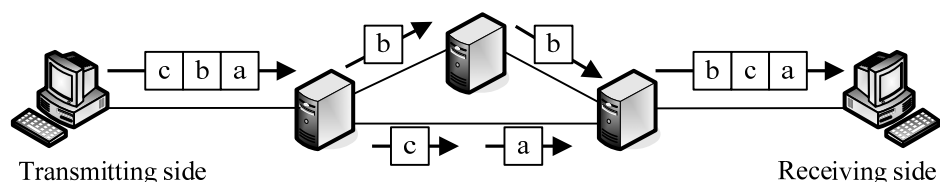


Figure 5-3 Image of the store-and-forward switching method

The store-and-forward switching method has the following three kinds of typical transmission techniques and services.

(1) **Packet switching (Packet switching service)**

This is a method that divides the transmission data into units called **packets**. A header in which the address of the other party is recorded is appended to the packet. Generally, a PT (Packet mode Terminal) with a built-in PAD (packet assembly and disassembly) function is used to connect to the packet switched network. However, even in the case of NPT (Non-Packet mode Terminal), a connection can be established by providing the PAD function in the switching equipment.

[Characteristics of packet switching]

- High-quality data transmission can be provided by performing error control in units of packets and by checking the transmission. However, since control is performed in units of packets, the communication speed decreases.
- The communication line can be used effectively (e.g., sharing of the relay line, routing function).
- By setting several logical communication lines, broadcast communication is enabled.
- It includes a PVC (Permanent Virtual Circuit) in which the other party is decided and a VC (Virtual Circuit) in which the transmission partner can be selected.

(2) **Frame relay (Frame relay service)**

This is a method that divides the transmission data into **frame** units of a variable length, and may be thought of as a packet switching method with a high speed. An address part including a DLCI (Data Link Connection Identifier) that represents the other party is appended to the frame. A connection is established to the frame relay network by a FRAD (FRame Assembly/Disassembly device) that performs assembly and disassembly of the frame.

[Characteristics of frame relay]

- High-speed data transmission can be provided by omitting error control and confirmation of data transfer to be performed in units of frames.
- Generally, CIR (Committed Information Rate) is set. CIR includes cases where the minimum communication speed is guaranteed in the normal status, and cases where the minimum communication speed is guaranteed in the congestion status when the communication line is extremely congested.

(3) **Cell relay/ATM switching (ATM service)**

This is a method that divides the transmission data into **cell** units of a fixed length. In

cell relay, high-speed data transmission is achieved through hardware routing using the **ATM (Asynchronous Transfer Mode)** technique.

[Characteristics of cell relay/ATM (Asynchronous Transfer Mode) switching]

- A cell is configured by 48 bytes of data and 5 bytes of header.
- When the communication line is congested, high speed is maintained with the help of a function for selecting a cell with a high priority order and a function for setting an alternate path.

1-3-5 Other Communication Techniques

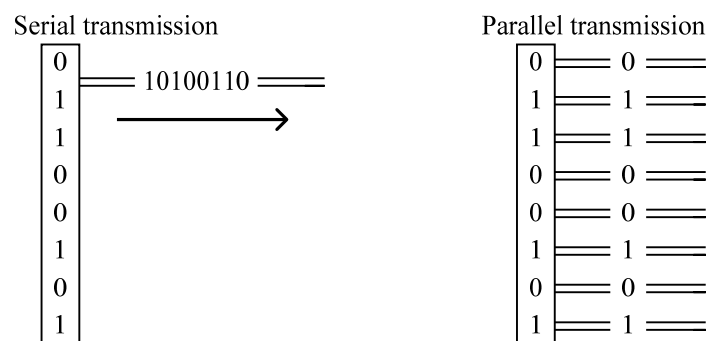
(1) Transmission method

- **Serial transmission**

This is a method that transmits data on a bit-by-bit basis. The transmission method is extremely simple, and the cost involved is also low. Generally, the transmission speed is slower than parallel transmission. However, the synchronization process of bits that are transmitted simultaneously can be skipped, and therefore the speed can be increased more easily than with parallel transmission.

- **Parallel transmission**

This is a method that transmits several bits simultaneously. Although the cost is high, the transmission speed is fast. It is used in cases where a large amount of data is to be transmitted in a batch.



(2) Communication method

- **Simplex communication** (one-way communication)

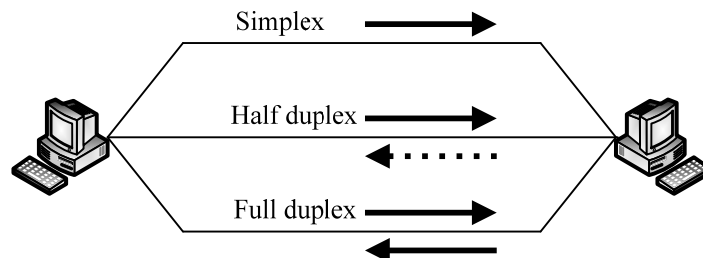
This is a communication method by which data can be transmitted only in one direction.

- **Half duplex communication**

This is a communication method by which data can be transmitted in both directions, but cannot be sent and received at the same time.

- **Full duplex communication**

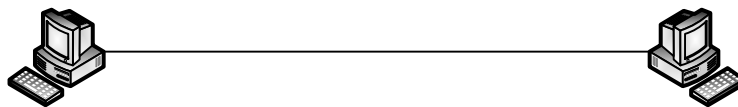
This is a communication method by which data can be transmitted simultaneously in both directions.



(3) Connection method

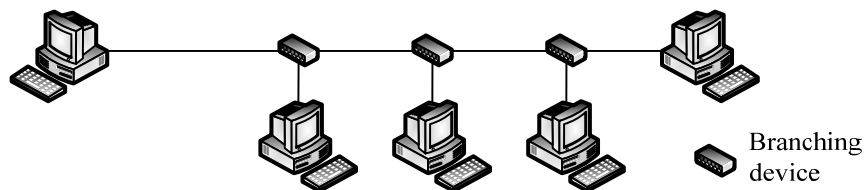
- **Point-to-point method** (point-to-point connection)

This is a method that connects computers on a one-to-one basis.



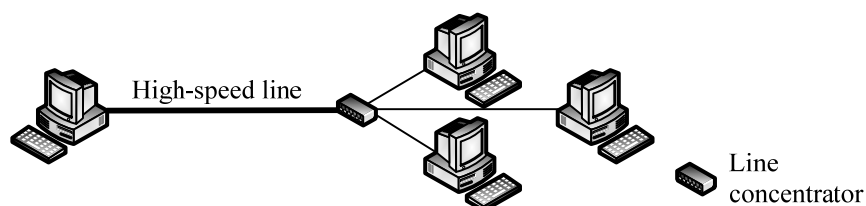
- **Multi-point (or multi-drop) method**

This is a method that connects several computers in the same line by installing branching devices in the communication line.



- **Line concentration method**

This is a method by which the lines of several computers are clustered together into a single high-speed line by a line concentrator, and connected to a remote computer.



(4) Multiplexing

- **FDM (Frequency Division Multiplexing)**

This is a multiplexing technique that is used in an analog line. By changing the frequency bandwidth in use, multiple data is transmitted simultaneously through one line.

- **TDM (Time Division Multiplexing)**

This is a multiplexing technique that is used in a digital line. By changing the connection destination in a fixed period (i.e., time slot), small pieces of data are transmitted simultaneously.

- **WDM (Wavelength Division Multiplexing)**

This is a multiplexing technique that is used in optical fiber cables. By changing the wavelength of the light that is used in data transmission, multiple data is transmitted simultaneously through one line.

- **CDM (Code Division Multiplexing)**

This is a multiplexing technique that is used in mobile communication (e.g., cell phones). By assigning a code to each user, the information flowing through the same frequency bandwidth is segregated and transmitted.

1 - 4 Transmission Control Procedures

Transmission control refers to the control that is implemented to perform efficient and accurate data transmission. The following types of controls are implemented under transmission control.

- **Line control**

This control performs physical connection/disconnection of the communication line

- **Synchronous control**

This control achieves synchronization between the transmitting side and the receiving side, and controls the transfer speed (i.e., **flow control**).

- **Error control**

This control detects (e.g., instructs retransmission) and corrects error data

- **Data link control**

This control establishes and terminates a data link (i.e., logical communication channel)

- **Routing control**

This control selects the optimum communication path for data transmission

When data transmission is performed, transmission control is implemented through each stage (i.e., phase) that is shown below.

- 1) Phase 1 <Connecting the communication line>
The communication line is connected physically. (Not required in the case of a leased line.)
- 2) Phase 2 <Establishing a data link>
The transmission partner is called and a logical communication channel is connected.
- 3) Phase 3 <Data transmission>
Synchronization is achieved, and data is transmitted. At the same time, error control of transmission data is also performed.
- 4) Phase 4 <Terminating a data link>
The end of data transmission is confirmed with the transmission partner, and the logical communication channel is closed.
- 5) Phase 5 <Disconnecting the communication line>
The physically connected communication line is disconnected. (Not required in the case of a leased line.)

Transmission control procedures refer to the procedures for implementing the above controls efficiently. However, since phase 1 and phase 5 are not required in the case of a leased line, phase 2 through phase 4 are defined in the transmission control procedures.

Similar to phase 2 through phase 4, the method of communicating by establishing a data link is called the **connection method**. In contrast, the method of communicating without establishing a data link is called the **connection-less method**.

Also, while the method of communicating by using a single data link is called the **single link procedure**, the method of communicating by using several parallel data links collectively as if they were a single data link is called the **multilink procedure**.

1-4-1 Non-Procedure (TTY Procedure)

Non-procedure (also known as TTY procedure or teletype procedure) is a method by which human beings specify the controls concerning data transmission without taking into consideration the transmission control procedures. Since only the minimum level of transmission control can be performed, it is used in low-speed lines.

Synchronous control	Start-stop synchronization method (bit synchronization method)
Error control	Parity check (vertical parity)

1-4-2 Basic Procedure (Basic Mode Data Transmission Control Procedure) ———

Basic procedure (or basic mode data transmission control procedure) is a method of transmitting data in units of blocks and performing control by using transmission control characters. Basic procedure is an interactive transmission control procedure in which subsequent processes are performed after the partner's response is waited for.

Synchronous control	SYN synchronization method (character synchronization method)
Error control	Parity check (horizontal/vertical parity)

(1) Transmission control characters

In the basic procedure, a total of ten transmission control characters that are shown below are used. The transmission control characters are special codes that are defined in the 0 column and the 1st column of the character code.

Code (column/line)	Code	Meaning
(01)	SOH	Start of header
(02)	STX	Start of text
(03)	ETX	End of text
(04)	EOT	End of transmission
(05)	ENQ	Enquiry
(06)	ACK	Acknowledgment
(10)	DLE	Data link escape
(15)	NAK	Negative acknowledgment
(16)	SYN	Synchronizing character
(17)	ETB	End of transmission block

(2) Transmission message

In the basic procedure, information is handled in units of messages. A message is divided into blocks, and information is transmitted in units of blocks.

SOH	Header	STX	Text	ETX
-----	--------	-----	------	-----

Figure 5-4 Block image

(3) Establishing a data link

The methods of establishing a data link in the basic procedure include the following two methods.

- **Contention**

This is a data link establishment method that is used when a connection is established through the point-to-point method. The transmitting side sends an ENQ (ENQuiry) for a transmission request to the other party and inquires about whether or not reception is possible. When an ACK (ACKnowledgment) is received from the receiving side, a data link is established and data is transmitted. When a NAK (Negative ACKnowledgment) is received, the enquiry (i.e., transmission request) is iterated until a data link is established.

- **Polling/selection**

This is a data link establishment method that is used when several terminals are connected in the same line by the multi-point method. The computer (i.e., control station) sends a “polling” message to each terminal (i.e., subsidiary station) in order to check if a terminal has a request to transmit data. If there is a transmission request, the computer allows the transmission. The computer also sends a “selecting” message to each terminal in order to check if a terminal is ready to receive data. If a response indicating that reception is possible is received, data is transmitted to that terminal.

1-4-3 HDLC (High-Level Data Link Control)

HDLC (High-level Data Link Control) is a transmission control procedure that is used when high-speed transmission of data is performed by transmitting data in units of frames.

Synchronous control	Flag synchronization method (Frame synchronization method)
Error control	CRC (Cyclic Redundancy Check)

(1) Format of frames

According to HDLC, the frames having the following format become the unit of transmission.

F	A	C	I	FCS	F
---	---	---	---	-----	---

F : Flag sequence (8 bits)

It is the start/end flag (01111110) that is added at the beginning and end of the frame.

A : Address field (8 bits)

The address of the transmitting station or receiving station of the frame is recorded.

C : Control field (8 bits)

The type and transmission sequence of the frame is recorded.

I : Information field (any arbitrary n bits)

The transmission data is recorded.

FCS: Frame Check Sequence (16 bits)

The CRC code for error control of the A, C, and I fields is recorded.

(2) Types of frames

Depending on the purpose, the frames that are used in HDLC can be classified into the following three types.

Type of frame	Role
I frame (Information frame)	This is used to transmit information.
S frame (Supervisory frame)	This is used to monitor the data link, and perform controls such as reception confirmation and resending request. There is no information field.
U frame (Unnumbered frame)	This is used to set the mode and report an error status. The information field may be present or may be absent.

(3) Establishing a data link

According to HDLC, the computer (or terminal) to be connected is called a station. The method of establishing a data link differs according to the type of the station.

- Unbalanced data link

This is a method of establishing a data link that is performed between a primary station that controls the establishment of the data link and a secondary station that is managed. Since the primary station performs all controls, the procedure is almost the same as the polling/selecting method.

- NRM (Normal Response Mode)

The secondary station can send a response frame to the primary station only

when a transmission permission (i.e., command) frame reaches from the primary station.

- ARM (Asynchronous Response Mode)

The secondary station can freely send a response frame even if a transmission permission (i.e., command) frame does not reach from the primary station.

Type of transmission frame	Value that is set in the address field
Command	Address of the receiving station (i.e., secondary station)
Response	Address of the transmitting station (i.e., secondary station)

- Balanced data link

This is a method of establishing a data link in a combined station in which the functions of a primary station and secondary station are combined. Since a command or a response can be sent from either station, the procedure is almost the same as the contention method.

- ABM (Asynchronous Balanced Mode)

A command frame or a response frame can be sent even if there is no transmission permission from the transmission partner.

According to HDLC, continuous transmission of information (i.e., frames) is possible by using the information (transmission sequence number/reception sequence number) of the control field after a data link is established.

1 - 5 Communication Services

In most cases, a **communication service** refers to the service that is provided by a telecommunications carrier. An overview of communication services is described below.

(1) Public line service

The **public line service** is a service that provides a network, as a public line, through which it is possible to communicate with an unspecified number of partners.

- **ADSL (Asymmetric Digital Subscriber Line)**

This is a service that provides high-speed data transmission with a different uplink and downlink speed by using an existing telephone line (i.e., twisted pair cable).

It is suitable for cases, such as the Internet, where there is a large difference in the amount of traffic depending on the communication direction. Most of the services have a communication speed with an uplink (i.e., transmission from the client) of approximately 512 kbps to 5 Mbps, and a downlink (i.e., reception of the client) of approximately 1.5 Mbps to 50 Mbps.

- **DDX (Digital Data eXchange)**

This is a public line service that is provided by NTT (Nippon Telegraph and Telephone Corporation) in Japan and includes the circuit switching service (e.g., DDX-C) and packet switching service (e.g., DDX-P, DDX-TP).

- **ISDN (Integrated Service Digital Network)**

This is a digital switching network in which several communication services, such as voice communication and data communication, are integrated, and in which the circuit switching service and packet switching service can be used. In ISDN, several channels (i.e., B channel and D channel) can be set in one line.

- **FTTH (Fiber To The Home)**

This is a connection service that uses a high-speed and high-capacity optical fiber cable. The communication speed is extremely high at 10 Mbps to 100 Mbps, and because of the development of various delivery services that use the Internet and optical lines, this service has spread to general homes.

- **Mobile communication**

This is a mobile communication service that uses cell phones and notebook PCs. In the case of a cell phone, the connection is established through the mobile base station. In the case of a notebook PC, a wireless access point and a protocol (i.e., PIAFS: PHS Internet Access Forum Standard) that makes use of PHS (Personal Handyphone System) are used, and the connection is established through a PHS base station. Another method that is available is **tethering** in which a notebook PC is connected to the Internet by using a personal digital assistant that is connected to the cell phone line. In the present-day cell phones, high-speed data communication is achieved by using **LTE (Long Term Evolution)** as a mobile communication standard in which the standards of third-generation cell phones have been extended.

- **Satellite communication service**

This is a public line service that uses a communication satellite, and can also be used in mountainous areas where it is difficult to construct a network. It is also used as an **international communication service** for communicating overseas.

(2) Leased line service

The **leased line service** is a service that provides a network through which it is possible to communicate only with an identified partner. It includes services in which a leased line is physically laid down and services that use a virtual leased line.

- **CATV (CABle TeleVision)**

Originally, it was used as a delivery service from a community antenna, but by making use of its high speed, it is now also being used in Internet connections and VOD (Video On Demand). Generally, a coaxial cable is used to improve the downlink communication speed.

- **Wide-area Ethernet**

This is a service that connects remote LANs. In-house LANs can be connected directly, and a high-speed and high-quality communication line can be used.

- **IP-VPN**

This is a **VPM (Virtual Private Network)** that is constructed by using an IP network owned by a telecommunications carrier. It can be used by multiple users as if each had a unique leased line. In contrast to IP-VPN, the VPN that is implemented on the Internet of a public line is called the **Internet VPN**.

(3) Other communication services/techniques

- **VAN (Value Added Network)**

This is a communication service that is provided by a non-facility-based telecommunications carrier by attaching some added value to a network that is constructed by borrowing a line from a facility-based telecommunications carrier.

- **PIAFS (PHS Internet Access Forum Standard)**

This is a digital data communication standard that uses PHS. High-speed digital communication at a rate of 32 kbps or more can be performed.

- **EDI (Electronic Data Interchange)**

This is a service that computerizes and exchanges the transaction data between companies through a network. The cases that are implemented on the Internet are particularly called Web-EDI.

- **IP telephone**

This is a service that uses the **VoIP (Voice over IP)** technique that converts voice information (i.e., analog signal) to a digital signal and delivers the information to the partner in the form of IP packets. Since an IP network such as the Internet can be used as a common communications infrastructure without using an existing public

telephone network, the communication cost can be reduced.

- **Ubiquitous network**

This is an information service that combines several information devices, such as intelligent home appliances and cell phones, with a high-speed network service called the broadband network, and is used at all places in everyday life. (“Ubiquitous” = “Present everywhere”)

2 Network Architecture

According to JIS (Japanese Industrial Standards), network architecture is defined as the “logical structure and operating principles of a computer network.” Specifically speaking, it refers to systematically organizing various protocols and interfaces, which are required for constructing a network, in a hierarchical structure.

2 - 1 What is Network Architecture?

In order to properly transmit data through a network, it is necessary to conclude various agreements with the communication partners. There are several aspects to be decided, such as the voltage level of the transmission signal, method of transmission, and format of data to be transmitted. These agreements (i.e., conventions) are called **protocols**.

When data communication is performed with several partners because of the expansion of network, it becomes difficult to set a protocol for each partner. Thus, **network architecture** is taken into consideration to systematically organize the protocols and simplify the construction/designing of the network.

2 - 2 OSI (Open Systems Interconnection)

OSI (Open Systems Interconnection) is the network architecture of international standards that was established in the year 1983 mainly by ISO (International Organization for Standardization) and ITU-T (International Telecommunication Union - Telecommunication Standardization Sector). When network architecture was first taken into consideration, each manufacturer published unique network architecture. However, when a network is to be constructed with the products of different manufacturers, the network architecture that is unique to manufacturers is not useful. Thus, standard network architecture was considered necessary.

According to OSI, the functions necessary for communication are arranged in seven hierarchies (i.e., layers). This is called the **OSI basic reference model**.

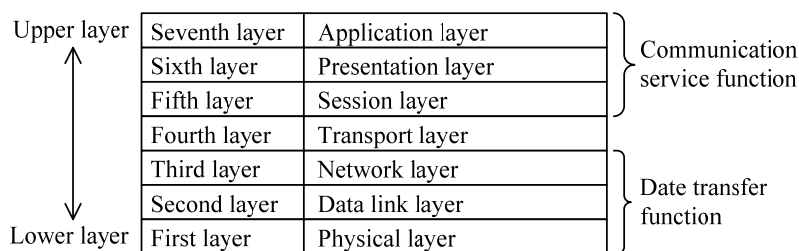


Figure 5-5 OSI basic reference model

Several function modules (i.e., entities) are present in each layer of OSI. (The entity present in the N -th layer is called the (N) entity.) The (N) entity is a service provided from the ($N-1$) entity that belongs one layer below, and provides service to the ($N+1$) entity that belongs one layer above. (The service provided by the entity of the N -th layer is called the (N) service.) Moreover, the protocol present between the (N) entities belonging to the same level of the transmitting side and the receiving side is called the (N) protocol. Data is exchanged between the (N) entities by using an (N) connection, which is a logical communication channel.

The service provided by each layer of OSI and the specific functions are as described below.

(1) **Application layer**

It is positioned at the highest level of the OSI basic reference model, and provides the communication function (i.e., service) to the application program. Specifically, it includes FTAM (File Transfer, Access, and Management) and RDA (Remote Database Access).

(2) **Presentation layer**

The presentation layer converts the presentation style (i.e., abstract syntax) of the exchanged data to general syntax (i.e., transfer syntax). Specifically, it converts the data of a different code system so as to be able to perform data communication in the same code system.

(3) **Session layer**

This provides the data transfer function that can be used in common in various applications. Specifically, it manages data exchange on the basis of interactive control functions, such as setting the synchronization point (i.e., data separation) and controlling the transmission right (i.e., token).

(4) **Transport layer**

This provides a transparent and high-quality data transfer at the transmitting side and receiving side (i.e., between end-to-end communication partners). Specifically, it complements the services that are lacking in the lower layer, and performs error control, flow control, and such other control.

(5) **Network layer**

This provides a communication channel from end to end. Specifically, it includes network control functions, such as the function for selecting the path (i.e., routing) for data transmission and the function for transferring/relaying data.

(6) **Data link layer**

This provides data transmission with high reliability and transparency between adjacent nodes. Specifically, it establishes a data link connection and performs transmission control that is typified by HDLC.

(7) Physical layer

This provides the transmission function in units of bits by using a communication line. Specifically, it defines mechanical standards such as the shape of the connector, electrical standards such as the voltage level, and logical standards such as the control method of the signal line.

2 - 3 TCP/IP

TCP/IP is a protocol that is used in the Internet. It was the protocol used at America's ARPANET (original model of the present Internet), but as a result of an increase in the number of users, it has become the **de facto standard** (i.e., industry standard) protocol.

TCP/IP is a set of several protocols, and is systematically organized in the hierarchical structure in the same way as OSI. The correspondence between each layer of TCP/IP and the OSI basic reference model is as shown below.

OSI basic reference model	TCP/IP model
Application layer	AP layer (Application layer)
Presentation layer	
Session layer	
Transport layer	Transport layer/TCP layer
Network layer	Internet layer/IP layer
Data link layer	Data link layer (link layer)
Physical layer	/ NI layer (Network interface layer)

Note: In the names of layers of the TCP/IP model, the term within () is the abbreviated name, and the term that is indicated after "/" is another name.

(1) AP layer (Application layer)

The **application layer** provides various services to the user. The services that are provided over the Internet are implemented through the protocols of this layer. (Various services of the Internet are explained in detail in Section 4.).

Protocol name	Basic role
DNS	This converts FQDN (Fully Qualified Domain Name) to the IP address.
DHCP	This dynamically assigns the IP address.
SMTP	This transfers e-mails to the mail server or between main servers.
POP	This downloads e-mails from the mail server.
IMAP	This retrieves e-mails from the mail server.

MIME	This enables handling of audio/video data through e-mails.
HTTP	This transfers hypertext (e.g., HTML documents).
FTP	This transfers files.
SNMP	This manages the network in a simple manner.
TELNET	This performs remote login from a remote terminal.
NTP	This synchronizes the time in several nodes.
NNTP	This distributes news articles.
RTP	This transfers video and audio data in a format suitable for real time.
BOOTP	This acquires the settings of the network during OS boot.
SOAP	This calls XML-based data and services.

(2) Transport layer / TCP layer

The **transport layer/TCP layer** provides transparent, high quality, and high-speed data transmission from end to end. (This layer corresponds to the transport layer of the OSI basic reference model.)

Protocol name	Basic role
TCP	This guarantees high reliability in connection-oriented communication in which a logical communication channel is established.
UDP	This provides high speed instead of not guaranteeing reliability in connectionless communication in which a logical communication channel is not established.

(3) Internet layer / IP layer

The **Internet layer / IP layer** provides the function for the path selection (i.e., routing) for data transmission and the function for transferring/relaying data. (This layer corresponds to the network layer of the OSI basic reference model.)

Protocol name	Basic role
IP	This uses the IP address to transfer packets.
RIP	This is used to select the communication path.
ARP	This acquires a MAC address from an IP address.
RARP	This acquires an IP address from a MAC address.
ICMP	This gives notification of communication error and network status.

(4) Data link layer / NI layer (Network Interface layer)

The **data link layer/network interface layer** provides transparent and error-free data transmission. It corresponds to the physical layer and data link layer in the OSI basic reference model. Generally, the data link layer is handled by dividing into the LLC (Logical Data Link) sublayer and the MAC (Media Address Control) sublayer.

Protocol name	Basic role
PPP	This uses the telephone line to connect to the network.
PPPoE	This establishes a dial-up connection on the Ethernet.

[CORBA (COmmon Request Broker Architecture)]

CORBA is a standard specification that enables exchange of messages between objects that are created with different program languages under a distributed network environment. In a broad sense, it can be referred to as part of the network architecture.

3 LAN

LAN (Local Area Network) is a network that is set up in a narrow area (or premise). This section provides an insight into the devices and techniques that are used to construct a LAN.

3 - 1 Basic Techniques of a LAN

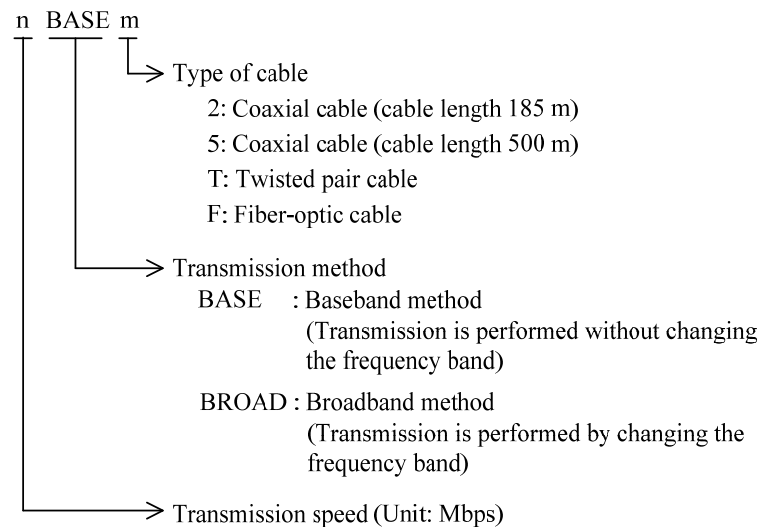
A LAN is constructed by connecting the computers and peripheral equipment in a company and premise with a private line. The construction of a LAN has the following merits.

- Since LAN is a private line, it involves expenses concerning line setup, but no line usage charge.
- Since it is a network within a restricted area, high-quality/high-speed data transmission can be achieved.
- By using groupware, effective group activities can be implemented. It also promotes a paperless culture in the office.
- Various resources, starting from hardware resources such as printers, right up to files and database, can be shared.

(1) Wired LAN

A **wired LAN** is connected by using a communication line (i.e., cable), and supports the **IEEE 802.3** standard. An IEEE 802.3 LAN is also called **Ethernet**, and is represented by the following symbolic notation. For example, 10BASE5 is a baseband LAN which has a transmission speed of 10 Mbps and uses a coaxial cable with a cable length of 500 m.

[Standard LAN symbols]



In a wired LAN, the three types of media (i.e., cables) that are described below are used for connecting devices. Moreover, a **LAN module** (e.g., a LAN adapter, a **NIC (Network Interface Card)**), as a standard installation or an additional installation) is used for connecting the PC to LAN.

- **Twisted pair cable**

This is a cable that is used as a telephone line. Although the price is low, this cable is sensitive to noise, and the transmission distance is also just a few hundred meters. It is generally used in small-size LANs.

- **Coaxial cable**

This is a cable in which the center conductor is covered by an outer conductor and an insulator. Although the price is somewhat high, it is resistant to noise, and the transmission distance is also up to a few tens of kilometers.

- **Optical fiber cable** (Fiber-optic cable)

This is a cable in which optical fibers that transmit light are bundled together, and it supports extremely high-speed/high-quality data transmission. Since it is resistant to noise and the damping of light is also less, a long distance transmission of approximately 100 km is possible. Although there is not much difference in the cost of the cable itself as compared with a coaxial cable, the cost of peripheral equipment and installation expenses are high. However, price reductions are being promoted, and it is presently being put to general use.

(2) Wireless LAN

A **wireless LAN** is connected by using electromagnetic waves or infrared rays, and supports the IEEE 802.11a/b/g/n standards (at present, **IEEE 802.11n**, which uses a frequency band of 2.4 GHz/5 GHz, and has a maximum communication speed of 600 Mbps is the main standard).

Since a wireless LAN does not use cables, the layout can be changed easily, and as long as there are no obstacles, communication can be performed even at a distance of 50 to 100 meters. When a wireless LAN is used, a user connects the terminal to a **wireless LAN access point**. At this time, the wireless LAN sets an access point identifier **SSID (Service Set Identifier)**, or a network identifier **ESSID (Extended SSID)**, which is an extended edition of SSID, in order to identify the wireless LAN access point. By making this setting, only the devices having the same SSID or ESSID are identified as the appropriate users (or devices), and therefore unauthorized use (or access) is prevented. However, since it is difficult to secure safety only through this method, it is necessary to examine sufficient security measures.

In addition, a wireless LAN can even be connected to a wired LAN and the Internet through a wireless LAN access point. (Generally, this method is often used.)

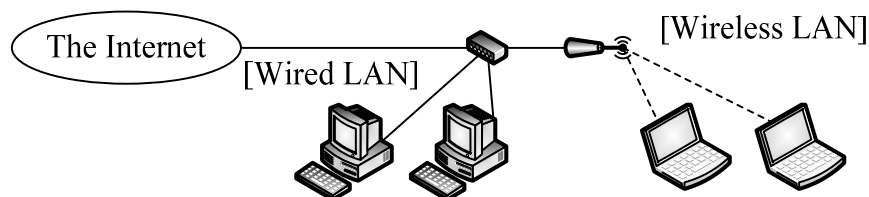


Figure 5-6 Connection between a wireless LAN and wired LAN / the Internet

3-1-1 Topology (Connection Form)

The types of LAN **topology** (i.e., connection form) are described below. The LAN topology includes physical topology and logical topology, and these need not necessarily match.

- **Star**

This is a topology in which all nodes (i.e., devices) are connected to the node (e.g., a line concentrator) that is installed at the center. Although the fault of one node does not have an overall influence, if a fault occurs in the concentrator, it has much effect on the entire system.

- **Ring**

This is a topology in which the nodes are connected in the shape of a ring. The communication channel is short, and the control is also simple and efficient, but the

failure of one node may have an overall effect.

- **Bus**

This is a topology in which all nodes are connected to one backbone cable (i.e., shared line). Addition and deletion of nodes can be performed easily without any overall effect. However, if the number of nodes increases and the amount of traffic also increases, collisions may occur frequently on the backbone cable, and the transmission efficiency declines rapidly.

- **Terminator**

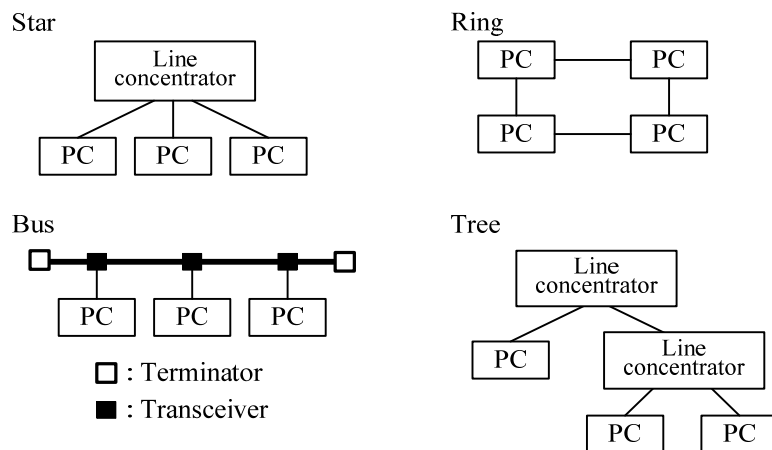
This is a device that is installed at both ends of the backbone cable in a bus LAN. The unnecessary data flowing across the backbone cable is removed.

- **Transceiver**

This is a device that connects the backbone cable and nodes in a bus LAN. It also has the function of detecting data collision.

- **Tree**

This is a topology in which nodes are connected in a way like branches and leaves grow out from the root by making line concentrators have a **cascade connection** (i.e., multi-stage connection).



3-1-2 MAC (Media Access Control)

MAC (Media Access Control) defines the data transmission method and error detection method. In a LAN, the method of acquiring access rights for data transmission is mainly controlled. The MAC of LAN is defined in the MAC sublayer of the data link layer of the OSI basic reference model. Therefore, when data (i.e., frames) is transmitted, the source address and the destination address are identified by the **MAC address** that is recorded in the LAN card as the identifier of the data link layer.

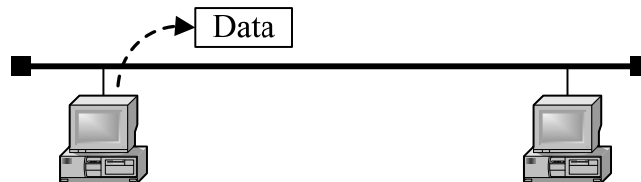
MAC address	OUI (vendor ID)	Unique manufacturing number
	24 bits	24 bits

(1) CSMA/CD

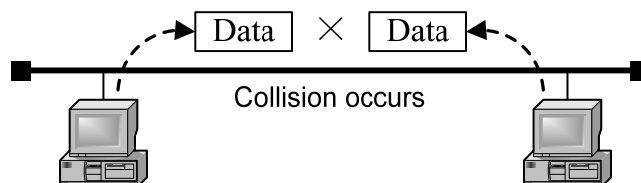
CSMA/CD (Carrier Sense Multiple Access with Collision Detection) is mainly used in bus LAN, which is defined on the basis of the IEEE 802.3 that is a standard of bus LAN.

When a node transmits data, it first checks whether data is flowing across the backbone cable. If data is not flowing, the node starts transmission by assuming that it has acquired the access permission. At this time, if there is another node that transmits data at almost the same time, data collision occurs on the backbone cable. If such a collision is detected, the node temporarily stops the transmission of data, and then performs retransmission after a fixed period of time (i.e., different time in each node) that is set in each node has elapsed.

- 1) The node starts transmission because there is no data in the backbone cable.



- 2) The node stops transmission when data collision occurs on the backbone cable.
→ It starts retransmission after a fixed period of time has elapsed.



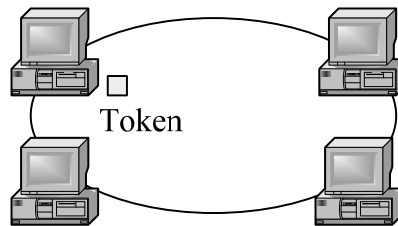
Since CSMA/CD must be used in half-duplex communication, it is not used very often in the current times when full duplex communication has become common.

Moreover, since a collision cannot be detected in a wireless LAN, the **CSMA/CA (CSMA with Collision Avoidance)** method that checks the arrival of data on the basis of an ACK signal from the receiving side is used.

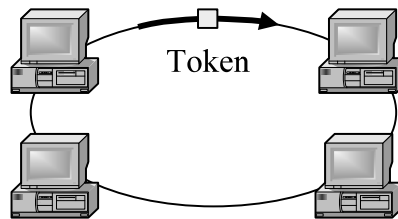
(2) Token passing

Token passing is a method by which a terminal that acquires special data called a token (i.e., transmission right) performs data transmission. It is referred to as **token ring** (i.e., IEEE 802.5) in the case of a ring LAN, and as **token bus** (i.e., IEEE 802.4) in the case of a bus LAN.

- 1) The terminal that acquires the token transmits data.



- 2) The terminal on which data transmission has ended circulates the token.
→ The next terminal can acquire the token and start data transmission.



(3) TDMA (Time Division Multiplex Access)

TDMA (Time Division Multiple Access) is a method by which only the node to which a time slot has been assigned can perform data transmission. Since data transmission can be performed at regular intervals, a large transmission delay does not generally occur even in the high-traffic, high-load environment.

3-1-3 Relationship Between Connected Devices

The relationship between the connected devices of LAN is classified into client/server type and peer-to-peer type.

In the client/server type, there is a clear vertical relationship between the connected devices. Specifically, it is configured by the client that requests the processing (or requests the service), and the server that performs the processing (or provides the service).

On the other hand, in the peer-to-peer type, all connected devices are at an equivalent position. The device that acts as the client requesting a process at a particular point of time also acts as the server that is requested to perform the processing at another point of time.

3-1-4 LAN-to-LAN Connection Device

The LAN-to-LAN connection device connects several LANs. LAN was originally a network in a restricted area. However, as a result of diversification of the processing content, it was required to become a wide-area network by connecting to other LANs, and therefore, a LAN-to-LAN connection device became necessary.

- **Repeater**

This is a device that connects LANs at the physical layer level. It physically connects the network, and performs only signal reshaping and amplification. Therefore, the connected LANs only extend the length of the transmission path and are physically handled as a single LAN. A similar device is a **hub** (or a **repeater hub**). It must be noted that although a hub can have a **cascade connection** (i.e., multi-stage connection), the number of connection stages is restricted.

- **Bridge**

This is a device that connects LANs at the data link layer level. The MAC address learning function and filtering function are available, and therefore, the data that is transmitted to the same LAN does not flow to another LAN. As a result, the amount of traffic within the LAN can be reduced. Moreover, in order to prevent the data from flowing infinitely because of the circulation of the LAN circuit, the **spanning tree** function that configures the virtual tree structure is also available. A similar device is a **switching hub** (i.e., **layer 2 switch** or **L2 switch**.)

- **Router**

This is a device that connects LANs at the network layer level. Since a routing function on the basis of the network address (i.e., IP address) is available, data can be transmitted only to the target LAN through the best path. The routing function further includes a function called **strict routing** that strictly defines the router that is passed until the target LAN is reached.

Moreover, a **layer 3 switch** (**L3 switch**), which is a layer 2 switch with a simple router function (i.e., routing function), is also available. Besides a physical connection, a layer 3 switch is used in constructing a **VLAN (Virtual LAN)** that forms a virtual LAN group.

In addition, a router also performs the role of a digital service unit that is used in establishing a connection to WAN, and includes a **dial-up router** that is used to establish a connection to a public line.

- **Gateway**

This is a device that connects LANs at the transport layer or higher. Besides connecting LANs with a different protocol, it is also used to establish a connection to WAN. At this time, another gateway is available that performs the role of a **proxy server** that is used to establish a connection to the Internet on behalf of a PC within the LAN. Since gateway also refers to the exit/entry of the LAN, the router that acts as the standard exit/entry of the LAN is mostly specified as the **default gateway**.

3 - 2 Other LAN Techniques

(1) FDDI (Fiber-Distributed Data Interface)

FDDI is a LAN that is implemented by developing a ring LAN, and is a network having a transmission capacity of approximately 100 Mbps that uses an optical fiber cable as the backbone cable.

It includes FDDI-I that is centered on packet switching and FDDI-II that can transmit voice and images.

(2) High-speed Ethernet

High-speed Ethernet is a LAN that achieves high-speed data transmission by inheriting the concept of the IEEE 802.3 bus LAN (i.e., Ethernet). **10BASE-T** (i.e., 10 Mbps), which was popular in the past, is used as the reference for high-speed data transmission.

- **Fast Ethernet**

This is a collective term for Ethernets having a transfer speed of 100 Mbps or more.

- **100BASE-TX**

This is a high-speed Ethernet that uses a twisted pair cable of category 5 or above. It is also commonly referred to as 100BASE-T when combined with 100BASE-T2/100BASE-T4 that uses a twisted pair cable. The category of the twisted pair cable is the classification on the basis of the frequency and the decline in signal being used. Moreover, a shielded twisted pair cable is classified as STP and an unshielded twisted pair cable is classified as UTP.

- **100BASE-FX**

This is a high-speed Ethernet that uses an optical fiber cable.

- **Gigabit Ethernet**

This is a collective term for Ethernets having a transfer speed of 1000 Mbps or more.

- **1000BASE-T**

This is a high-speed Ethernet that uses a twisted pair cable of category 5 or above.

- **1000BASE-FX**

This is a high-speed Ethernet that uses an optical fiber cable.

(3) ATM-LAN

ATM-LAN is a high-speed LAN that uses the ATM (Asynchronous Transfer Mode) technique.

This is a network that can transmit multimedia data, such as moving images, at a high speed.

4 The Internet

The Internet is a network that establishes an integrated connection between the networks that are interspersed across the world. Therefore, it is also called the network of networks.

4 - 1 TCP/IP Protocol

TCP/IP is the standard protocol of the Internet. The relatively simple configuration as compared with the OSI basic reference model is also considered to have become the indirect cause of the evolution of the Internet.

In TCP/IP, data is transmitted by appending the management information of each layer as a **header** to the **packets** to be sent. The role of each layer in connection-oriented communication using TCP and the typical management information that is included in each header are as shown below.

1) Application layer

This layer manages the data that is transmitted by the protocol corresponding to the provided service.

Transmission data

2) Transport layer (protocol: TCP)

This layer manages the information concerning the service (i.e., protocol) by appending the **TCP header**.

TCP header			Transmission data
Source Port number	Destination Port number	Sequence number	

- **Port number**: This is a number that represents the service (i.e., protocol).
- **Sequence number**: This is a number that represents the order of the packets.

3) Internet layer (protocol: IP)

This layer manages the information concerning the IP address by appending the **IP header**.

IP header			TCP header	Transmission data
Protocol number	Source IP address	Destination IP address		

- **Protocol number**: This is a number that represents the protocol of the next header to the IP header.
- **IP address**: This is an address for identifying the network and devices.

4) Data link layer

This layer manages the information concerning the MAC address by appending the **Ethernet header**.

Ethernet header		IP header	TCP header	Transmission data
Source MAC address	Destination MAC address			

- **MAC address**: This is a device ((i.e., hardware) specific identifier (i.e., address).

4-1-1 Role of the Transport Layer

The main role of the transport layer is to deliver the data that is sent from the service (i.e., protocol) that is used by the source application to the service (i.e., protocol) that is used by the destination application. Moreover, sequence control and retransmission control using the sequence number are also performed.

In the transport layer, a **port number** (i.e., protocol number) is used to identify the source and destination service (i.e., protocol). Generally, a **well-known port number** that is decided for each transport layer protocol (e.g., TCP, UDP) is used as the port number. The typical well-known port numbers that are used in **TCP (Transmission Control Protocol)** are as shown below.

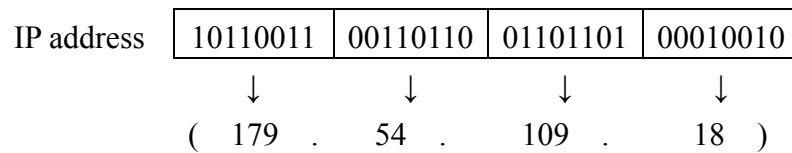
Port number	Service	Protocol
21	File transfer service	FTP
25	E-mail transfer service	SMTP
80	HTML file transfer service	HTTP
110	E-mail reception service	POP3

4-1-2 Role of the Internet Layer

The main role of the Internet layer is to deliver the data that is sent from the source device (e.g., computer) to the destination device via the network.

In **IP (Internet Protocol)**, which is the protocol of the Internet layer, the **IP address** is used to identify the source and destination devices. The IP address is a unique address that is assigned to each device, and is managed by **NIC (Network Information Center)** and **JPNIC (Japan Network Information Center)** so that no duplication occurs across the world.

IP addresses include those defined by **IPv4 (IP version 4)** and those defined by **IPv6 (IP version 6)**. IP addresses that are defined by IPv4 are configured by 32 bits, and are normally represented as decimal numbers separated into four portions of eight bit each.



On the other hand, IP addresses that are defined by IPv6 are configured by 128 bits, and are normally represented as hexadecimal numbers separated into eight portions of 16 bits each. IPv6 resolves the shortage of IP addresses that are defined by IPv4, and at the same time, strengthens the security function through packet encryption and authentication by using the security protocol called IPsec as the standard specification.

An explanation is provided below concerning the structure of the IP address by using a 32-bit IP address that is defined by IPv4.

(1) Classification of IP addresses

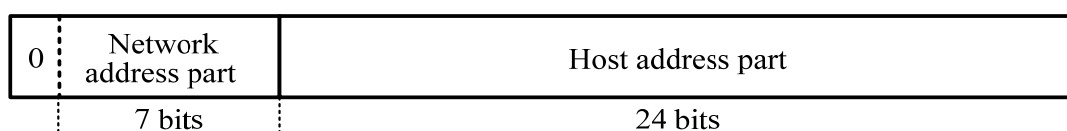
The IP address is configured by a network address part and a host address part. The identifier (i.e., **network address**) indicating the network through which the devices are connected is recorded in the network address part, and the identifier (i.e., **host address**) indicating the devices is recorded in the host address part.

Depending on the number of bits that is used in each part, the IP address is classified into classes A through D. Among these, class D is used only in **multicast** communication (where data is sent to all devices belonging to a specific group). Moreover, among the IP addresses of classes A through C, those with the host address part of all 0s are used as the network address, and those with the host address part of all 1s are used as the broadcast address (i.e., an address used in **broadcast** communication where data is sent to all devices that are connected to the network), and are not used generally because of their special meanings. One-to-one communication in which a host address indicating a single device is specified is called **unicast**.

The configuration of IP addresses of classes A through D is as shown below.

- Class A

This class is assigned to large networks.



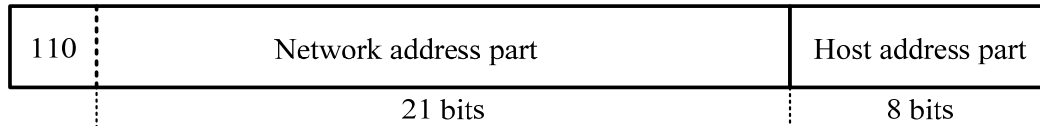
- Class B

This class is assigned to medium- to large-sized networks.



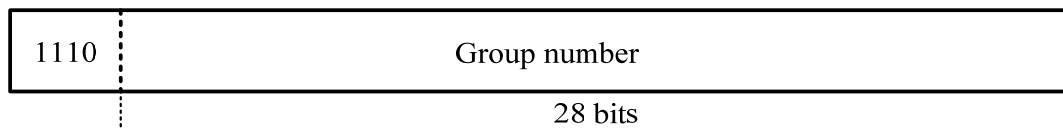
- Class C

This class is assigned to small networks.



- Class D

This class is used in multicast communication.



(2) Utilization of IP addresses

The following two methods are available for an effective use of the IP address.

- Subnet

This is a concept that divides one network address into several **subnet addresses** by using part of the host address as a subnet address part.

At this time, the bit string that is used for extracting the subnet address through a logical product operation is called a **subnet mask**. The concept of a subnet is also used when a large network is segmented into small networks and managed.

	Network address part		Host address part	
IP address (179.33.108.178)	10110011	00100001	01101100	10110010
Subnet mask (255.255.224.0)	11111111	11111111	11100000	00000000
Subnet address (179.33.96.0)	10110011	00100001	01100000	00000000

This part is extracted and is used to identify the network.

- CIDR (Classless Inter-Domain Routing)

This is a concept of using the IP address by excluding the general idea of a class. In CIDR, the number of bits of the network address part is not decided in a fixed manner,

but is decided at the time of assigning the IP address. For example, when 19 bits from the beginning are used as the network address, the number of bits (i.e., **prefix value**) of the network address is also specified after “/” as in “179.33.96.0/19”. In addition, by using the **VLSM (Variable Length Subnet Mask)**, it becomes possible to use a network address of a different length in each group.

(3) Global IP address and private IP address

The **global IP address** is a unique IP address by which a device can be identified. While there is no problem if a global IP address is assigned to all devices, the number of IP address that can be assigned is limited. This has led to the birth of the concept of a **private IP address** that can be used only within a restricted area. Since there is no problem as long as the private IP address can be identified within a restricted area, the user can set any arbitrary private IP address. However, the range of the private IP addresses that can be set is determined as shown below for each class.

Class A	10.	0.0.0/8	to	10.255.255.255/8
Class B	172.	16.0.0/12	to	172.31.255.255/12
Class C	192.	168.0.0/16	to	192.168.255.255/16

As long as the usage area is different, there is no problem even if the private IP address is a duplicate. However, since uniqueness is not maintained in an external network, it is not possible to connect to an external network through a private IP address. Therefore, in order to connect to an external network through a terminal on which a private IP address is set, it is necessary to convert the private IP address to a global IP address through a router or such other device. This conversion can be performed with the two types of methods as described below.

- **NAT (Network Address Translation)**

This performs one-to-one conversion of the private IP address and its corresponding global IP address.

- **NAPT (Network Address Port Translation)**

This converts several private IP addresses, including the port number, to one global IP address. It is also called **IP masquerade**.

(4) DNS (Domain Name System)

DNS is a protocol for converting **FQDN (Fully Qualified Domain Name)**; fully qualified domain name = host name + domain name) to an IP address. This conversion is referred to as name resolution.

Since the IP address is an enumeration of numeric characters, it is difficult for humans to remember it. Therefore, names (FQDNs) that can be understood easily by humans are used in the **e-mail address** of e-mails and also in the **URL (Uniform Resource Locator)** that is used to browse a web page. (For example, the URL “www1.its.co.jp” is an FQDN with the host name (i.e., server name) as “www1” and the domain name as “its.co.jp”.)

An FQDN is used by converting to an IP address within the network. Therefore, conversion to the IP address is performed by the **DNS server** that manages the correspondence between the FQDN and IP address. However, it is not possible to record the FQDNs and IP addresses of the entire world in one DNS server. Therefore, the DNS server itself is managed in a hierarchical structure. Thus, the structure is such that by moving in order from an upper DNS server to a lower DNS server, it is possible to reach the DNS server in which the information of the target IP address is recorded.

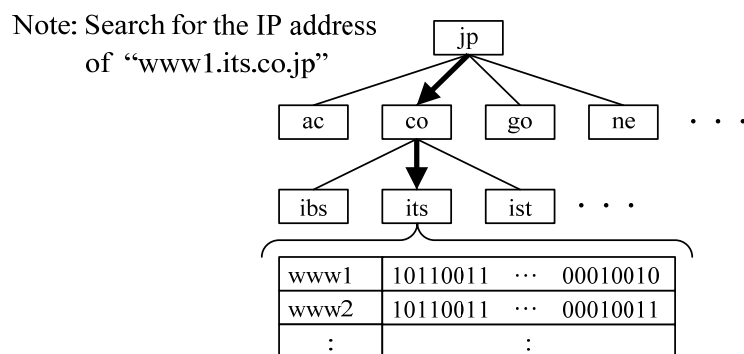


Figure 5-7 A DNS server managed in a hierarchical structure

(5) DHCP (Dynamic Host Configuration Protocol)

DHCP is a protocol for dynamically assigning an IP address to the devices (i.e., computers) that are connected to the network. By using DHCP, one IP address can be shared among several devices, and this saves the user from the trouble of configuring settings related to the IP address. However, since the same IP address cannot be used simultaneously, use the procedure that is described below to confirm that the IP address is not in use in the DHCP server and the DHCP client.

[Checking by the DHCP server]

The DHCP server sends the **ICMP** echo request packet that includes the IP address to be

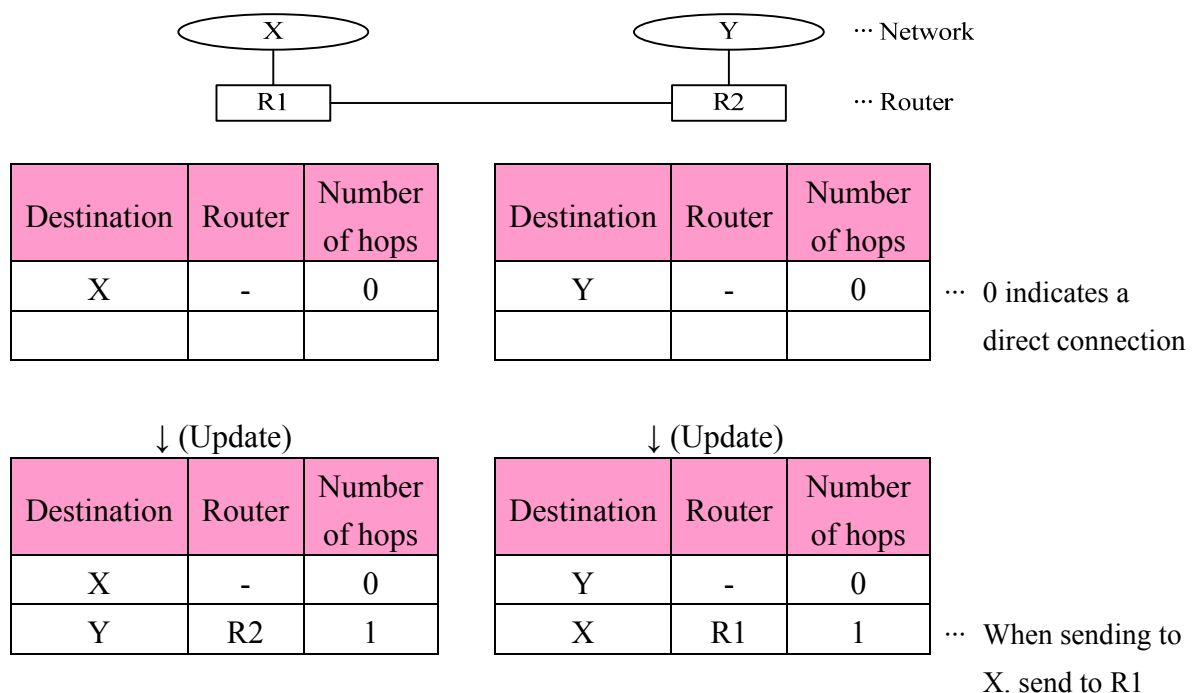
assigned, in order to make sure that an echo response packet is not returned. (ICMP is a protocol for notifying the status of communication error and network, and the echo function confirms the packet delivery through the existence of a response to the request.)

[Checking by the DHCP client]

The DHCP client sends the **ARP** request packet that includes the assigned IP address, in order to make sure that a response packet is not returned. (ARP is a protocol for acquiring the MAC address corresponding to the IP address.)

(6) RIP (Routing Information Protocol)

RIP is a protocol for selecting the path up to the target network by using the IP address. (The process of directing transmitted data to another network is called **routing**.) In RIP, the number of hops (i.e., number of routers) that are passed is taken as the reference, and the distance vector algorithm for selecting the path with the minimum number of hops is used. Each router maintains a routing table, and the content of the routing table is updated through transmission at regular time intervals. However, there are disadvantages, such as the fact that “a parallel path cannot be created” and that “the minimum number of hops does not always become the shortest path.”

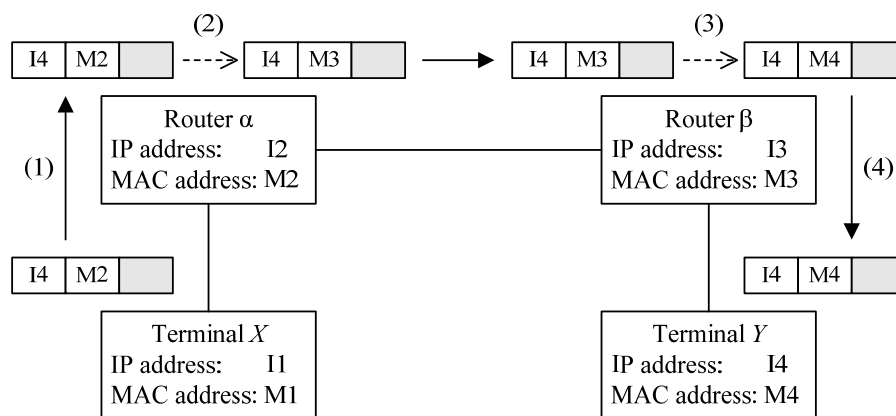


4-1-3 Role of the Data Link Layer

The main role of the data link layer is to deliver data to the devices (e.g., computers) that are connected directly by using the communication medium (e.g., cables).

In the data link layer, a **MAC address** is used to identify the individual device. The MAC address is a device (i.e., hardware) specific identifier. (However, since it is not managed in a consolidated manner, it is not guaranteed to be absolutely unique.) If the destination MAC address of a packet flowing across the communication channel matches the MAC address of a device that is connected to the network, the device acquires that packet.

The flow of the communication using an IP address and a MAC address is as shown below. The MAC address corresponding to the destination IP address is acquired by **ARP (Address Resolution Protocol)**.



- (1) [Terminal X] In order to send a packet to terminal Y, set the IP address (i.e., I4) of terminal Y in the destination IP address. However, since terminal Y is not a directly connected device, set the MAC address (i.e., M2) of the directly connected router α in the destination MAC address to send the packet.
- (2) [Router α] Since the destination MAC address (i.e., M2) is addressed to self, the packet is acquired. However, since the destination IP address (i.e., I4) is not addressed to self, the packet is sent by setting the MAC address (i.e., M3) of the router β that is connected to the network of the destination IP address in the destination MAC address.
- (3) [Router β] Since the destination MAC address (i.e., M3) is addressed to self, the packet is acquired. However, since the destination IP address (i.e., I4) is not addressed to self, the packet is sent by setting the MAC address (i.e., M4) of the device (i.e., terminal Y) of the directly connected destination IP address in the destination MAC address.
- (4) [Terminal Y] Since the destination MAC address (i.e., M4) is addressed to self,

the packet is acquired. Moreover, since the destination IP address (i.e., I4) is also addressed to self, the packet is processed in an upper layer.

4 - 2 Basic Configuration of the Internet

The basic configuration for using the Internet from a company's internal LAN is as shown below. (This is just an example of the basic configuration, and the configuration need not be absolutely identical.)

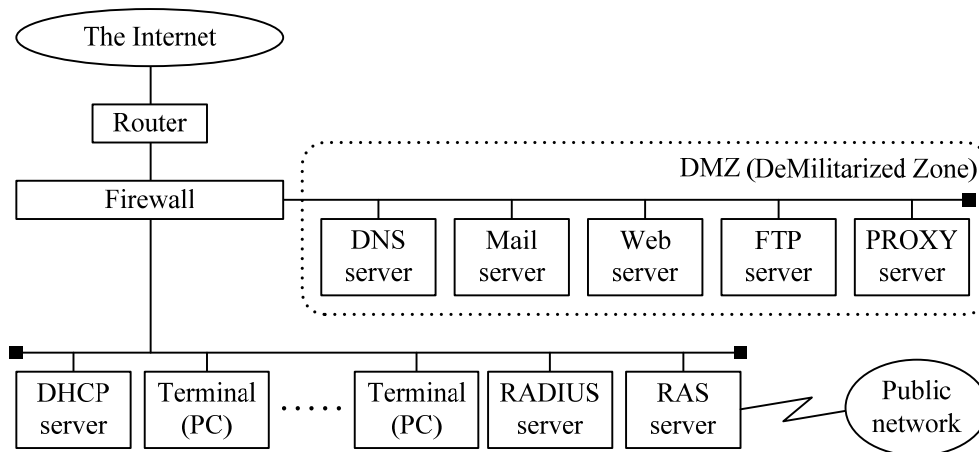


Figure 5-8 Basic configuration when the Internet is used (Example)

The role of the components (e.g., the server) in Figure 8 is as described below. The mail server, web server, and FTP server are explained later together with the services.

- **Firewall**

This is a device for preventing unauthorized intrusion from an external network. A **DMZ (DeMilitarized Zone)** that is isolated from the internal network is set up, and it separates the external network from the internal network to avoid a direct connection.

- **DNS (Domain Name System) server**

This is a server that provides the name resolution service through DNS.

- **PROXY server**

This is a server that provides the service for establishing a proxy connection to the Internet instead of the client terminal of an internal network.

- **DHCP (Dynamic Host Configuration Protocol) server**

This is a server that provides the service for dynamic assignment of the IP address by DHCP.

- **RADIUS (Remote Authentication Dial-In User Service) server**

This is a server that provides the service for simplifying network management by centralizing the authentication of the overall system.

- **RAS (Remote Access Service) server**

This is a server that provides the service to enable remote access to the network through a public network, such as a telephone line.

4 - 3 Internet Services

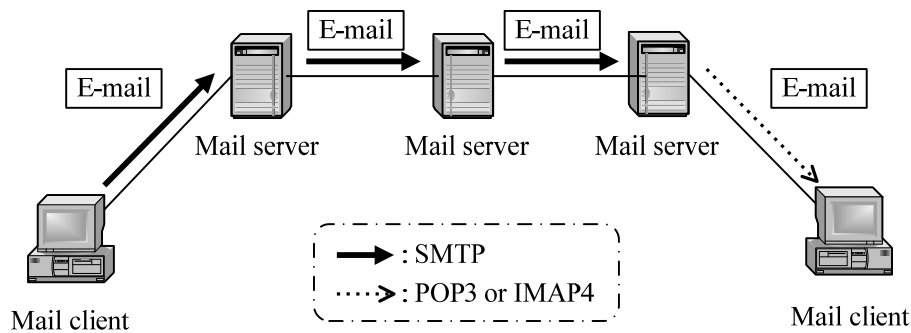
Various services are provided over the Internet. The typical Internet services are as described below.

(1) Electronic mail (e-mail)

An **electronic mail** is an electronic letter that is sent to a specific partner. In addition to text (i.e., characters) data, image data and voice data can also be sent by using a protocol called **MIME (Multipurpose Internet Mail Extensions)**. MIME uses an encoding method called **BASE64** in which data is separated into six-bit portions and converted to 64 types of alphanumeric characters. An electronic mail that makes use of the Internet is also called an “e-mail.”

When an e-mail is sent, the e-mail is first sent to the **mail server** to which the **mail client** of the transmitting side is connected. The mail server determines the destination mail server from the mail address and selects the most appropriate path. (In certain cases, the e-mail may pass through another mail server before the e-mail reaches the mail server of the other party.) When the e-mail reaches the mail server of the other party, the mail client of the receiving side can acquire the e-mail from the mail server at any desired timing. (Since the e-mail is sent through the store-and-forward switching method, the sending order and the receiving order may differ when there is a transmission delay.)

The e-mail transfer protocol **SMTP (Simple Mail Transfer Protocol)** is used to transfer an e-mail to the mail server, or to transfer an e-mail between mail servers. On the other hand, in order to acquire an e-mail from the mail server, mail reception protocols such as **POP3 (Post Office Protocol version 3)**, which acquires the entire information within the mail server in a batch, or **IMAP4 (Internet Message Access Protocol version 4)**, which enables the selection of the e-mail by initially acquiring only the header information, are used. IMAP4 is effective in e-mail usage in a mobile environment.



After an e-mail is sent, the e-mail may be returned along with a message “Returned e-mail: ... unknown”. In such a case, if “Host” is unknown, check the portion after the @ mark of the mail address because the “mail server of the destination is not found,” and if “User” is unknown, check the portion before the @ mark of the e-mail address because the “the user of the destination is not found.”

(2) Web / WWW (World Wide Web)

Web/WWW is a mechanism (i.e., service) that enables the acquisition of information (e.g., web pages) that is registered on the **web server** by an individual or company through a simple operation.

In most cases, the web pages that are published on the web server are created by a markup language, such as HTML and XML. A web page is hypertext (or hypermedia) in which reference information (**hyperlink**) that can be accessed on the related web page is embedded. Therefore, in order to acquire a web page from the web server, the **web client** uses **HTTP (HyperText Transfer Protocol)**. The acquired web page can be browsed through with the help of software called a **browser** that exists in the web client.

Generally, a **URL (Uniform Resource Locator)** is used to reference a web page. The URL is configured by the “Protocol name://Host name + domain name/File name” (e.g., “http://www.its.co.jp/map.html”). The URL shows the location of the files that are accumulated by the web server on the Internet, and is used through conversion to the IP address.

[Web-related technology]

- **CGI (Common Gateway Interface)**

This is a mechanism by which the web server starts an application program (i.e., CGI program) according to the processing request from the browser.

- **SSI (Server Side Include)**

This is a mechanism by which the process embedded in a web page (i.e., HTML) is executed on the web server, and the result is sent to the web client.

- **cookie**

This is a mechanism of writing to the browser the user information (e.g., IP address, user name) and date and time of establishing the last session, in order to identify the user from whom an attempt to access the web server is made. By using this mechanism, service can be provided by identifying the user even after a session has been released.

- **Web beacon**

This refers to a small image embedded in a web page that is used to collect information, such as the access trend of the user. It is used together with a cookie.

- **RSS (RDF Site Summary)**

This is a document format in which metadata, such as the heading and abstract of a page and the update time, is structured and described in order to effectively collect and deliver information from the web server. The update information created in the RSS format is called a **feed**, and can be collected easily through software called **RSS reader**.

- **Ajax (Asynchronous JavaScript + XML)**

This is a mechanism of asynchronously transmitting XML documents and dynamically redrawing screens.

(3) Search engine

A **search engine** is a program that performs the search process on a search site that is a collection of web pages for searching information. The information search service that is provided by the information search site is included as one of the web services.

The mechanism of information search that is provided by the search engine is broadly classified into three types.

- **Full text search**

The search string (i.e., keyword) is searched from all documents of the web page.

- **Directory type**

Search is performed by using a hierarchical index (i.e., directory) that is created manually.

- **Robot type**

Search is performed from the information that is accumulated and analyzed by using software (i.e., **crawler**) that automatically collects the information on the Internet and saves it as the database.

(4) File transfer service

The **file transfer service** is a service that **downloads** files to the client from an **FTP server**, and **uploads** files to the FTP server from the client. The URL is used for specifying the address in the file transfer service as well. However, since **FTP (File Transfer Protocol)** is used as a protocol, the portion indicating the protocol name of the URL becomes “ftp”.

(5) Other services

The following services (or utilization methods) that make use of Internet technologies are also available.

- **Intranet**

This refers to the network within a company that uses the Internet technologies (or services). Installation is relatively easy, and the initial cost can also be reduced. When a connection is to be established between Intranets, it is necessary to consider using **VPN (Virtual Private Network)** and PVC (Permanent Virtual Circuit) of a packet switched network to prevent leakage of information.

- **Extranet**

This refers to the network that is used to establish an interconnection between the Intranets of several companies. It is used in **EC (Electronic Commerce)** and **EDI (Electronic Data Interchange)**.

- **Overlay network**

This refers to a logical network that is constructed on the Internet as the base platform. A network having better **QoS (Quality of Service)** than the Internet can be logically constructed and provided.

5 Network Management

Network management refers to the management of network operations. The purpose of network management is to improve the reliability, security, and efficiency of the network.

5 - 1 Network Operations Management

In **network operations management**, the following five types of management are implemented on a priority basis.

(1) Configuration management

The information on resources (e.g., devices) configuring the network is collected and managed. Configuration change of the network and **version management** are also included in this management.

(2) Fault management

Information concerning the faults that have occurred on the network is collected, measures are taken against faults, and cause analysis is performed. Fault management is useful in preventing recurrence and in preventive control of faults.

[General fault handling procedure]

1) Detecting the fault

The network information is collected, and the fault is detected.

2) Analyzing the failure

The details of the detected fault are analyzed and classified, and the cause of the fault is identified.

3) Fault handling

The recovery action is taken to eliminate the cause of the fault. Moreover, the information concerning the fault (e.g., the fault details, cause of the fault, measures) is recorded and is used in preventing recurrence and in preventive control of the fault.

(3) Security management

The network usage status (i.e., access status) is monitored, and information leakage because of unauthorized access or unauthorized use is prevented.

(4) Performance management

The performance information of the network is collected/managed by monitoring the amount of traffic on the network and measuring the data transfer time and process response time. At this time, if the network performance falls below the reference value, measures are taken to maintain the network performance (e.g.,

installing more lines and replacing devices).

(5) **Accounting management**

The usage status (i.e., access status) of the network is collected, and the accounting information of the users is managed.

5 - 2 Network Management Techniques

(1) Network management tools

A **network management tool** is used to actually collect the network information and analyze the collected information. The typical network management tools include the following.

- **LAN analyzer**

This is a device or a piece of software that monitors the packets and traffic on a LAN and makes a diagnosis of the failure. It is also equipped with a function to analyze and display packets, in order to perform monitoring and failure diagnosis, and it is necessary to take care to prevent its misuse.

- **Command program for network management**

This is a tool that enables checking of the network status by entering a command and executing a program.

- **ping**

This checks the connection status (i.e., communication status) of devices, and also performs response measurement or diagnosis. The connection status is confirmed by specifying the IP address of the network device whose connection status is to be checked, sending a message, and then receiving a reply from the target device.

- **ipconfig**

This checks the network setup status such as the IP address.

- **arp**

This checks the ARP table (i.e., correspondence table of the IP address and MAC address).

- **netstat**

This checks various types of statistical information concerning the network.

(2) SNMP (Simple Network Management Protocol)

SNMP is a simple network management protocol of TCP/IP. It manages the network by collecting and setting information between the **manager** operating at the **SNMP management station** and the **agent** operating at the management target, and also by using

SNMP trap from the management target. The devices to be managed provide their own information as **MIB (Management Information Base)**.

(3) Network OS

The **network OS** provides the files, printer sharing function, and security function that are necessary for network management. Since such functions were not available in the conventional OSs, network OSs such as **NetWare** developed by Novell, Inc. were used. Currently, OSs having these functions are common, and the general idea of a network OS has been disappearing.

Chapter 5 Exercises

Q1

When a file of 10^6 bytes is transmitted by using a 64 kbps line, approximately how long (in seconds) does it take to transfer the file? Here, the line utilization rate is 80%.

- a) 19.6 b) 100 c) 125 d) 157

Q2

Among the descriptions concerning devices that constitute a network, which of the following is an explanation concerning the CCU (Communication Control Unit)?

- a) It converts the digital signals in a computer to a format that is suitable for transmission.
- b) It dials the telephone number of the terminal in order to call the terminal.
- c) It performs conversion from a digital signal to an analog signal, and vice versa.
- d) It assembles or disassembles the data to be transmitted, or performs error control of the data.

Q3

During start-stop data transmission, character “T” (JIS 7-unit code: 1010100) is sent by using the error detection method on the basis of even parity. In this case, which of the following is the bit string at the receiving side when data is received properly? Here, transmission is in order of the start bit (0), the character bits from the lower-order bit to the higher-order bit, the parity bit, and the stop bit (1), and the received bits are written in order from the left.

- a) 0001010101 b) 0001010111
c) 1001010110 d) 1001010111

Q4

Which of the following is a communication service that achieves high-speed communication by omitting error control or transmission checks within the network on the assumption that a high-quality digital network is used?

- a) Circuit switching service
- b) Leased line service
- c) Packet switching service
- d) Frame relay service

Q5

Which of the following is an appropriate explanation of TDM that is one of the multiplexing techniques?

- a) It is a multiplexing technique that is used to change the frequency band in use.
- b) It is a multiplexing technique that is used to change the wavelength of the light in use.
- c) It is a multiplexing technique that is used to change the connection destination in units of time slots.
- d) It is a multiplexing technique that is used to assign a code to each user.

Q6

Which of the following is a method of establishing a data link by which a message is sent after the terminal which is about to send data issues a transmission request and confirms the receiving status of the destination?

- a) Contention
- b) Token bus
- c) Token ring
- d) Polling

Q7

Which of the following is an appropriate role of FCS in a frame that is transmitted through the HDLC procedure?

- a) It records the error control code of the frame.
- b) It records the code for identifying the start or end of the frame.
- c) It records the type and transmission order of the frame.
- d) It records the information for identifying the transmitting station or receiving station of the frame.

Q8

Which of the following is an appropriate description of ADSL?

- a) It performs high-speed data transmission with a different uplink and downlink speed by using an existing telephone line (i.e., twisted pair cable).
- b) It achieves sharing on one line by separating telephone voice and data with a terminal adapter (TA).
- c) It connects a notebook PC to the network via a cell phone that is connected to a cell phone line.
- d) It provides various communication services, such as telephone and ISDN, and data communication by laying optical fiber cables up to residences.

Q9

Which of the following is a technique that encodes audio information and achieves transmission in the form of a packet used on the Internet?

- a) EDI
- b) PIAFS
- c) VAN
- d) VoIP

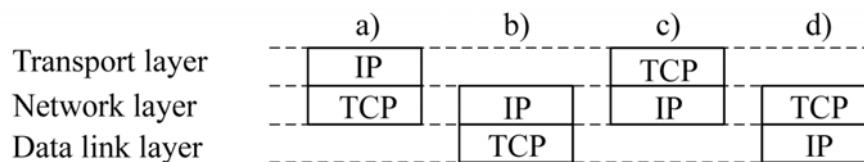
Q10

Which of the following is an appropriate explanation of the network layer in the OSI basic reference model?

- a) It provides routing and data transmission/relay function in order to achieve transparent data transmission from end to end.
- b) It provides a transparent transmission path to an upper layer by absorbing the differences in the features of physical communication media, and defining mechanical and electrical standards.
- c) It is the closest to the user (or application program) and provides the functions of file transfer and e-mail.
- d) It provides transmission control procedures (e.g., error detection, retransmission control) in order to achieve transparent data transmission between adjacent nodes.

Q11

Which of the following is an appropriate description of the relationship between the TCP/IP protocols that are used on the Internet and the layers of the OSI basic reference model?

**Q12**

Which of the following is an appropriate description of 10BASE5?

- a) The maximum transmission distance is 5 km.
- b) The transmission speed is 10 Mbps.
- c) The transmission medium is a twisted pair cable.
- d) The transmission method is the broadband method.

Q13

In the MAC (Media Access Control) in LAN, which of the following is the method that has the function of detecting data frame collision on the transmission medium?

- a) CSMA/CA
- b) CSMA/CD
- c) Token bus
- d) Token ring

Q14

Which of the following is an appropriate description of LAN-to-LAN connection devices?

- a) The gateway is used to convert the protocols of the lower layers (i.e., first layer through third layer) of the OSI basic reference model.
- b) The bridge relays frames on the basis of the IP address.
- c) The repeater extends the transmission distance by amplifying signals.
- d) The router provides the routing function on the basis of the MAC address.

Q15

Which of the following is the information that is included in the Ethernet header of the packet that is used in TCP/IP?

- a) IP address
- b) MAC address
- c) Sequence number
- d) Port number

Q16

Which of the following can be assigned to a computer as the IP address of class C?

- a) 192.0.0.255
- b) 192.0.256.16
- c) 192.128.0.0
- d) 192.128.0.128

Q17

Which of the following is a protocol for dynamically assigning an IP address to a computer that is connected to the network?

- a) DHCP b) DNS c) NAT d) RIP

Q18

Which of the following is an appropriate service that is provided by a RADIUS server?

- a) Converting FQDN to IP address
- b) Establishing a proxy connection to an external network
- c) Remotely accessing the network through a public network
- d) Centralizing the authentication of the overall system

Q19

Which of the following is an appropriate explanation of SMTP?

- a) It is a program for searching for the information that is stored in a web server.
- b) It is a protocol for transferring the information that is stored in a web server.
- c) It is a protocol for downloading file from a file server.
- d) It is a protocol for transferring an e-mail to a mail server.

Q20



Which of the following is an appropriate meaning of an Intranet?

- a) It is an interconnected network between several companies that is used in EC and EDI.
- b) It is a network within an organization that uses Internet technologies, such as the web server and the web browser.
- c) It is a logical network that is constructed on the Internet as the base platform.
- d) It is a network for transmitting data in units of variable length frames.

Q21



Which of the following is an appropriate role of the “ping” command, which is one of the network management tools?

- a) It checks the correspondence table of the IP address and the MAC address.
- b) It checks the network setup status such as the IP address.
- c) It checks the device connection status by sending a message specifying the IP address.
- d) It checks the status of the packets and the traffic on a LAN.



Chapter 6

Security



1 Overview of Information Security

Information systems and the Internet are becoming part of the infrastructure of modern society. As dependence upon IT increases, so does the importance of information security. Since measures implemented in information security are broadly divided into technological measures and management measures, this section discusses information security overall in terms of both technology and management.

1-1 Concept of Information Security

ISO/IEC 27000:2014 (JIS Q 27000:2014) describes information security as “preservation of confidentiality, integrity, and availability of information; in addition, other properties such as, authenticity, accountability, non-repudiation, and reliability can also be involved.”

Confidentiality	The property that information is not made available or disclosed to unauthorized individuals, entities, or processes
Integrity	The property of safeguarding the accuracy and completeness of assets
Availability	The property of being accessible and usable upon demand by an authorized entity
Authenticity	The property that an entity is what it claims to be
Accountability	The property that ensures that the actions of an entity may be traced uniquely to the entity
Non-repudiation	The property of proving that an action or event has taken place, so that this event or action cannot be repudiated later
Reliability	The property of consistent intended behavior and results

In information security, the following three objects to be managed and four management functions are given attention to maintain the confidentiality, integrity, and availability of information.

[Objects to be managed in information security]

Asset	Anything that has value to the organization
Threat	A potential cause of an unwanted incident, which may result in harm to a system or organization

Vulnerability	A weakness of an asset or group of assets that can be exploited by one or more threats
---------------	--

[Management functions of information security]

Prevention functions	Functions to prevent the occurrence of threats
Detection functions	Functions to discover and detect threats that have occurred
Minimization functions	Functions to minimize the damage from a threat that has occurred and to prevent its expansion
Restoration functions	Functions for prompt restoration from damage caused by threats

In addition, the **OECD (Organization for Economic Cooperation and Development)** recommends the following nine principles for participants (information system owners, providers, and users) in its “**Guidelines for the Security of Information Systems and Networks.**”

- (1) **Awareness**
Participants should be aware of the need for security of information systems and networks and what they can do to enhance security.
- (2) **Responsibility**
All participants are responsible for the security of information systems and networks.
- (3) **Response**
Participants should act in a timely and co-operative manner to prevent, detect, and respond to security incidents.
- (4) **Ethics**
Participants should respect the legitimate interests of others.
- (5) **Democracy**
The security of information systems and networks should be compatible with essential values of a democratic society.
- (6) **Risk assessment**
Participants should conduct risk assessments of information systems and networks.
- (7) **Security design and implementation**
Participants should incorporate security as an essential element of information systems and networks.
- (8) **Security management**
Participants should adopt a comprehensive approach to security management.

(9) Reassessment

Participants should review and reassess the security of information systems and networks, and make appropriate modifications to security policies, practices, measures, and procedures.

For understanding of the concept of information security, objects to be managed by information security are detailed in order.

1-1-1 Assets

Assets are defined as anything that has value to be protected by the organization. Among these, assets that involve important information in particular are known as **information assets**.

[Types of information assets]

- **Tangible information assets**

These are information assets with tangible form, including hardware assets such as computers and communication equipment, and software assets such as business software, system software, and documents.

- **Intangible information assets**

These are information assets without tangible form, including some types of information such as customer information, sales information, intellectual property-related information, and personnel information, and other types of information such as the reputation and image of an organization.

1-1-2 Threats (or Perils)

Threats (or perils) are things which may cause loss to information assets. Examples of threats that pertain to the Internet and other networks include the following.

- **Tapping**

The interception of data by a third party with malicious intent

- **Falsification**

The fraudulent rewriting of information in e-mail or web pages

- **Spoofing**

The performance of fraudulent actions by impersonating another person (e.g., authorized user)

- **Theft**

The theft of files or data by a third party with malicious intent

- **Destruction**

The fraudulent destruction or erasure of files or data

Threats are classified into three types as follows:

- **Personal threat**

This is the type of threat that is caused by human behavior (with or without malicious intent).

- **Technological threat**

This is the type of threat in which a third party with malicious intent uses computer technology to make attacks.

- **Physical threat**

This is the type of threat against equipment itself or against the buildings in which equipment is located.

Specific examples of each type of threat are presented below. A case that corresponds to more than one type of threat is included among the type of threat that displays its characteristics most strongly.

(1) Personal threats

Typical examples of **personal threats** include the following. In particular, the acts that are recognized as fraudulent are sometimes termed **fraudulent behavior**.

- **Information leakage**

This is the leakage of information to a third party. It includes intentional leakage with the aim of receiving payment for information provision, and unintentional leakage of important information accidentally overheard by a third party. In addition, information in discarded equipment may be restored and leaked if not physically deleted (i.e., destroyed).

- **Loss / Theft / Damage**

This means that IT devices, such as PCs and USB memory, where information is stored are left behind, stolen, or destroyed during use.

- **Error / Incorrect operation**

This is data erasure or such other error that is caused by wrong operation. It includes the leakage of important information through mistaken entry of recipient e-mail addresses.

- **Social engineering**

This is the act of stealing information through everyday and common means.

- **Trashing (scavenging, dumpster diving)**

This is the act of stealing important information from memos thrown away in the garbage bin, data left in memory or cache, etc. It is also used as a method of **foot printing** for prior collection of information about the target of attacks.

- **Spoofing**

This is the impersonation of a person by a third party. The spoofer may pretend to be a customer or a supervisor in order to ask for PINs (PIN Numbers) or passwords.

- **Peeping**

This is the act of sneaking a peek at keyboard operation of a person who is entering a password, or classified information displayed on another person's screen. In particular, the act of sneaking a peek at information over a person's shoulder is called **shoulder hacking**.

- **Cracking**

This is the act of intruding into another person's PC with malicious intent, to steal or destroy data. A person who engages in cracking is called a **cracker**. Note that the software package used by a cracker after unauthorized intrusion is called a **rootkit**, and the path installed to facilitate later intrusion is called a **back door**.

- **Targeted attack**

This is the act of attacking a specific organization or person as a target. Since humans select the target of the attack, this is classified as a personal threat. However, the attack method itself is primarily classified as a technological threat.

(2) Technological threats

Typical examples of **technological threats** include **unauthorized access** or **denial of service** using computer technology, and **computer crime**. The **compromise** of safety due to advances in computer technology can also be called a technological threat.

Typical attack methods using computer technology, which are classified as technological threats, include the following.

- **DoS attack (Denial of Service)**

This is an attack that sends a large amount of data continually to the target server to place an excessive load on the server's CPU and memory, and thereby obstructs service. In addition, there is also a **DDoS (Distributed DoS)** attack in which malicious programs used for targeted attacks are used to attack the single target all at once from multiple PCs.

- **Key logger**

This is an attack that uses the mechanism (e.g., software) that records keyboard input, and fraudulently acquires information (e.g., password) entered by another person.

- **Clickjacking**

This is an attack that sets up a web page with some sort of function that causes a user's click to execute operations not intended by the user.

- **Phishing**

This is an attack that leads a user to a fake website through means such as e-mail pretending to be sent from a real company (e.g., financial institution), and defrauds the user of the credit card number, a bank account number, a PIN, and other personal information.

- **Cache poisoning**

This is an attack that fraudulently overwrites cache information. In particular, **DNS cache poisoning**, which overwrites DNS cache, is used to lead users to fake websites for phishing.

- **IP spoofing**

This is an attack that sends packets to another party with the source IP address disguised. This is used in actions including leading users to fake websites for phishing.

- **XSS (Cross Site Scripting)**

This is an attack where a vulnerable target website is used as a stepping stone; a malicious script is sent to a user who is accessing the target website, and then executed on the user's browser to enable the theft of information.

- **CSRF (Cross Site Request Forgery)**

This is an attack which, when a user is logged in to a website and then accesses another website that has a trap installed, causes a malicious request to be sent to and executed by the logged-in website in the guise of a request from the user (i.e., as a forgery).

- **Session hijacking**

This is an attack that takes over a session (i.e., a series of communications between specified parties) during communication between correctly authorized users.

- **Directory traversal**

This is an attack that accesses normally undisclosed directories (or files) by appending "../" to file names, to traverse upward through directories.

- **Drive-by download**

This is an attack that causes a user to download a malicious program, without permission during website browsing.

- **SQL injection**

This is an attack that falsely modifies a database or fraudulently obtains information by providing part of an SQL statement as a parameter to a program (CGI program) in the website that is linked to the database.

- **Side channel attack**

This is an attack that obtains confidential information by measuring and analyzing some additional information (i.e., side channel information), such as the electric power consumption or radiated electromagnetic waves of active IC chips.

- **Zero-day attack**

This is an attack that takes advantage of a vulnerability in software before a fix for the vulnerability can be released by the software vendor.

- **Password cracking**

This is an attack that fraudulently decodes or otherwise obtains the password of a true user.

- **Dictionary attack**

This is a method that uses a file (i.e., a dictionary file) that contains character strings likely to be used as passwords, to try such words in sequence.

- **Brute force attack**

This is a brute-force method that attempts every combination of characters. It is used as an attack method of performing the exhaustive search for a decryption key.

- **Third-party relay**

This is an attack that abuses a freely usable server (e.g., mail server) as a “stepping stone” to transmit e-mail and other data.

- **Gumblar**

This is an attack that falsifies the website of a famous company or public institution, and infects the computer of a user who is browsing the falsified website with a computer virus.

- **Buffer overflow**

This is an attack that continually sends long character strings or such other data to flood the memory area (i.e., buffer) secured by a program, for the purpose of seizing access privileges to the program and creating malfunctions.

Fraudulent programs (i.e., **malware**) created with malicious intent are also classified as technological threats.

The following are typical examples of malware.

- **Computer virus**

In the Japanese Ministry of Economy, Trade and Industry's "Standards for Measures Against Computer Viruses," a computer virus is defined as "a program that is created to intentionally cause some form of damage to third parties' programs or databases, and that has one or more of the following functions."

Self-infecting function	Viruses make copies of themselves to infect other systems.
Concealment function	Viruses do not reveal symptoms until the onset of their action.
Onset function	Viruses perform actions not intended by designers, such as destruction of data.

However, in general at present, file-infecting viruses that infect specific files are called computer viruses (in a narrow sense).

- **Boot sector virus**

This virus infects the boot sector (i.e., the system area that contains the boot program) that is read before an OS starts up.

- **Program file virus**

This virus infects the executable program files such as applications.

- **Interpreter virus**

This virus infects non-executable files, such as data files, other than program files. It includes two types of viruses: a **macro virus** that infects through the macro functions of application software, and a **script virus** that infects through a scripting language like JavaScript or VBScript.

- **Worm**

A worm proliferates by duplicating itself on other computers through networks, without the need for a program to be infected. It often spreads a copy of itself automatically as an e-mail attachment file, or uses networks to continue spreading infection.

- **Bot**

This is a program that is created for the purpose of controlling infected computers from outside via networks (e.g., the Internet).

- **Spyware**

This is a program that illicitly obtains a user's information, such as personal information and access histories, and automatically sends such information to another party other than the user.

- **Trojan horse**

This is a malicious program that pretends to be useful software but causes damage to

users. While a Trojan horse does not infect files nor self-propagate, the concealed virus is delivered to a PC to transmit private files on the PC over the Internet, destroy the contents of files or disks, or otherwise cause damage.

The following computer crimes are also said to be types of technological threats.

- **Salami technique (Salami slicing)**

This is a method of repeatedly stealing assets little by little so that they are negligibly small when taken as a whole. An example is a technique that collects money from a bank account into another account, in fractions of less than one yen.

- **One-click fraud**

This is a type of fraudulent act; for example, clicking an image or link on matchmaking or adult websites causes an unfair fee to be charged.

- **Phishing fraud**

This is a general name for the act of phishing, or for fraudulent acts committed using information obtained illicitly through phishing.

(3) Physical threats

The following are typical examples of **physical threats**.

- **Disaster**

This means that equipment or buildings are made unusable, or equipment itself is lost, due to a natural disaster (e.g., earthquake, flood) or a human disaster (e.g., fire).

- **Destruction**

This means that equipment or buildings are made unusable, due to sabotage or destructive acts by a third party with malicious intent.

- **Accident / failure**

This means that equipment or buildings are made unusable, due to unforeseen accidents or failures.

- **Unauthorized intrusion**

This means that unauthorized persons intrude into buildings or rooms in which equipment is located.

1-1-3 Vulnerabilities (or Hazards)

Vulnerabilities (or hazards) are weaknesses or flaws that are exploited by threats, becoming

the cause of even greater threats. A variety of vulnerabilities in equipment, technologies, management, and many other areas cause problems.

- **Security hole**

This is a vulnerability of software or systems that is caused by software design flaws, bugs, etc.

- **Man-made vulnerability**

This is a vulnerability that is caused by human behavior, due to a lack of enforcement or preparation of a code of conduct for companies, organizations, and people.

1 - 2 Information Security Technology

Information security technology is computer technology used within the technological measures that are implemented as information security measures.

1-2-1 Cryptography

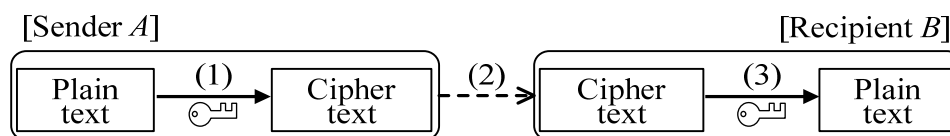
Cryptography is encryption technology implemented as measures against information leakage or measures against tapping of communications. Terminology that pertains to cryptography is defined as follows:

Term	Meaning
Plain text	Data in an unencrypted state
Cipher text	Data in an encrypted state
Encryption	The conversion of plain text to cipher text
Decryption	The conversion of cipher text to plain text
Encryption key / decryption key	Special data used in encryption/decryption
Key length	The length (usually in bits) of an encryption key or decryption key A longer key length is more difficult to decode.
Decoding	The obtaining of plain text from cipher text, by improper means
Encryption algorithm	An algorithm (i.e., program) that performs encryption / decryption
Encryption strength	The degree to which cipher text is difficult to decode

(1) Common key cryptography

Common key cryptography (also known as symmetric key cryptography or secret key cryptography) is a method for performing encryption/decryption between parties that exchange data, using a **common key**. This encryption method performs both encryption and decryption using the same key, and therefore, the parties must in advance have a common key that is kept secret from third parties.

[Common key cryptography procedure]



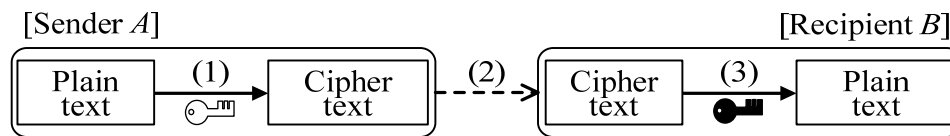
- (1) Sender *A* encrypts plain text by using a common key.
- (2) Sender *A* sends the cipher text (i.e., encrypted plain text) to Recipient *B*.
- (3) Recipient *B* decrypts the cipher text by using the common key.

Typical methods of common key cryptography include **AES (Advanced Encryption Standard)** from the NIST (National Institute of Standards and Technology) which adopted the Rijndael method, **TripleDES** which repeats DES three times, and **IDEA (International Data Encryption Algorithm)** which repeats several rounds of the XOR (exclusive OR), addition, and multiplication operations.

(2) Public key cryptography

Public key cryptography is a method that uses a pair of **private key** and **public key**. The private key is concealed by its owner, while the public key is disclosed or distributed, and is made usable by anyone. This cryptography can encrypt plain text by using one key, and can decrypt the cipher text with the other key. However, this does not mean that the public key is always used for encryption and the private key for decryption; rather, they are used according to purpose (see details later). In order to prevent decryption of encrypted text by third parties when measures against information leakage or measures against tapping of communications are taken, encryption is performed with the recipient's public key, and decryption with the recipient's private key.

[Public key cryptography procedure]



- (1) Sender *A* encrypts plain text by using Recipient *B*'s public key.
- (2) Sender *A* sends the cipher text (i.e., encrypted plain text) to Recipient *B*.
- (3) Recipient *B* decrypts the cipher text by using Recipient *B*'s private key.

Typical methods of public key cryptography include **RSA** (from the initials of the three developers, Rivest, Shamir, and Adleman), **ElGamal encryption** which applies the discrete logarithm problem, and **Elliptic Curve Cryptography** which uses elliptic curve equations.

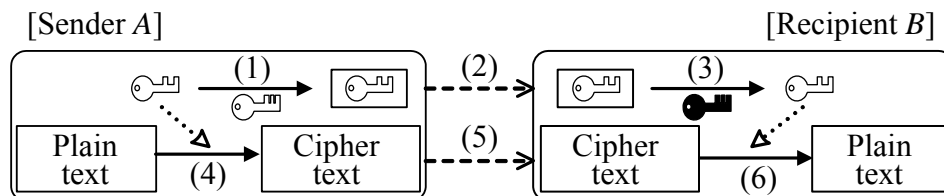
The strengths and weaknesses of common key cryptography and public key cryptography can be summarized as follows:

	Common key cryptography	Public key cryptography
Strengths	<ul style="list-style-type: none"> For the same key length, common key cryptography has higher encryption strength than public key cryptography. Common key cryptography requires less time for encryption/decryption than does public key cryptography. 	<ul style="list-style-type: none"> Management and transfer of keys is easy. (In encrypted communication among n persons, the number of keys is n pairs [the number of types of keys is $2n$].)
Weaknesses	<ul style="list-style-type: none"> Common keys may be leaked under long-term tapping. Management and transfer of keys is difficult. (In encrypted communication among n persons, the number of types of keys is $n(n-1)/2$.) 	<ul style="list-style-type: none"> For the same key length, public key cryptography has lower encryption strength than common key cryptography. Public key cryptography requires more time to process due to complex encryption/decryption algorithms. Public key cryptography may allow spoofing.

(3) Session key cryptography (hybrid cryptography)

Session key cryptography (hybrid cryptography) uses public key cryptography for the transfer of keys, which has a weakness of common key cryptography, and performs encrypted communication using common key cryptography.

[Session key cryptography procedure]



- (1) Sender A encrypts a generated common key by using Recipient B's public key.
- (2) Sender A sends the encrypted common key to Recipient B.
- (3) Recipient B decrypts the encrypted common key by using Recipient B's private key.
- (4) Sender A encrypts plain text by using the common key.
- (5) Sender A sends the cipher text (i.e., encrypted plain text) to Recipient B.
- (6) Recipient B decrypts the cipher text by using the common key.

In session key cryptography, common keys are generated using means such as a **hash function** (i.e., a function that obtains a unique, fixed-length output value from an input value) which is based on the communication session number or a random number. The generation of a common key is performed for each encrypted communication session, and the key is transferred using public key cryptography. For that reason, damage is limited even if the common key is leaked. In addition, since only one pair of public key and private key usually needs to be managed, management of keys is simple, and encrypted communication can be carried out at high speed by using common key cryptography. In other words, this approach can be said to combine the strengths of common key cryptography and public key cryptography. However, even session key cryptography is not able to eliminate the possibility of spoofing.

(Block cipher mode of operation)

Block cipher mode of operation defines the use of a **block cipher** that performs encryption in units of fixed-length blocks. (The cipher used in the method of encrypting in units of bits or bytes is called a **stream cipher**.) For example, AES of common key cryptography adopts a 128-bit block cipher, while IDEA adopts a 64-bit block cipher.

In block cipher modes of operation, there are two major types of modes: one mode is used for concealing messages (encryption method), and the other mode for message authentication (message authentication code). For example, ECB mode is a mode of operation for concealing messages. When this mode is used to encrypt multiple blocks, there is the possibility that the same cipher text may be generated from the same plain text, and the contents may be guessed. For this reason, another mode such as CBC mode, which generates differing cipher text even when the same plain text is encrypted, is recommended.

1-2-2 Authentication Technique

(1) User authentication

User authentication is an authentication technique that includes the process of verifying a user's identity. It is used as a measure against the impersonation of authorized users by malicious third parties.

In user authentication, there are a variety of methods according to purpose and application.

(1) User ID/Password

A **user ID** is an identifier that is assigned to an individual user, while a **password** is a character string (i.e., a keyword) that is registered in advance. During the login process, this technique searches for a registered password corresponding the user ID entered by the user. If the registered password matches the password entered by the user, the user is authenticated as a valid user who knows the password.

When this technique is used, it is necessary for users to thoroughly follow password management and password requirements to prevent any abuse of passwords.

[Points to note for password management]

- Use a meaningless character string that combines alphanumeric characters, symbols, etc.
- Make the password as long as necessary. (at least 6 or 8 characters)
- Do not use the same password for a long time. Change it regularly.
- Do not reuse passwords (i.e., do not use the same password for multiple authentications).

- Do not record passwords where they will be seen. (Jotting down passwords does not need to be prohibited, but this should be properly managed).
- Do not share passwords with others nor give passwords to others.
- If a password is forgotten, leaked, or no longer needed, promptly contact the administrator and take required actions (e.g., invalidation of password)
- To prevent theft (or leakage), encrypt a password (or convert it to a hash value) and record the password.

(2) IC card

This is a technique that authenticates users through plastic IC cards with embedded IC chips that are able to record information. It is used for commuter passes, employee ID cards, etc. When an IC card is used, the user may be asked to enter the **PIN code (Personal Identification Number)** recorded in the IC chip in order to confirm that the user is the rightful holder of the card.

IC cards have over 100 times more storage capacity than magnetic cards and enable data encryption to make the cards resistant to forgery. The cards also have an advantage in **tamper resistance** to prevent unauthorized access or falsification.

(3) OTP (One-Time Password)

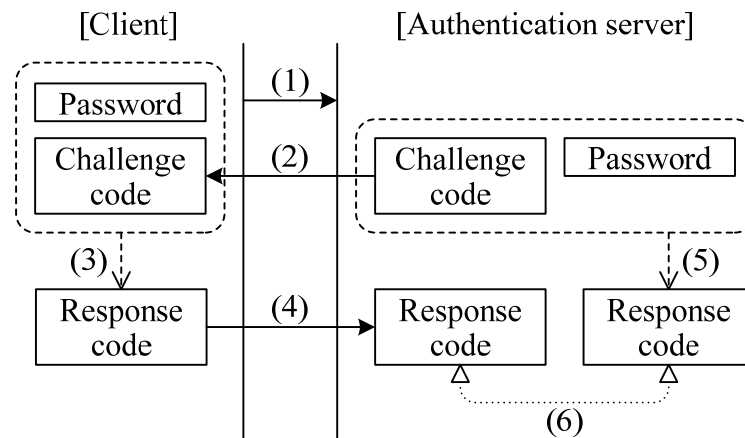
This is a password-based user authentication technique that is primarily used by computers connected to communication lines. By using a different password for each login, this technique avoids the risk of password leakage.

Typical one-time password methods are as follows:

- **Challenge-response authentication**

This method does not transmit passwords directly over communication lines. Instead of a password, the method uses a response code that is generated from both the password and a challenge code (a random number) that differs for each access request, in order to authenticate a user.

In some cases, a one-time password generation function is built into the IC card, which is used together with PIN code-based authentication.



- (1) The client sends a user ID and requests access.
- (2) The authentication server generates and sends a challenge code.
- (3) The client generates a response code from its own password and the received challenge code.
- (4) The client sends the response code to the authentication server.
- (5) The authentication server generates another response code from both the password of the user indicated by the user ID and the challenge code that was sent.
- (6) The authentication server compares the received response code with the response code generated in (5), and authenticates the client if those two response codes are the same.

- **Time synchronous authentication**

This method uses cash card-sized dedicated client hardware that displays a PIN code that changes at set time intervals. The code is attached after the time and user ID, and sent along with them. Since the dedicated client hardware and authentication server are synchronized, the login is approved if the PIN code sent by the client is the same as that of the server.

(4) **Biometric authentication**

This authentication technique identifies an individual (i.e., a user) by using physical information or behavioral information (e.g., speed or pen pressure used in a signature) that is unique to the individual, and such information is registered in advance on the authentication system. In addition to managing entry into secure areas, this authentication is also used by bank ATMs and individual PCs.

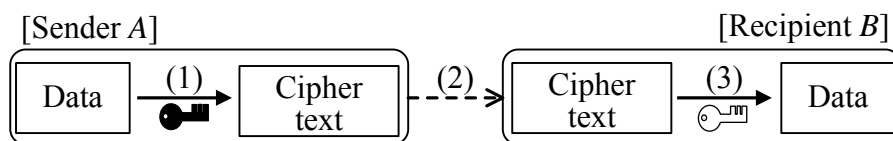
Although it is effective in preventing spoofing, it requires special authentication systems and continues to present issues in areas such as authentication precision (FRR (False Rejection Rate) and FAR (False Acceptance Rate)), difficulty of substitution, adaptation to physical changes over time, and personal information protection.

Name	Authentication information
Face authentication	Facial characteristics (e.g., positional relationships among eyes, nose, and mouth)
Iris authentication	Pattern shape, shading, etc. of the iris (the folds emanating from the pupil) of the eye
Voice authentication	Characteristics of the voice wave pattern
Palm authentication	Width of the palm, length of the fingers, etc.
Vein authentication	Branching angle, length, etc. of veins * This is sometimes included in palm authentication.
Fingerprint authentication	Characteristics (called “minutia points”) of fingerprints (i.e., the pattern formed by ridges on fingertips)

(5) Public key cryptography

This is an authentication technique that uses private keys that are kept secret by the holders.

[Authentication procedure in public key cryptography]



- (1) Sender *A* encrypts data by using Sender *A*'s private key.
- (2) Sender *A* sends the cipher text (i.e., encrypted data) to Recipient *B*.
- (3) Recipient *B* decrypts the cipher text by using Sender *A*'s public key.
 - When decryption is successfully performed, it is confirmed that “*A*” is the sender.

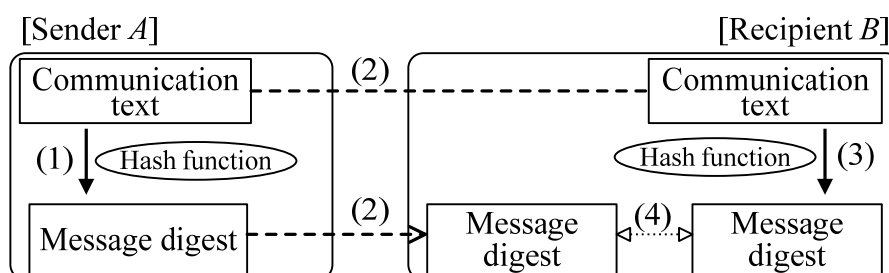
Step (3) is the key point in user authentication. The fact that Recipient *B* successfully performed decryption using Sender *A*'s public key means that the cipher text had been encrypted using *A*'s private key (because cipher text encrypted from plain text by using one of the key pairs can only be decrypted using the other key). Since only *A* has (i.e., conceals) *A*'s private key, this cipher text cannot be created by any party other than *A*. In other words, it is confirmed (i.e., authenticated) that the sender is unquestionably *A*. However, it is necessary to note that this authentication works well on the assumption that *A*'s public key, which *B* has, must belong to *A* without doubt, and its paired private key must be owned by *A* alone.

(2) Message authentication

Message authentication is an authentication technique that confirms that data is not improperly overwritten. It is used as a measure to prevent falsification involving improper overwriting of data.

As a typical message authentication method used in data communication, there is a method of calculating a message digest from the communication text (i.e., the data) by using a **hash function**.

[Message authentication procedure]



- (1) Sender A calculates a message digest from the communication text.
- (2) Sender A sends the communication text and its message digest to Recipient B.
- (3) Recipient B calculates a message digest from the received communication text.
- (4) Recipient B compares the received message digest with the message digest generated in (3), and if the two message digests are the same, it is confirmed that the communication text is not falsified (nor tampered with) during communication.

• Hash function

This is a function that obtains an output value (i.e., a hash value) of fixed length from an input value of arbitrary length. It has the property (often referred to as the one-way property) that the same output value is obtained from the same input value, and the input value cannot be obtained from the output value.

Hash function name		Length of hash value	Standardization/normalization
SHA-1		160 bits	Standardized by NIST
SHA-2	SHA-256	256 bits	Standardized by NIST as the successor to SHA-1
	SHA-512	512 bits	
MD5		128 bits	Standardized as RFC 1321

On the basis of the property of a hash function, if there is even a small difference in the communication text between the sender side and the recipient side, the message digests that

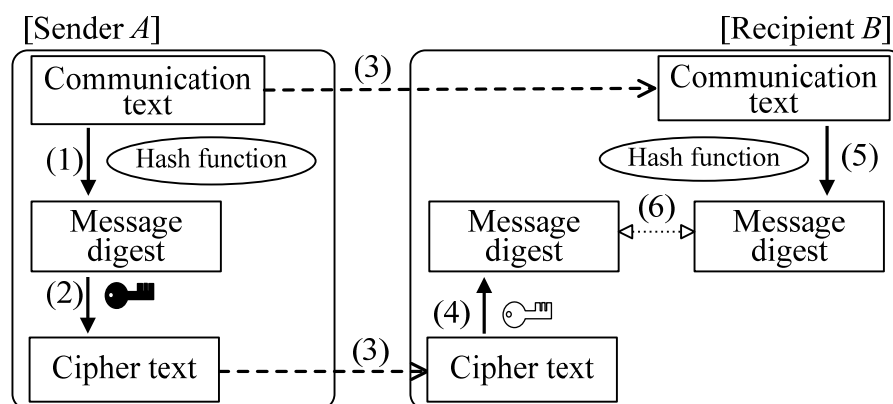
are delivered to and generated by the recipient side will be different, and as a result, falsification can be detected.

However, when an already-known hash function is used, there is a possibility that both the communication text and the message digest can be falsified so as not to be detected. **MAC (Message Authentication Code)** is a technique that can resolve this issue. This method uses a key (i.e., a common key) owned by the sender and the recipient, and also uses a MAC value, which is calculated from the communication text plus the common key, as a message digest. (As a method for calculating the MAC value, a block cipher mode of operation or a hash function is used.) When this method is used, a third party who does not know the common key cannot calculate the correct MAC value, and therefore, the level of security is enhanced.

(3) Digital signature

Digital signature is an authentication technique that combines user authentication and message authentication using public key cryptography. (To distinguish digital or other electronic-based signatures from regular signatures, these are sometimes called **electronic signatures**.)

[Digital signature procedure]



- (1) Sender A calculates a message digest from the communication text.
- (2) Sender A encrypts the message digest by using Sender A's private key.
- (3) Sender A sends the communication text and its cipher text to Recipient B.
- (4) Recipient B decrypts the cipher text by using Sender A's public key.
- (5) Recipient B calculates a message digest from the received communication text.
- (6) Recipient B compares the message digest decrypted in (4) with the message digest calculated in (5). If these two are the same, it is confirmed that "A" is the sender, and in addition, the communication text is not falsified.

A typical digital signature method is **DSA (Digital Signature Algorithm)** developed by NIST

(National Institute of Standards and Technology) as a US government standard. DSA is an improved version of ElGamal signature, which uses the ElGamal encryption scheme known as one of the public key cryptography standards, and generates message digests using SHA-1 or SHA-2. It is noted that SHA-1 is being migrated to SHA-2 (e.g., a general term for SHA-256, SHA-512).

Another method is the W3C-recommended **XML signature**, which defines a digital signature syntax for XML documents. XML signatures can be attached to communication text like normal signatures, and can also be attached to specified elements or content.

In digital signatures, third parties (private companies) known as certification bodies guarantee the validity of public and private keys on which user authentication by public key cryptography is predicated. These certification bodies are also used for specified purposes (e.g., public key cryptography) other than digital signatures.

Certification bodies are composed of third-party organizations, such as **CA (Certification Authority)**, **RA (Registration Authority)**, and **VA (Validation Authority)**, which issue **digital certificates** to users and lower-level certification authorities in order to guarantee the validity of public keys and digital signatures. In addition, in order to certify the validity of certification bodies themselves and distribute public keys, they issue **root certificates** signed with their own private keys.

- **Digital certificate (public key certificate)**

This is an electronic certificate for an individual or organization, which is issued by a certification authority. It contains the signature algorithm (hash function), encryption method, expiration date, public key of the certified party, and other information. In addition, it is encrypted with the private key of the certification authority so that users can confirm that it was issued by the certification authority.

The decryption key (i.e., the public key) for a digital certificate can be obtained from the root certificate released by the certification authority. The following is the standard specification for digital certificate released by ITU-T (International Telecommunication Union — Telecommunication Standardization Sector) X.509.

Item name	Content
Version information	v1–v3
Serial number	Certificate number
Signature algorithm	The type of algorithm used for signatures: SHA-1 or MD5 (hash function), RSA (public key), etc.

Issuer name	Issuing body (usually the name of a certification authority)
Period of validity	Starting date and time and ending data and time (from several seconds to hundreds of years)
Certifier	Subject name (holder of certificate)
Public key information	Certifier's public key information

- **CRL (Certificate Revocation List)**

This is a list of digital certificates that must be revoked due to leakage or loss of keys, even during the period of validity. It contains the serial numbers and expiration dates of digital certificates, and is made publicly available in the repository of a certification authority. Users can confirm the validity of a digital certificate by searching the CRL.

- **OCSP (On-line Certificate Status Protocol)**

This is a protocol to confirm the validity of a digital certificate on an online real-time basis. OCSP is defined in RFC 2560 of the IETF (Internet Engineering Task Force). OCSP servers are operated by CAs (Certification Authorities) and by VAs (Validation Authorities) which centrally manage CRLs. By confirming digital certificates with OCSP servers, OCSP clients can reduce the burden of CRL acquisition and collation. However, OCSP only confirms the revocation status of a digital certificate, and therefore, the period of validity and other information must be confirmed on the client side.

CAs (Certification Authorities) include two types of CAs: **public CA** that issues certificates for external web services, and **private CA** that takes on a role within a closed environment, such as a specified organization.

(4) Other certification techniques

(1) **Time authentication (timestamp authentication)**

This is a certification technique by which a third-party time stamping authority (TS authority, which may be a certification authority) guarantees the existence and authenticity of electronic documents, in accordance with the Japanese Act on Utilization of Telecommunications Technology in Document Preservation, etc. Conducted by Private Business Operators, etc. of 2005. Normally, a creator's digital signature and digital certificate are attached to an electronic document.

[Authentication procedure in time authentication]

- 1) The user generates a message digest from an electronic document.
- 2) The user sends the message digest to a time stamping authority.
- 3) The time stamping authority generates a TS token that stamps the received message digest with time information, and securely stores a duplicate.
- 4) The time stamping authority encrypts the TS token by using its own private key.
- 5) The time stamping authority sends the encrypted TS token and its own digital certificate to the user.

<The following are steps to confirm the electronic document's existence and authenticity.>

- 6) The checker obtains the public key from the digital certificate of the time stamping authority.
- 7) The checker decrypts the encrypted TS token by using the obtained public key.
- 8) The checker compares the TS token's message digest with the message digest obtained from the original electronic document, to confirm that there is no falsification. In addition, the checker looks up the time information and confirms the existence of the electronic document.

(2) IEEE 802.1X

This is a client authentication technique used for LANs. It is implemented using a **RADIUS server** or other authentication server, and authentication software (i.e., **supplicant**).

(3) CAPTCHA (image authentication)

This is an authentication technique that requires users to identify a distorted image of letters or numbers displayed on the screen and then enter the corresponding correct characters, in order to confirm that they are human beings. It is used to prevent automatic posting by programs and such other automated processes.

1-2-3 PKI (Public Key Infrastructure)

PKI (Public Key Infrastructure) is a security platform using public key cryptography, certification bodies, and digital signatures. It is used in EC (Electronic Commerce) and in the **SSL (Secure Sockets Layer)** security protocol. Another authentication infrastructure is **GPKI (Government PKI)**, which uses PKI for performing applications and notifications to administrative bodies. Since mutual authentication between government agencies'

certification authorities and private certification authorities is required in GPKI, a **BCA (Bridge Certification Authority)** must mediate between such certification authorities.

1-3 Information Security Management

Information security management refers to the actions of analyzing and evaluating diverse risks pertaining to information assets (e.g., physical assets, software assets, intangible information assets), and planning and implementing appropriate security measures. This subsection discusses information security management, information security evaluation and certification scheme, and risk management in companies and other organizations.

1-3-1 Information Security Management

Information security management refers to the actions of clearly defining an organization's perspective to consider information security, and actually implementing and managing it.

(1) Information security policy

Information security policy clearly defines an organization's perspective to consider information security. Information security policy is systematically organized and managed as follows:

Name	Role
Information security policy	Concepts concerning information security
Fundamental information security policy	The organization's unified and fundamental concepts and principles
Information security measures criteria	Compliance items and criteria for the practice of fundamental policy
Information security measures implementation procedures, etc.	Specific implementation procedures, etc.

Fundamental information security policy clarifies the information assets to be protected by the organization and the reasons for their protection, and describes approaches (ideas and principles) to information security in a document.

Information security measures criteria defines the organization's unified behavior and decision-making criteria that should be observed with regard to information security, on the basis of its fundamental information security policy.

Information security measures procedures is a set of the specific implementation

procedures for information security measures. They define specific security measures (security controls) for each specific target (e.g., hardware products, software products), from the standpoint of information security management functions (prevention, detection, minimization, and restoration). In particular, it is important to define in advance any response measures for situations that would present difficulties for business continuity.

- **Contingency plan** (emergency response plan)

A contingency plan defines a set of response measures (e.g., minimization, recovery) that is primarily used in the event of urgent or emergency situations. Recovery from emergency situations resulting from disaster, in particular, may be termed **disaster recovery**.

Within companies and other organizations, a variety of security rules and regulations must be created on the basis of their information security policies. The following list shows some typical rules and regulations. In general, approval for these rules and regulations comes from top management.

[**Security rules and regulations of corporate activities**]

- Rules and regulations concerning employment agreements
- Office rules and regulations
- Security control rules and regulations
- Documentation control rules and regulations
- Information management rules and regulations
- Privacy policies (personal information protection policies)
- Rules and regulations on measures to be taken against computer virus infection
- Security education rules and regulations
- Penal rules and regulations
- Outward explanation rules and regulations
- Rules and regulations for updating rules

Among these rules and regulations, **privacy policies** (**personal information protection policies**) have been regarded as important in recent years.

Privacy policies (personal information protection policies) are policies related to activities for the protection of information (i.e., personal information) identifying individuals, such as name and date of birth. Personal information protection is implemented on the basis of these policies, as a part of a systematically integrated **compliance program** (i.e., a plan for compliance with laws and regulations by the company).

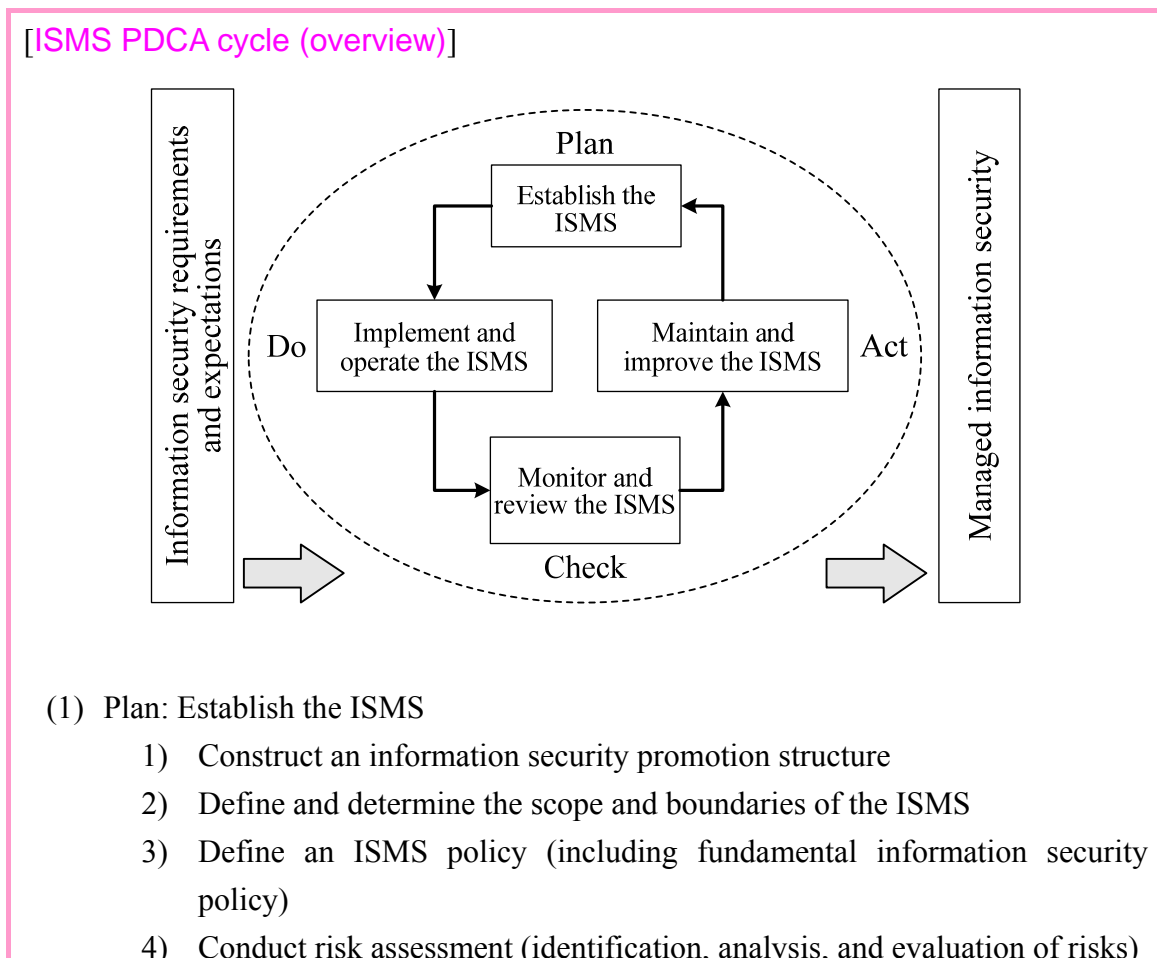
(2) ISMS (Information Security Management System)

ISMS (Information Security Management System) refers to a management system for the proper operational management of the organization's information assets overall, and for securing and maintaining the security of these. ISMS is built upon BS7799-1 and BS7799-2 of the British Standards Institution, and at present is standardized as the ISO 27000 family.

ISO/IEC standard	Overview
ISO/IEC 27000	Information security management systems - Overview and vocabulary
ISO/IEC 27001	Information security management systems – Requirements
ISO/IEC 27002	Code of practice for information security controls

ISMS provides a model for responsible persons (management team) and staff members in an organization so that they can build and operate an effective information security management system. Responsible persons must have commitment to (i.e., must have an official involvement in, and give approval to) the establishment, implementation, operation, monitoring, review, maintenance, and improvement of ISMS.

ISMS is operated according to the **PDCA cycle (Plan, Do, Check, Act)** indicated below.



- 5) Select control objectives and controls for the treatment of risks
- 6) Obtain management authorization to implement and operate the ISMS
- 7) Prepare a Statement of Applicability
- (2) Do: Implement and operate the ISMS
 - 1) Formulate and implement a risk treatment plan
 - 2) Implement controls to meet the control objectives for risk treatment
 - 3) Define how to measure the effectiveness of the controls
 - 4) Implement training and awareness programs
 - 5) Manage operation of the ISMS and resources for the ISMS
- (3) Check: Monitor and review the ISMS
 - 1) Undertake regular reviews of the effectiveness of the ISMS
 - 2) Measure the effectiveness of controls
 - 3) Conduct internal ISMS audits
- (4) Act: Maintain and improve the ISMS
 - 1) Implement the identified improvements
 - 2) Take appropriate corrective and preventive actions

In Japan, the **ISMS conformity assessment system** of the JIPDEC (Japan Institute for Promotion of Digital Economy and Community) is available for evaluation and certification of ISMS by a third party. In order to acquire ISMS certification, the application documents must be prepared and submitted to an organization (i.e., an ISMS certification body or registration body) accredited by JIPDEC. The ISMS certification body evaluates the ISMS of certification applicants by using ISO/IEC 27001 (JIS Q 27001) control objectives and controls as criteria.

[Control objectives in ISO/IEC 27001 (JIS Q 27001)]

1. Security policy
Fundamental information security policy
2. Organization of information security
Internal organization, external parties
3. Asset management
Responsibility for assets, classification of information
4. Human resource security
Prior to employment; during employment; termination or change of employment
5. Physical and environmental security
Secure areas, security of equipment
6. Communications and operations management

Operational procedures and responsibilities, third party service delivery management, system planning and acceptance, protection against malicious code and mobile code, back-up, network security management, media handling, exchange of information, electronic commerce services, monitoring

7. Access control

Business requirements of access control, user access management, user responsibilities, network access control, operating system access control, Application and information access control, mobile computing and teleworking

8. Information systems acquisition, development and maintenance

Security requirements of information systems, correct processing in applications, cryptographic controls, security of system files, security in development and support processes, technical vulnerability management

9. Information security incident management

Reporting information security events and weaknesses, management of information security incidents and improvements

10. Business continuity management

Information security aspects of business continuity management

11. Compliance

Compliance with legal requirements; compliance with security policies and standards, and technical compliance; information system audit considerations

1-3-2 Risk Management

ISO 31000:2009 defines **risk management** as “coordinated activities to direct and control an organization with regard to risk.”

Risk is defined as “the effect of uncertainties on objectives” in ISO 31000:2009. Here, an effect is a positive or negative deviation from what is expected. As an example, companies perform business activities to obtain targeted profits through a variety of capital investments, such as investment in land or stocks, or development of new products. A certain type of risk makes more profits than expected, or conversely, causes losses. (Such type of risk that occurs under the control of the managing entity is called a **speculative risk**). In contrast, there is also another type of risk such as natural disaster, man-made disaster, and theft that result only in negative effect. (The type of risk that occurs outside of the control of the managing entity is called a **pure risk**).

Risk management determines and manages organizational responses to these risks (especially speculative risks) in advance. Risk management is also implemented as part of ISMS.

[Risk management procedure]

(1) Confirmation of the organization's status

Identify the information assets that are subject to risk analysis, and classify those assets in consideration of the importance of each asset from aspects of confidentiality, integrity, and availability. On the basis of the results, determine criteria (i.e., the required information security standards) for protection of information assets.

(2) Implementation of **risk assessment**

1) Risk identification

Identify risks on the basis of threats to information assets, vulnerabilities, etc.

2) Risk analysis

Analyze the frequency of occurrence of risks, the scale of impacts (damage, losses) made when the risks are exposed, or such other factor. Risk analysis focuses on **perils** (i.e., causes of loss) and **hazards** (i.e., dangerous conditions). Perils are factors that lead to risk, but no risk is likely to occur so readily without any hazards. As an example, even in the event of an earthquake (a peril), a building will not collapse readily if it is not in deteriorated condition (a hazard). (Threats can be thought of as perils, and vulnerabilities as hazards). It is also necessary to be cautious of **moral hazards** that are conditions where the sense of danger is decreased due to compensations, such as insurance against risks, and as a result, risks are increased.

3) Risk evaluation

On the basis of the results of risk analysis, evaluate whether each risk is acceptable or allowable.

(3) Treatment for risk

Taking into account both an evaluated risk and the required level of information security, determine and implement treatment measures for that risk.

(1) Types of risks

Risk identification distinguishes risks for each information asset. Each targeted information asset has the different types of risks, such as failure, breakage, or theft for hardware and other physical assets, and errors or malfunctions for software assets, so it is necessary to check for missing assets.

In addition, risk analysis may classify risks according to the types of losses caused by the risks. Losses caused by risks include the following types. Note that risk analysis targets not

only pure risks that cause only losses but also speculative risks that can cause both profits and losses.

- **Property loss**

Loss incurred due to loss of property or decline in the value of property

- **Responsibility loss**

Loss incurred due to compensation, penalties, etc. paid for liabilities borne

- **Net operating income loss**

Loss incurred due to reduced income caused by sales opportunity loss, etc.

- **Human cost**

Loss incurred due to human resources or due to decline in human capacity

(2) Measures against risks

Measures to be considered for implementation as risk treatment include the following types.

Risk measure name	Content
Risk control	Prevents the occurrence of risk and reduces loss.
Risk avoidance	Ceases activities or the use of assets, or makes substitutes.
Risk prevention	Makes improvements to vulnerabilities, and decreases the frequency of occurrence of threats.
Risk isolation	Isolates assets and reduces the effect of risks.
Risk concentration	Concentrates the sources of risks, and performs centralized management.
Risk transfer	Transfer risks through means such as agreements with rental businesses.
Risk optimization	Makes risks acceptable through risk isolation or other means.
Risk finance	Conducts financing to cover damages borne by the exposure of risks.
Risk retention	While retaining risks, prepares for losses through operating expenses or reserve funds.
Risk transfer	Purchases insurance to share losses with other companies at the occurrence of risks.

Measures against risks are to be selected with consideration of cost-effectiveness (i.e., the balance between the values (or incurred losses)) of information assets and the costs of the measures. For that reason, since **residual risk** may be exposed after risk treatment, it is

necessary to fully investigate the status of damages from the risk and confirm that there is no problem.

Moreover, it is also necessary to determine the priority of risk treatment in preparation for the simultaneous occurrence of multiple risks.

1-4 Information Security Agencies and Evaluation Criteria

Before ending the Overview of Information Security, this subsection discusses agencies related to information security and typical information security evaluation criteria.

1-4-1 Information Security Agencies

- **CSIRT (Computer Security Incident Response Team)**

This is a general name for organizations that collect and monitor information pertaining to security incidents, and investigate causes and the scope of effects in the event of problems.

- **NISC (National Information Security Center)**

This is an information security center established in the Japanese Cabinet Secretariat to perform integrated and efficient execution of Japan's information security policies.

- **IPA Security Center**

This is an organization within IPA (Information-technology Promotion Agency, Japan) that cooperates with related institutions to perform information collection and analysis not easily performed by private parties, and generalizes the knowledge.

- **CRYPTREC (CRYPTography Research and Evaluation Committees)**

This is a project and an institution that evaluate and monitor the security characteristics of ciphers recommended for digital government, and investigate and consider appropriate implementation methods and operational methods for cryptography.

- **JPCERT/CC (JaPan Computer Emergency Response Team / Coordination Center)**

This is an incorporated association that collects and communicates information concerning security. It also plays a role as a coordinating body for the Information Security Early Warning Partnership.

1-4-2 Information Security Evaluation Criteria

- **Common Criteria** (ISO/IEC 15408)

This is a standard for evaluation criteria disclosed by certification bodies, in the **JISEC (Japan Information technology Security Evaluation and Certification scheme)** operated by IPA (Information-technology Promotion Agency, Japan). It regulates security functional requirements and security assurance requirements, and serves as criteria for **EAL (Evaluation Assurance Level)**.

- **JCMVP (Japan Cryptographic Module Validation Program)**

This is a set of evaluation criteria and an operational procedure to certify that cryptographic modules, implementing cryptography, digital signatures, and such other functions, appropriately protect sensitive information contained within.

- **PCIDSS (Payment Card Industry Data Security Standard)**

This is security criteria (i.e., certification evaluation criteria) for the protection of credit card information and transaction information. PCIDSS also defines the implementation of **penetration testing**, the tamper resistance of cards, etc.

2 Information Security Measures

Information security measures must be implemented comprehensively, so as to include not only information systems but also the environments surrounding people.

An ISMS addresses three aspects: human security, technical security, and physical security from the perspective of information security measures.

2-1 Human Security Measures

Human security is security concerning human procedures, management, and operational rules and has an objective to prevent accidents and incidents such as security breaches by concerned parties.

With respect to the organization, management and monitoring are performed by appointing a **CISO (Chief Information Security Officer)**, an information officer of each division or section, and an information system administrator, all of whom own authority and responsibility for information security. On the other hand, with respect to the organization's personnel, management and monitoring are performed by formulating a **company regulation** that includes the following items, in the form of **information security policy**.

- Screening at the time of employment, dispatch, or outsourcing contract of a staff member
- Non-disclosure (or confidentiality) agreement
- Information security education/training
- Handling/contact in the event of a security accident or an incident
- Disciplinary process for security breach
- Handling of information systems
- Account management / password management

In the PDCA cycle of human security, information security policy is established at the ISMS planning (Plan) stage, and security education and training are implemented at the execution (Do) stage. Furthermore, the security compliance status of personnel is checked through **log management** and monitoring at the inspection (Check) stage. Any deficiencies or improprieties are handled and corrected at the Act stage.

In human security measures, the greatest importance should be placed on information leakage by concerned parties. Information leakage due to inattention of concerned parties is difficult to completely prevent through agreements, rules, or information security education. As a measure against information leakage, unnecessary information disclosure should be avoided

under the “**need-to-know**” principle. For example, appropriate access permissions should be set or important documents should be kept in a locked location.

Such activities regarding information leakage measures are also important in acquiring the **Privacy Mark (P Mark)**. The Privacy Mark system confers the mark on private or other businesses that have prepared an appropriate protection system for the handling of personal information.

2-2 Technical Security Measures

Technical security is security that utilizes hardware, software, networks, and other technologies. The following are types of technical security measures and related techniques (Details of implementation technology are discussed in “2-4 Security Implementation Technology”).

- **Anti-cracking measure**

It is difficult to prevent cracking only through a human security measure (e.g., a punitive regulation) that is aimed at a cracker. For that reason, a **measure against unauthorized access** is implemented to prevent unauthorized intrusion into another person’s PC, and a **measure against information leakage** such as cryptographic processing is implemented to prevent peeping of data by others.

- **Measure against malware / measure against computer virus**

This is a technical measure for the prevention and detection of infection, prevention of the spread of damage, and restoration from damage caused by malware (i.e., computer viruses). This measure often uses security software that combines multiple security measure functions, such as antivirus measures or anti-spyware measures. Since an infection is often passed via networks by means of e-mail, an attachment file. etc., this measure is effective when used in conjunction with software that achieves network security.

- **Antivirus software (vaccine software)**

This is software that performs detection or elimination of computer viruses. The following are two typical methods of computer virus detection.

- **Pattern matching method**

This is a method that creates a database (a pattern file or a definition file) of the characteristic portions of viruses in the form of “signature code” (i.e., patterns), and performs a matching check between the database and a target program code. Since this method is unable to detect viruses that have not been registered, it is necessary to always update the pattern files to the latest versions.

- **Rule-based method**

This is a method that detects viruses by analyzing program behavior on the basis

of established rules. This method includes the **heuristic method** that statically analyzes source code obtained by means such as disassembly of binary code, and the **behavior method** that runs programs to detect (i.e., dynamically analyze) dangerous behavior that indicates a virus.

Initial measures (e.g., disconnection from networks, contact with administrators) taken in response to a virus infection are defined within information security policy and are implemented as human security measures. The person initially detecting a virus should, in principle, not perform operations such as elimination of the virus or initialization of memory. This is because doing so disallows investigation and analysis of the type of virus, the infection path, the status of damage, and other information.

- **Updating**

This is a measure that is provided by a manufacturer or a software vendor to remove vulnerability from an operating system or an application, through an update program (**patch program**) which is aimed at defects in software security. This can also be effective in preventing the newly discovered malware or infection by computer viruses. It should be noted that while it is recommended to use the latest updates, **OS updating** may have impacts on the operation of other programs. This should be confirmed in advance.

- **Measure against spam**

Spam (i.e., spam e-mail, also called junk e-mail) is an advertisement message, a chain message, or other e-mail that is sent to an unspecified number of recipients. Measures against spam include denial of receiving spam, and avoidance of being used as a stepping stone for unauthorized relaying of spam. When spam is received, the optimal measure is to delete the spam without taking any other action, such as opening or replying to the spam.

- **Blacklist / whitelist**

This is a method that uses a blacklist of e-mail addresses from which receipt is rejected, and a whitelist of e-mail addresses from which receipt is accepted.

- **SPF (Sender Policy Framework)**

This is a method for sender domain authentication that manages an IP address list of proper mail servers authorized to send e-mail from a given domain. It enables automated rejection of e-mail sent from unrelated mail servers. It is an extended specification of SMTP, and is defined as RFC 4408.

- **OP25B (Outbound Port 25 Blocking)**

This is a method that is used by an ISP in order to prevent outbound communication from its internal networks via TCP port 25 (SMTP), through routers or such other devices at the boundary with external networks. When OP25B is active, SMTP communications attempting to connect through the ISP's

network to mail servers outside the ISP are all blocked, and it is possible to prevent unauthorized relaying of spam.

- **Measure against mail header injection**

This is a measure that prevents mail header injection attacks which improperly rewrite e-mail recipient addresses found on websites, and use them as a stepping stone for unauthorized relay.

- **DKIM (Domain Keys Identified Mail)**

This is a digital signature-based technique for source domain authentication, which checks whether received e-mail is from authorized senders and is not falsified. At the time of sending an e-mail message, it writes signature information, generated from a private key, into the header of outbound mail message, and at the time of receiving the e-mail message, verifies signatures by using a public key on the DNS server of the signing domain. This allows verification of the legitimacy of the source domain and body text of an e-mail message (i.e., verification that the message was not falsified during sending).

- **Content filtering**

This is a technique that restricts the viewing of content published on websites, through filtering to check for inappropriate keywords, etc. This is sometimes used to prohibit minors from viewing content containing harmful information, and is sometimes used for security purposes to prohibit access when the safety of content cannot be confirmed.

- **URL filtering**

This is a technique that prohibits the viewing of content at specified URLs. Many products have a prohibited-viewing URL list (i.e., filter) prepared from the start.

- **Honey pot**

This is a “decoy” server or device that is set up on the Internet for the purpose of investigating and analyzing methods and pathways of unauthorized intrusion, or the behavior of viruses. A honey pot is set up to disable external attacks from the honey pot (i.e., to prevent damage from spreading).

- **Digital watermarking**

This is a technique to embed data, which is imperceptible in normal viewing and playback, into documents, still images, videos, and other content. Since the information is displayed when special detection software is used to read the data, unauthorized copying or data falsification can be detected.

- **Personal Digital Assistance (PDA) security**

PDA contains personal data, like a telephone directory, and confidential data, so security measures for such a device are also necessary. PDA is portable, so it is important to make special preparations against loss and theft.

- **Cell phone / smartphone**

A cell phone and a smartphone commonly have a feature to lock the device after a set period of non-use, and such a feature should be used. Although a password method is commonly used to unlock the devices, some models use fingerprint authentication or face authentication. When the device is lost or stolen, the features can also be used by which a key operation on the device can be remotely disabled, or the device itself is initialized. Furthermore, smartphones lent by companies are sometimes centrally managed according to a security policy by using MDM (Mobile Device Management).

- **Tablet device**

A tablet device requires the same sort of security measures as a notebook PC. A computer viruses that targets a tablet device begins to appear, which means that antivirus software must be installed. In addition, data should be encrypted or otherwise protected as a measure against loss or theft.

- **Digital forensics**

Forensics has meanings of medical jurisprudence, criminal identification, and scientific investigation, and here refers to digital criminal identification. It is a general name for the technique that collects and analyzes devices, data, and electronic records required for investigation of cause in legal disputes and computer crimes such as unauthorized access and leakage of information, to clarify legal grounds for action.

2-3 Physical Security Measures

Physical security is security that pertains to environmental management including facilities and equipment, and document management including recording media. While physical security measures enhance the S (Security) of RASIS, the adoption of **RAS technology** (e.g., technology to control the occurrence of failure) to enhance RAS (Reliability, Availability, Serviceability) can also be thought of as part of physical security.

Typical types of physical security measures and related technologies include the following.

- **Zoning** (i.e., the setting of security zones for facilities and equipment)

Zoning sets areas that are to be managed as security zones, such as locations of installed hardware, storage locations for software, and storage locations for confidential documents and recording media. Earthquake- and fire-resistant equipment is adopted for security zones, and preparations are made for disasters and other physical threats.

- **Entrance and exit control**

This is the management of security zone entrance and exit, and management of visitors

from outside. Entry is controlled through means such as ID cards, with the names of entrants, dates of entry, purpose of entry, and other information recorded.

- **Locking management**

Security zones are locked so as not to allow someone to enter and exit such zones without permission.

- **Monitoring camera**

Persons entering and exiting zones can be recorded by installed cameras.

- **Remote backup**

This is a method of storing backups in a remote location, in preparation for disasters and such others.

- **Security cable**

This is a device for connecting equipment to a desk leg or secure point on fixture, to prevent theft.

- **Disk encryption**

This is a method for encrypting information on recording media to prevent information leakage.

- **Authentication device**

This is a device for user authentication used in entrance and exit control. Depending on the usage environment, the following points of caution should be noted.

- When large amounts of electric power are required, it is advised to use a contact-type IC card to which electric power is directly supplied and is not advised to use a non-contact type that generates electric power from electromagnetic waves.
- Since an optical type of device for biometric authentication may be affected by lighting, it is advised to compare an optical type with a capacitive type prior to installation.

- **USB key**

This is a user authentication device on which unique authentication information is recorded, to enable user authentication by inserting the device into equipment. This is used in client authentication that is based on IEEE 802.1X used with LANs, or such other authentication and there is no standard for the capacity of built-in memory or such other specification.

- **Disposition**

Physical destruction of media or overwriting of specified bit sequences prevents information leakage.

2-4 Security Implementation Technology

Security implementation technology refers to the technological measures implemented for

each target of attacks (threats), such as networks and databases.

It is important to use the security technology that is optimal against each of these attacks.

2-4-1 Secure Protocols

Secure protocols are protocols to prevent eavesdropping of communication data or unauthorized connections. A variety of secure protocols are used in TCP/IP networks.

- **IPsec (Security Architecture for Internet Protocol)**

This is a secure protocol that is a standard specification for IPv6. Since IPsec performs packet cryptography and authentication in the Internet layer, it enhances the confidentiality and validity of data.

- **SSL (Secure Sockets Layer)**

This is a secure protocol that is used between the application layer and the TCP layer, and performs authentication (digital signature) and encryption of communication content (session key cryptography) between client and server. **TLS (Transport Layer Security) 1.0** was drafted by the IETF as RFC 2246 that was standardized on the basis of SSL.

Original protocol	Secure protocols combined with SSL
MIME	S/MIME (Secure MIME)
HTTP	HTTPS (HTTP over SSL/TLS)
FTP	FTPS (FTP over SSL/TLS)
SMTP	SMTP over SSL/TLS
POP3	POP3 over SSL/TLS

- **SSL accelerator**

This is hardware that reduces the cryptographic processing load on servers to increase speed.

- **SSH (Secure SHell)**

This is a protocol that performs encryption of passwords and data at the TCP layer and application layer, and enhances the security of a UNIX-like command (e.g., rsh).

- **PGP (Pretty Good Privacy)**

This is an individual-oriented secure protocol that encrypts the body text of e-mail by using a hybrid method (common key: IDEA; public key: RSA).

- **APOP (Authenticated POP)**

This is a protocol that uses the POP with added security functions and encrypts passwords during login to mail servers. However, it does not encrypt the body text of e-mail.

- **SET (Secure Electronic Transactions)**

This is a credit card settlement protocol that uses cryptography and digital signatures. It restricts information that can be looked up at the retail store and credit card company, offering high confidentiality.

The protocols used for SPF and DKIM, described in “2-2 Technical Security Measures,” are called **authentication protocols**. Authentication protocols can be called a type of secure protocol used to prevent unauthorized use of services or unauthorized connections caused by or resulting from spoofing.

- **SMTP-AUTH**

This is a protocol by which authentication of users is performed by the mail server during transmission of e-mail. SMTP does not have a user authentication mechanism, which creates a problem in the increasing unauthorized use of mail servers for sending spam e-mail. SMTP-AUTH authenticates users by specifying user IDs and passwords in advance.

- **OAuth**

This is an authentication protocol that performs handover of user privileges, with user consent, between services (i.e., applications) that have an already-established relationship of trust. Specifically, it accesses resources on Web servers on behalf of users, and enables a service to perform acquisition, addition, updating, and deletion of information that is managed by another service. With OAuth, a user can transfer access privileges to a service without handing over a password, and can set the scope of applicability and the period of validity.

- **DNSSEC (DNS SEcurity Extensions)**

This is an extended specification that improves the security of DNS. While a normal DNS server is unable to authenticate communicating parties, DNSSEC is able to confirm the creator and integrity of data by using digital signatures. The responding DNS server signs with a private key, and the recipient validates with a public key.

- **EAP (PPP Extensible Authentication Protocol)**

This is a protocol that is used to authenticate remote access users. Authentication methods using EAP, which adds authentication functions to PPP, include the following.

- **EAP-TLS (EAP Transport Layer Security)**

This uses digital certificates on both server and client. It is defined by the IETF as RFC 2716.

- **PEAP (Protected EAP)**

This issues digital certificates on the server side, and authenticates on the client side by user ID and password. It was developed by three companies: Microsoft, Cisco Systems, and RSA Security.

In contrast to authentication protocols aimed at user authentication, the **anonymous FTP** protocol enables provision of service to unspecified users. This protocol enables anyone to use an FTP service by entering “anonymous” as a user ID. While it is convenient as a means of reducing workload such as registering and administering user IDs, it is easily targeted by crackers and requires that sufficient security measures be taken.

2-4-2 Network Security

Network security is a general name for security against attacks using networks. Network access control to prevent intrusion (i.e., unauthorized access) into internal networks is important in network security.

(1) Firewall

Firewalls are mechanisms that are installed at the boundary between internal networks (e.g., LANs) and other networks in order to protect internal networks from unauthorized access.

Two typical firewall techniques are as follows:

- **Packet filtering**

This is a technique that permits or denies the passing of packets according to configuration rules of delivery information, such as the source IP address/port number and destination IP address/port number in the packets. Control is primarily based on information included in headers of IP, TCP, UDP, ICMP, and such other protocol, to restrict unnecessary communication from outside.

- **Application gateway**

This is a gateway that is deployed at the boundary of networks, and controls (i.e., permits or denies) communications between internal and external networks by means of a program (i.e., proxy program) that relays an application. Although a separate relay program must be prepared for each type of application, it can control application commands and data at a detailed level, and thereby can help achieve a high level of security.

A network segment isolated by a firewall from other internal LAN segments is called a **DMZ (DeMilitarized Zone)**. If a server to be made publicly available is installed in a DMZ, unauthorized access from the outside can be blocked by a firewall. Moreover, even if the server is attacked, damage to internal segments can be prevented. However, if sufficient measures are not taken on mail servers and other servers, the servers may be used as stepping stones for spam e-mail.

A **reverse proxy** also performs the same type of role as a firewall. A reverse proxy receives requests from a client in place of a specified server and relays the request to the specified server. Since this flow is reversed from that of a regular proxy server, which acts as a proxy to relay access from the inside to the outside, the server is called a reverse proxy. By embedding the functions to scan the content of packets and URLs during relay, access to a specified server can be blocked. (This function corresponds to an application gateway). However, a reverse proxy is often used for the purpose of reducing the load on servers by acting as a proxy to respond to requests coming from large numbers of clients, or for the purpose of speeding response by storing content in cache.

(2) IDS (Intrusion Detection System)

IDS (Intrusion Detection System) refers to a system that detects intrusion into network-connected equipment and such others, and performs collection and analysis of logs. It detects patterns corresponding to registered intrusion patterns, or those differing from access patterns in normal operation.

- **Host-based IDS**

This is a host-based IDS that monitors the access status of specified servers or clients, and detects intrusion.

- **Network-based IDS**

This is a network-based IDS that monitors packets passing through a network, and detects intrusion.

An **IPS (Intrusion Protection System)** sometimes have an extended function of IDS that can prevent any unauthorized intrusion by blocking the connection of an unauthorized intrusion when detected

(3) Quarantine network

A **quarantine network** is a mechanism to confirm the security of PCs connected to an internal network. The PCs is temporarily connected to an independent network (the quarantine network) for check, and if they present a problem with the network, measures are taken. Quarantine networks are generally used together with authentication servers, to confirm security through the flow “authentication → quarantine → connection.” The mechanism is also used to ensure security under **BYOD (Bring Your Own Device)** whereby employees are allowed to use personally-owned information devices for work.

- **Authentication server**

This is a server that is dedicated to performing authentication processing. There are various types such as **RADIUS (Remote Authentication Dial-In User Service) servers** that unify the management of system-wide authentication. The use of the authentication server enables **single sign-on** that allows users to access multiple systems (i.e., services) by logging on just once (i.e., one-time authentication).

(4) Call back

Call back is a user verification technique used for connections to internal networks via public lines such as regular telephone lines. Under this method of user verification, during dial-up access to a RAS (Remote Access Service) server via a public line, the RAS server cuts the outgoing call, identifies the user from a caller telephone number (or caller ID) registered with the RAS server, and calls back. However, as only the telephone or other communication device of the caller can be identified, unauthorized use of such a device by a third party cannot be detected. For that reason, the method is often used in combination with another user verification method, such as verification by user ID and password.

(5) Wireless LAN security

Since a wireless LAN allows easy connection to the LAN as long as the user is within an area where electromagnetic waves used for the wireless LAN reach, security measures to prevent unauthorized connection are important.

- **ANY connection refusal function**

This is a function that refuses connection requests when the ESSID set for the access point and the ESSID of the device do not match (i.e., when the ESSID is set to “ANY” or is left blank).

- **MAC address filtering function**

This is a function by which the MAC addresses of devices that are allowed to connect are registered with an access point, and connections from devices with unregistered MAC addresses are refused.

- **Stealth function**

This is a function that conceals ESSIDs by stopping the beacon signals emitted regularly from access points.

- **WEP (Wired Equivalent Privacy) / WPA (Wi-Fi Protected Access)**

This is an encryption method for wireless LANs. WEP is a method that performs encryption using a WEP password and SSID. WPA is an improved WEP that uses the

encryption protocol TKIP which generates and renews keys at regular intervals. WPA2 is also available, which uses the new encryption protocol CCMP (sometimes labeled AES).

(6) Other forms of network security

- **NAT / NAPT (IP masquerade)**

This is a function that converts private IP addresses to global IP addresses. Since private IP addresses are concealed, NAT and NAPT have some effect on security.

- **VPN (Virtual Private Network)**

This is a service that can be used by multiple users as if each had a unique dedicated line. A VPN can aid in preventing eavesdropping, etc.

- **Access log analysis**

This is a method of analyzing access logs to uncover traces of **address scanning** (an attack that repeats the ping command to find connectable IP addresses) and **port scanning** (an attack that attempts access while changing the port number to find services allowing intrusion). While it does not provide immediate results, it is relatively effective.

- **UTM (Unified Threat Management)**

This is a method that integrates multiple differing security functions into a single hardware device and performs centralized network management.

- **Penetration test**

This is a test that attempts actual intrusion to verify the efficacy of security measures.

2-4-3 Database Security

Database security is a general name for security against attacks on databases.

In general, it uses the **security protection functions** offered by DBMS. Security is further enhanced by concurrently implementing usage control of external media, detection of unauthorized access, or such other function.

Security protection functions	Protected content
Encryption	Encrypt the content that is stored in a database.
Setting of access privilege	Set up access privileges to a database.
Setting of password	Authenticate users with their user IDs and passwords.
Recording to log files	Record the status of use to detect unauthorized use.

2-4-4 Application Security

Application security is a general name for security against attacks that exploit vulnerabilities (i.e., security holes) in application software. Application security includes **secure programming** that implements security functions in the process of programming. As an example of such functions, there is a function that places restrictions on data written out to the buffer area of a program, in order to increase protection against “buffer overflow attacks” that improperly write data in excess of its buffer size.

- **WAF (Web Application Firewall)**

This is a device or software that blocks attacks against vulnerabilities created by Web application security holes, etc.

- **Security measures for Web systems**

Measures against attacks using Web systems include the following.

- Measures against SQL injection: Character string conversion through **escape processing**
- Measures against XSS: **Sanitizing** to make dangerous scripts harmless
- Measures against CSRF: **Re-authentication** that requests a repeat of password entry before execution

2-4-5 Secure OS

Secure OS refers to an OS developed under the concept of least privileges and the access mechanism called mandatory access control, in order to implement sufficient security functions.

- **MAC (Mandatory Access Control)**

This is a mechanism where a system administrator set an access right and the access right is enforced.

- **Least privilege**

This is a mechanism where a finely-set privilege is given as needed.

Chapter 6 Exercises

Q1

According to ISO/IEC 27001 (JIS Q 27001), which of the following is the definition of availability in information security?

- a) Ensuring that information and a processing method are accurate and complete
- b) Ensuring that a user of an information system is the correctly authorized user
- c) Ensuring that information is not disclosed to third parties
- d) Ensuring that a user can access information assets at the required time

Q2

Which of the following is the action that corresponds to social engineering?

- a) Performing an attack that exploits the security hole in an OS
- b) Externally controlling a virus-infected computer
- c) Intruding into a computer room by using a PIN that is analyzed by a program
- d) Impersonating an authorized person to request the password via telephone

Q3

When there exists a vulnerability that enables scripts to be embedded in a Web application, which of the following is an attack that exploits this vulnerability to execute the unauthorized script on a user's browser of the Web application?

- a) DoS attack
- b) SQL injection
- c) Cross Site Scripting
- d) Phishing

Q4

Which of the following is an effect of e-mail encryption?

- a) The loss of an encryption key can be prevented.
- b) The leakage of e-mail content can be prevented.
- c) The sending log of the mail server can be protected from falsification.
- d) The attack that obstructs mail service can be prevented.

Q5

When a retail store receives an order (i.e., a message) from a customer via a network, the retail store uses public key cryptography to keep the content of the order from being seen by third parties. Which of the following is the key that the customer of this retail store uses for encryption?

- a) Public key of the customer
- b) Private key of the customer
- c) Public key of the retail store
- d) Private key of the retail store

Q6

Which of the following is an appropriate description of cryptography?

- a) Common key cryptography is safe for communication with multiple parties, even when the same encryption key is used.
- b) Public key cryptography requires that the encryption key be privately distributed to the communicating party.
- c) Public key cryptography offers simpler and faster decryption than common key cryptography does.
- d) A method is made practical by which at the start of communication a common key is encrypted by public key cryptography and sent to the other party, and encryption of data is performed by common key cryptography.

Q7

When an internal user of a company forgets the password, which of the following is an appropriate action a security administrator should take after the identity of the user is verified?

- a) Decrypting the encrypted password that is managed by the security administrator, and informing the user by e-mail
- b) Decrypting the encrypted password that is managed by the security administrator, and informing the user by telephone
- c) Converting the password that is managed on the security administrator's own PC into a hash value, and informing the user by using a classified internal document
- d) Resetting the password, and having the user set a new password

Q8

Which of the following is biometric authentication that uses information which can be obtained from the human eye?

- a) Iris authentication
- b) Fingerprint authentication
- c) Voice authentication
- d) Palm authentication

Q9

Which of the following is a purpose of use of a message digest in message authentication?

- a) To confirm that the message is not falsified
- b) To confirm the encryption method of the message
- c) To confirm an overview of the message
- d) To ensure the confidentiality of the message

Q10

Which of the following is an objective of a software developer in attaching a digital signature to software when software is released on the Internet?

- a) To assure that the software developer bears responsibility for maintenance
- b) To restrict use of the software to specified users
- c) To show that the software copyright lies with the developer
- d) To assure that the content of the software is not changed illegally

Q11

Which of the following is an appropriate description of information security policy?

- a) A company's security policy is for the purpose of defining the content that should be set within each security system, and therefore, its content differs according to the security-related product to be installed.
- b) A company's security policy consists of behavior and judgment criteria to be followed, and does not include stance and direction concerning the security activities.
- c) A company's top management should externally disclose information system vulnerabilities that are factors behind the development of a security policy.
- d) In order to achieve the targeted security level, it is necessary to clearly indicate the organization's thinking concerning behavior and judgment to be followed.

Q12

Which of the following is performed in the Plan phase of a PDCA model that is applied to any ISMS process?

- a) Management of operational status
- b) Implementation of improvement measures
- c) Review of implementation status
- d) Risk assessment of information assets

Q13

Which of the following is an appropriate description of risk assessment?

- a) Since it requires too much time and expense to address all conceivable risks, an organization should forecast the loss values and frequency of occurrence, and rank risks in order of size.
- b) Until all measures to risks that are evaluated through risk analyses are completed, an organization should avoid implementing repeated risk analyses.
- c) Since risk analysis is for the purpose of preventing future losses, an organization should avoid referencing data that is collected from a similar past project.
- d) Since the purpose of risk analysis is to determine the value of losses resulting from the materialization of risk, an organization should consider measures on *prioritized* risks with the highest loss value.

Q14

Which of the following is an appropriate description of JPCERT/CC?

- a) It is a project to investigate appropriate implementation methods and operational methods for cryptography.
- b) It is a coordinating body for the Information Security Early Warning Partnership.
- c) It is a security center under the jurisdiction of the Information-technology Promotion Agency, Japan.
- d) It is an information security center that is established in the Cabinet Secretariat of Japan.

Q15

Which of the following is an appropriate description concerning antivirus software?

- a) A signature file for antivirus software is a database that contains the first 16 bytes or 32 bytes of the code of each virus.
- b) Virus detection using the signature files of antivirus software is an effective method for detecting known viruses and identifying virus names.
- c) If the size of a file that is infected by a virus is the same as the size before the infection, the file can be restored to its pre-infection state by removing the virus.
- d) In the method of detecting a virus by identifying unauthorized behavior, the name of the virus can be identified from the behavior characteristics.

Q16

Which of the following is an appropriate explanation of OP25B?

- a) Setting up a server on the Internet which appears to have vulnerabilities, and collecting information about a method and pathway of unauthorized intrusion
- b) Authenticating the source domain of the e-mail by recording signature information in e-mail headers
- c) Restricting access to websites from the internal network by filtering content
- d) Performing port number-based filtering of e-mail which is sent from the internal network to an external mail server

Q17

Which of the following is a secure protocol that combines an authentication function between a client and a server with an encryption function for communication data?

- a) APOP
- b) EAP
- c) OAuth
- d) SSL

Q18

Which of the following is an appropriate description of the role of a reverse proxy?

- a) It sends an access request, in place of a client, to a server.
- b) It receives an access request, in place of a server, from a client.
- c) It ensures the security of a PC that connects to a network.
- d) It detects intrusion into a network.

Q19

Which of the following is an appropriate description of WPA?

- a) It is software that blocks an attack against vulnerabilities in the Web application.
- b) It is a method for one-to-one conversion of an internal address to an external address.
- c) It is a method that integrates and centrally manages multiple different security functions.
- d) It is an encryption method for a wireless LAN

Chapter 7

Data Structure and Algorithm

1 Data Structure

For processing data in a computer, it is necessary to give instructions with a program. At that time, if the way to handle the data is predetermined according to the process, the efficiency is increased. This section explains data structure, which is the format of recording data.

1 - 1 Array

Array is a data structure that is used when a set of data using the same format is handled in a consolidated manner. Every a single data item (i.e., element) is distinguished with a number (a **subscript** or an **index**).

[Notation system of array elements]

- Index may start from 0 or 1. This textbook uses the notation that starts from 1 in consideration of the ease of understanding.
- Index can be enclosed with (), [], and so on. This textbook uses ().

Example 1: Without using an array, make the score of three subjects 0.

English	Japanese	Mathematics
---------	----------	-------------

<Process>

- 1) 0 → English
- 2) 0 → Japanese
- 3) 0 → Mathematics

Example 2: Using an array, make the score of three subjects 0.

Score	Score(1)	Score(2)	Score(3)
-------	----------	----------	----------

<Process>

- 1) 0 → Score(1)
- 2) 0 → Score(2)
- 3) 0 → Score(3)

The advantage of using an array is that when the same process is performed on multiple data items, the process can be described by just changing the value of the index. For example, in the process of example 2, by representing the index with variable I , it can be described as follows:

Repeat the following process while variable I is changed from 1 through 3.
 0 → Score(I)

In this manner, when a data structure is used, the processing sequence can easily be

represented. In this example, there may not seem a big difference because there are fewer process targets (i.e., three subjects). However, for larger volumes of data (e.g., 20 subjects), the user may understand how efficient it is.

(1) One-dimensional array

One-dimensional array is a data structure that handles a set of data in the same format by arranging it in a row. “Array” generally refers to a one-dimensional array.

Example: A one-dimensional array that records the results (i.e., scores) of exams in four subjects

	1	2	3	4
Score	Score(1)	Score(2)	Score(3)	Score(4)

(2) Multidimensional array

Multidimensional array is a data structure that handles array elements with multiple relations. For example, a two-dimensional array handles elements with two relations, namely, horizontal and vertical. Therefore, it requires multiple indexes for specifying elements. (The number of indexes is the same as the number of dimensions.)

Example: A two-dimensional array that records the results (i.e., scores) of three persons who took exams in four subjects

Score	1	2	3	4	
1	Score(1,1)	Score(1,2)	Score(1,3)	Score(1,4)	: 1st person
2	Score(2,1)	Score(2,2)	Score(2,3)	Score(2,4)	: 2nd person
3	Score(3,1)	Score(3,2)	Score(3,3)	Score(3,4)	: 3rd person

(3) Structured array

Structured array is a data structure where array elements are a set of data of different formats. It is also called **record array** because it has the same concept as records that are stored in a file.

Example: A structured array that records examinee’s number, name, and total score of three examinees

	Exam_number	Name	Total_score	
1	Exam_number(1)	Name (1)	Total_score (1)	: 1st examinee
2	Exam_number(2)	Name (2)	Total_score (2)	: 2nd examinee
3	Exam_number(3)	Name (3)	Total_score (3)	: 3rd examinee

A general array is stored with the fixed number of elements in contiguous areas of memory (main memory). This is called **static array**. However, some programming languages support **dynamic array** where the number of array elements can be varied during the process.

In addition, an array is an index based data structure that can be accessed directly, and therefore, it is also used as a **hash table** where storage location (i.e., index) is determined from the data to be stored. As the methods of determining storage location (i.e., index), **division method**, **superposition method**, and **radix conversion method** can be applied, which are used when the record address is calculated from the record key of a direct organization file. (A dedicated **hash function** can also be used.) Here, it is also necessary to consider the way of handling synonyms.

1 - 2 List

List (linked list) is a data structure that decides the arrangement of data with a pointer (i.e., position (or address) where element is recorded).

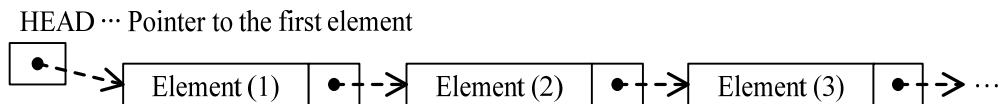


Figure 7-1 Image of list

A list can also be represented by using an array. In this case, indexes that indicates the following elements are recorded in the pointers. Figure 7-2 shows an example of a list where an array is used.

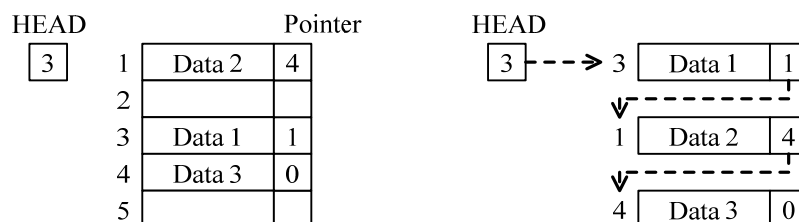


Figure 7-2 Example of a list that uses an array

Since an array is a data structure where the recording order of data has significance, insertion and deletion of data become manipulation of the entire array and are not efficient. By contrast,

in the case of a list where the recording order of data has no significance, insertion and deletion of data can be easily performed by just replacing the pointer.

For example, for inserting data *X* that is newly added in element (5) between data 1 and data 2 in the list shown in Figure 7-2, the pointer of data 1 (i.e., data that is located just before the data to be inserted) and data *X* (i.e., data to be inserted) only need to be replaced.

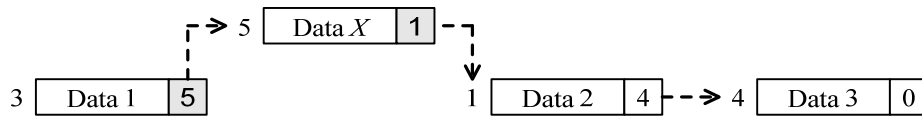


Figure 7-3 Example of inserting data in the list

On the other hand, for deleting data 2 from the list shown in Figure 7-2, the pointer of data 1 (i.e., data that is located just before the data to be deleted) can simply be replaced as shown in Figure 7-4. However, while data 2 is logically deleted from the list, data itself remains behind. Therefore, it is necessary to physically erase the data.



Figure 7-4 Example of deleting data from the list

(1) Singly-linked list

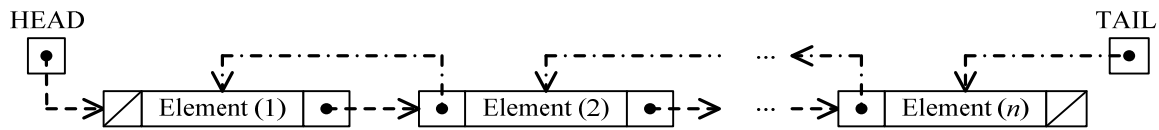
Singly-linked list is a list that can trace data in only one direction. It is also referred to as a **linear list**, a single linked list, or a unidirectional list. Since each element just has a pointer that indicates the recording position of the next element, the amount of data is less. However, its applications are limited because data cannot be traced in the reverse direction. In the pointer of last element of the list, value (e.g., NULL, 0) that indicates end is recorded.



(2) Doubly-linked list

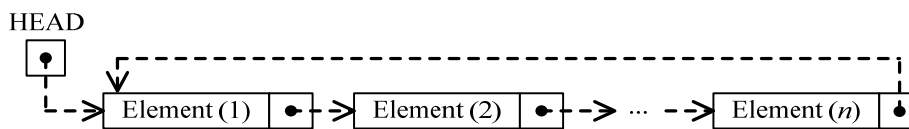
Doubly-linked list is a list where data can be tracked in both directions because it has a forward pointer that indicates the recording position of the next element and a backward pointer that indicates the recording position of the previous element. It is also called a two-way list or a bidirectional list. Since elements can be tracked in both directions, it offers wider usage of data. However, the amount of data becomes large because more pointers are

required, and processing (i.e., replacement of pointers) is also complex.



(3) Circular list

Circular list is a list where all elements are linked in the form of a ring. When a circular list is implemented with a singly-linked list, the pointer to the last element points at the first element.



A general list can be dynamically recorded in the storage area as required. Therefore, this data structure can be easily used even when the number of data items to be handled cannot be predicted. However, elements can only be traced in a sequential access, and therefore it is not suitable when some specific elements are to be processed most of the time. Even in singly-linked lists and circular lists, when a process is mostly performed on the last element of the list (for example, adding data in the last of the list), the tail pointer is used just like in a doubly-linked list. However, it is not possible to trace before the last element, and therefore, it does not result in any significant improvement of efficiency. (Efficiency hardly changes when the last element is deleted from the list.)

1 - 3 Stack and Queue

Stack and **queue** are data structures that model the methods of using data. They are also called **problem resolution data structures**, and they are implemented by using an array.

1-3-1 Stack

Stack is an **LIFO (Last-In First-Out)** data structure where data that is recorded last is extracted first. You can simply imagine a data structure where data is stacked in the box, and data is extracted from the box. Since data is stacked and extracted from the same side, data that is stored most recently is extracted first. Storing data in the stack is called **PUSH**, and extracting data from the stack is called **POP**.

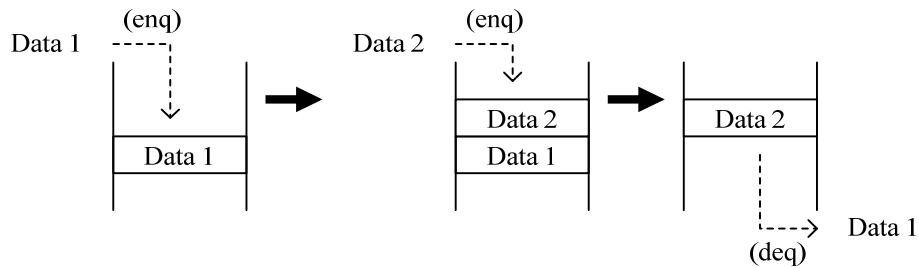
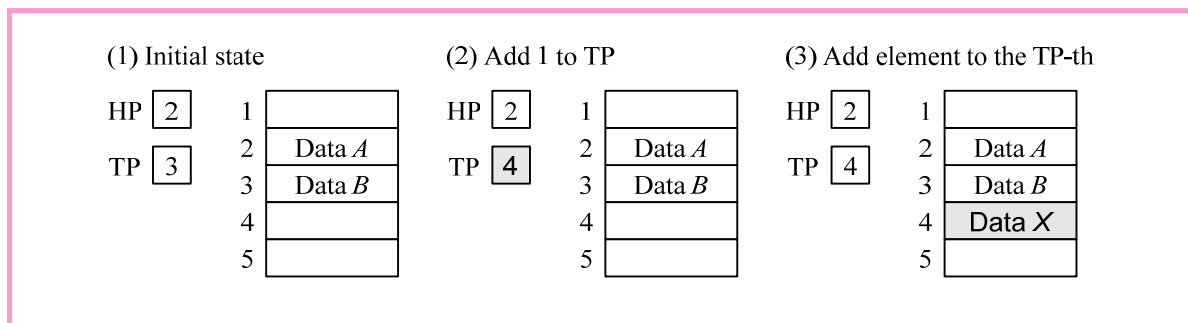


Figure 7-6 Image of queue

For implementing a queue, an HP (Head Pointer) that points at the first element and a TP (Tail Pointer) that points at the last element are used. When data is stored, it is stored at the position that is obtained by updating the value (e.g., generally +1) of pointer TP. In contrast, when data is extracted, the value of pointer HP is updated (e.g., generally +1) after data is extracted from the position of pointer HP. The process (i.e., enq) of storing data in a queue is shown as follows:



When a queue is represented with an array, both pointers (HP and TP) keep increasing by one at a time. Therefore, an infinite number of array elements is required. So, in the actual process, when the value of a pointer becomes larger than the number of array elements, the concept of returning this value to 1 is used. With this concept, array elements are used in circulation, and it becomes possible to implement a queue with an array of the finite number of elements.

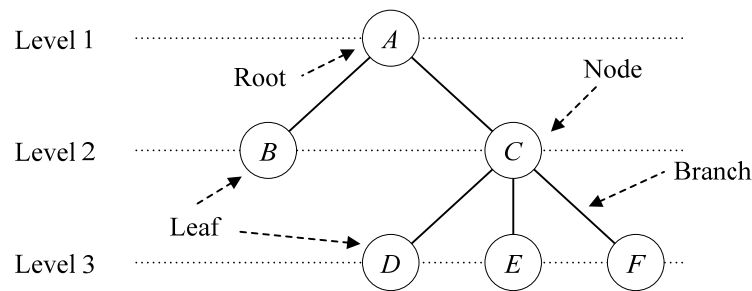


Figure 7-7 Image of circular use of array

1 - 4 Tree Structure

Tree structure is a data structure for representing a one-to-many hierarchical structure where there are multiple children under one parent.

[Constituent elements of tree structure]



- **Node** : This is an individual element (i.e., data) that is shown as a circle.
- **Root** : This is the highest-level node. A tree structure has only one root.
- **Leaf** : This is the node that does not have any lower-level node.
- **Branch** : This is a line that connects to each node (including root, leaf).
- **Level** : This is the depth of hierarchy of a tree structure.

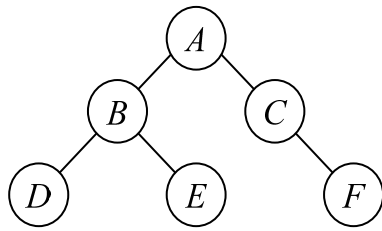
For implementing a tree structure with an array, the relation between parent node and child node is represented with a pointer (i.e., index). For example, if the tree structure shown in the figure above is represented with an array by using a pointer to parent node (parent PT), the following table is obtained. Here, Parent PT of root is 0 (not connected).

	1	2	3	4	5	6
Node value	A	B	C	D	E	F
Parent PT	0	1	1	3	3	3

On the other hand, when a relation is represented with child nodes, it is difficult to handle if the number of child nodes is not decided. Therefore, a **binary tree** is used where the number of child nodes is restricted to two nodes or less. In some cases, a tree structure that has n child nodes where the number of child nodes is more than two is distinguished as **n -ary tree** (also known as multi-way tree or multi-branch tree).

For representing a binary tree by using an array, a left pointer (left TP) that connects to the left child node and a right pointer (right PT) that connects to the right child node are used.

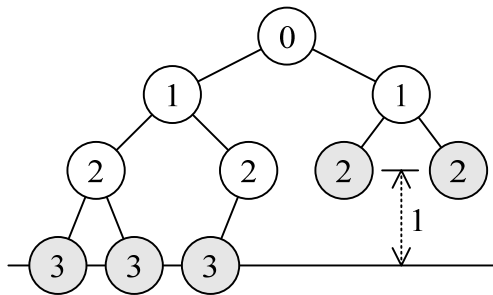
[Array representation of binary tree]



	Left PT	Node value	Right PT
1	2	A	3
2	4	B	5
3	0	C	6
4	0	D	0
5	0	E	0
6	0	F	0

In a binary tree where all final level leaves are left aligned, if level from root up to all leaves is equal or different by one level at most, it is specifically called a **complete binary tree**.

- Complete binary tree



- Tree that is not a complete binary tree

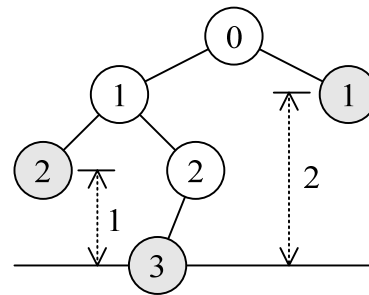


Figure 7-8 Examples of complete binary tree and tree that is not a complete binary tree

NOTE: There is another definition of a complete binary tree; in other words, a complete binary tree (sometimes called a perfect binary tree) is a full binary tree in which all leaves are at the same depth.

In addition, a method of extracting node values (i.e., data) from a tree structure is called **tree traversal**. There are the following types of tree traversals.

- **Breadth-first search**

Node values are extracted at one level at a time from left to right and from root towards leaves.

- **Depth-first search**

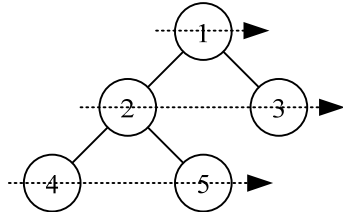
Node values are extracted in order while the periphery is tracked from the root and from left subtree to right subtree.

- **Pre-order**: This extracts node values in the sequence of “node → left subtree →

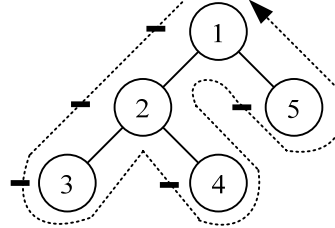
right subtree” (pre-order traversal).

- **In-order:** This extracts node values in the sequence of “left subtree → node → right subtree” (in-order traversal).
- **Post-order:** This extracts node values in the sequence of “left subtree → right subtree → node” (post-order traversal).

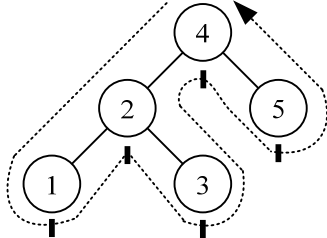
(1) Breadth-first search



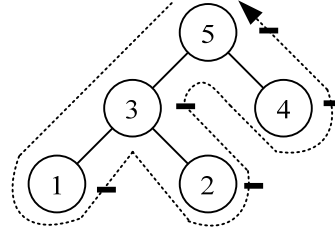
(2) Depth-first search (pre-order)



(3) Depth-first search (in-order)



(4) Depth-first search (post-order)



1-4-1 Practical Use of Binary Tree

(1) Notation of expressions with binary trees

An arithmetic expression can be represented with a binary tree in accordance with the following rules. When an arithmetic expression is represented with a binary tree, extracting node values with depth-first search (i.e., pre-order traversal) gives an arithmetic expression in the **Polish notation**, and extracting node values with depth-first search (i.e., post-order traversal) gives an arithmetic expression in the **reverse Polish notation**.

Rule 1: Let the arithmetic operator be the node, and both sides (e.g., variable, expression) of the arithmetic operator be the left child and the right child.

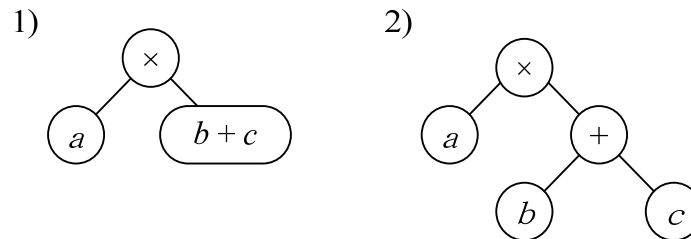
Rule 2: When there are multiple operators, keep the arithmetic operator of the arithmetic operation to be performed later at the higher level.

Rule 3: () is only used for changing the order of arithmetic operations, and it is not described in the node.

Example: Represent the expression “ $a \times (b + c)$ ” in the reverse Polish notation.

(1) Represent the expression with binary tree.

- 1) Create a binary tree where the node is arithmetic operator “ \times ” which is executed later.
- 2) Create a binary tree where “ $b+c$ ” is the right subtree; in other words, arithmetic operator “ $+$ ” is the node, and then, “ b ” and “ c ” are the left child and the right child respectively.

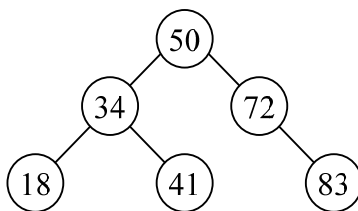


(2) Extract node values with depth-first search (i.e., post-order traversal).

Reverse Polish notation: $a\ b\ c\ +\ \times$

(2) Binary search tree

Binary search tree is a binary tree where the relation “left subtree $<$ node $<$ right subtree” holds true in all nodes. As its name suggests, binary search tree is a tree structure that is suitable for searching data. (This kind of tree structure is referred to as **search tree**.)



Left subtree	Node	Right subtree
18, 34, 41	50	72, 83
18	34	41
—	72	83

Note: In all nodes,

Left subtree $<$ Node $<$ Right subtree

Figure 7-9 Example of binary search tree

[Data searching procedure in binary search tree]

- 1) Let the root be the first search node.
- 2) Repeat the following process until search nodes are exhausted or the target value is found:
 - Compare node value of the search node with target value.
 - “Node value $>$ Target value”: Let the left child node be the next search node.
 - “Node value $<$ Target value”: Let the right child node be the next search node.

(3) Heap

Heap is a complete binary tree where a constant magnitude relation holds true between the parent node and the corresponding child nodes. Magnitude relation of only the child and parent is constant. There is no constant magnitude relation between sister nodes. (Like a binary search tree or heap, a tree structure with a constant order relation between nodes is called an **ordered tree**).

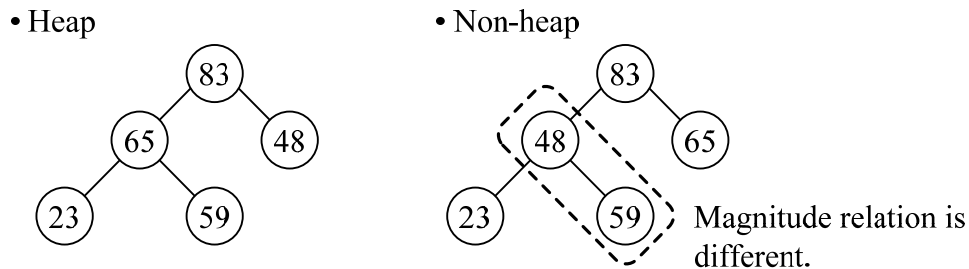


Figure 7-10 Examples of a heap and a non-heap

A heap is **reorganized** for maintaining a complete binary tree structure. In addition, in the root of a heap, the maximum value (or minimum value) of all nodes is recorded. On the basis of this property, it is possible to sort the data by extracting the data of the root in order. (This sorting method is called **heap sort**.)

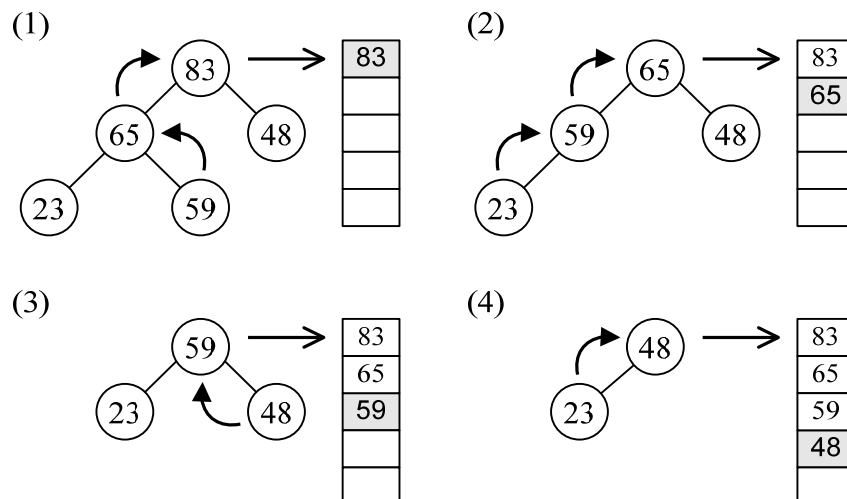


Figure 7-11 Image of heap sort

1-4-2 Balanced Tree

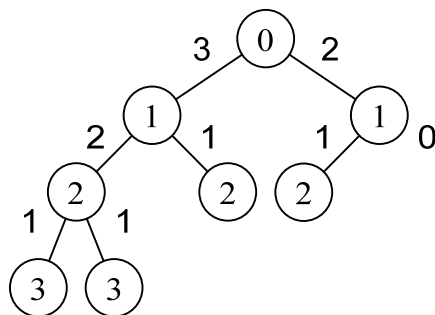
Balanced tree is a tree structure that is **reorganized** to prevent any decline in access efficiency because the level of only a particular leaf becomes deep by adding or deleting data. (A heap is also a type of balanced tree.)

The following two tree structures are the representative balanced trees.

(1) AVL tree

AVL tree is a binary tree where difference in depth between right and left leaves in each node is zero or one.

• AVL tree



• Non-AVL tree

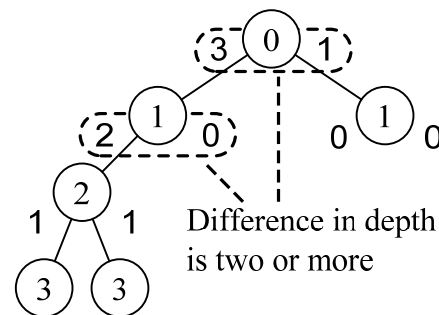


Figure 7-12 Examples of an AVL tree and a non-AVL tree

(2) B tree

B tree is a balanced tree where the concept of a binary tree is developed into n-ary tree. B tree is a data structure that has the following features.

- All leaves are at the same level.
- Each node except the root has n or more and $2n$ or fewer elements.
- In each node (including the root), the number of child node is “number of elements + 1”.
- Each element in the node is sorted.

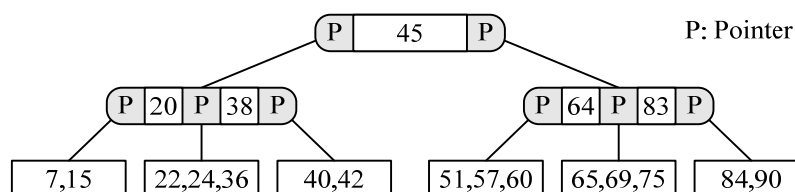


Figure 7-13 Image of B tree

2 Basic Algorithm

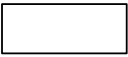
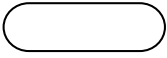
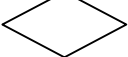
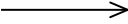
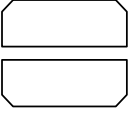
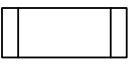
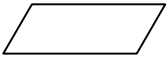
An algorithm is a sequence of solving a problem or a sequence that is used when a process is performed on a computer. An algorithm should have as much processing efficiency as possible, and it should be easy to understand. This section explains how to write flowcharts that describe algorithms, and also explains typical algorithms.

2 - 1 Flowchart

Flowchart is a commonly used notational convention of algorithms. Its main advantage is that it is visually easy to understand because the processing sequence is represented as a combination of symbols.

2-1-1 Main Symbols

The following are the main symbols that are used in flowcharts.

Symbol	Symbol name	Meaning
	Process	This represents various processes (e.g., substitution, calculation).
	Terminal	This represents start/end of flowchart. In some cases, a name is added to the terminal symbol that indicates the start.
	Decision	This branches the process according to conditions.
	Flowline	This connects each symbol and represents execution order.
	Loop limit	This represents the start/end of the iterative process (i.e., loop). It can also be described in the form of "Variable name: Initial value, Increment, Final value".
	Predefined process	This calls a predefined process. (It runs a flowchart that starts with the terminal symbol of identical names.)
	Input/Output	This represents input, output of data (it is also substituted for a process symbol in some cases).

[Structured chart]

Structured charts that are used for representing algorithms include **PAD**, **NS chart**, and so on.

2-1-2 Three Basic Structures

Three basic structures are structure units that are used in **structured theorem** for creating easy-to-understand algorithms that have excellent processing efficiency. (In the structured theorem, if the program has one entry and one exit, it can be represented as a combination of three basic structure units.) When a program is created in consideration of the processing sequence based on the structured theorem, this programming technique is called **structured programming**.

(1) Sequence

“**Sequence**” is a structure that runs the process in sequence from top to bottom. It is the most basic structure, and other structure units are also combined in “Sequence”.

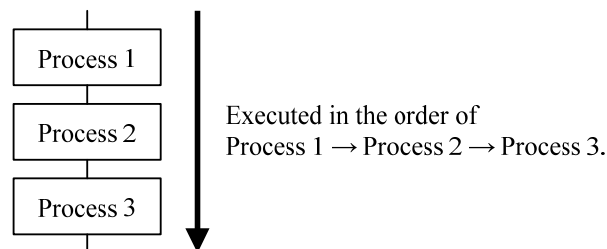


Figure 7-14 Sequence

(2) Selection (Decision)

“**Selection (Decision)**” is a structure that branches the process according to conditions. There is “**two-way selection (IF-THEN-ELSE selection)**” which branches the process into two according to conditions, and there is “**multiple selection (CASE selection)**” which branches the process into multiple (usually three or more) paths. (Generally, “Selection” mostly means “IF-THEN-ELSE selection”.)

“Two-way selection” executes one of the processes depending on whether the condition is true or false. On the other hand, “CASE selection” executes the process that corresponds to the condition that holds true among multiple conditions. However, in both cases, processes are not simultaneously executed in parallel.

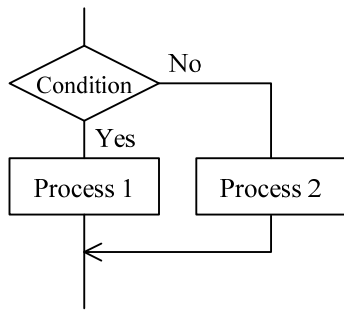


Figure 7-15 IF-THEN-ELSE selection

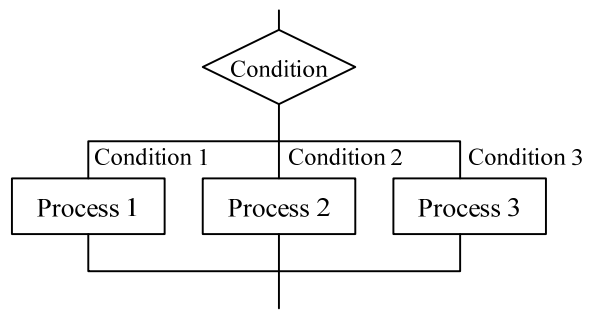


Figure 7-16 CASE selection

(3) Iteration (Loop)

“**Iteration (Loop)**” is a structure that repeats the process while the condition holds true (or until the condition holds true). In the conditions that are used in “iteration,” the **continuing condition** continues the iteration process when the condition holds true, and the **terminating condition** terminates the iteration process as soon as the condition becomes true.

In “Iteration”, there is “**pre-test iteration (DO-WHILE iteration)**” which determines the condition before the iteration process, and there is “**post-test iteration (REPEAT-UNTIL iteration)**” which determines the condition after the iteration process. “DO-WHILE iteration” may not execute the iteration process even once. However, “REPEAT-UNTIL iteration” executes the iteration process at least once.

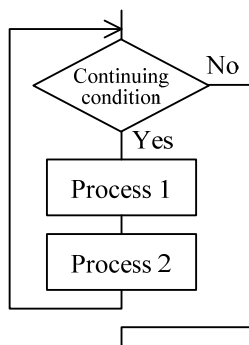


Figure 7-17 DO-WHILE iteration

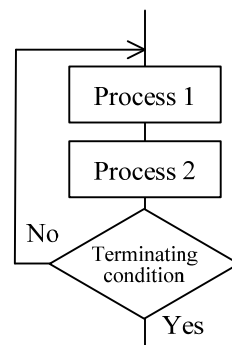


Figure 7-18 REPEAT-UNTIL iteration

In loop limit symbol, the terminating condition is basically used as the condition of “iteration”. Moreover, other than the terminating condition, changes in the value of a variable can be described in the form “Variable name: Initial value, Increment, Final value”.

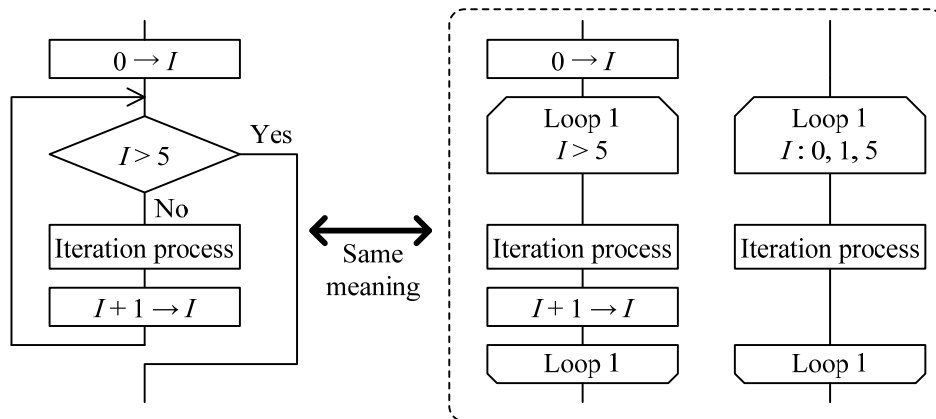


Figure 7-19 Iteration by using loop limit symbol

Three basic structures can be used in any combination. (“Iteration” can be put into the “Selection” process, and “Selection” or “Iteration” can be put into the “Iteration” process.) However, it is better to avoid the **nested structure** as much as possible where there is “Selection” in the “Selection” process.

2 - 2 Data Search Process

The process of finding the target data from the recorded data is called **search**. Representative search algorithms are **linear search** and **binary search**.

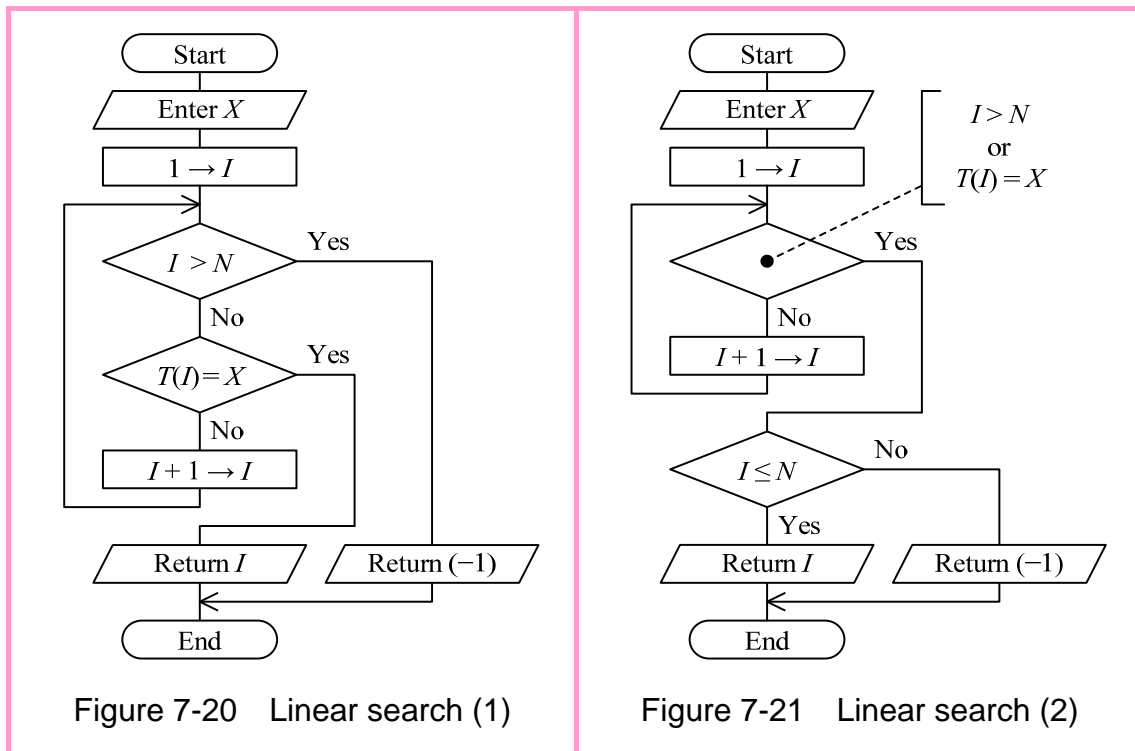
2-2-1 Linear Search

Linear search (sequential search) is the algorithm that searches for the target value in sequence from the first element.

A linear search algorithm that searches for the input target value X from array T is considered. The array T has N elements where data is recorded and returns the position (i.e., index) where the target value X is recorded. When an element that matches with the target value X does not exist, “-1” is returned instead of the index.



In linear search, comparison is made in sequence from the first element $T(1)$ to $T(N)$, in order to determine whether the element matches with the target value X or not. Therefore, in Figure 7-20, the algorithm searches for the element for which “ $T(I)=X$ ” while index I is changed in sequence from 1 to N . (Condition “ $I > N$ ” holds true when there is no element in $T(1)$ through $T(N)$ that matches with X .) However, Figure 7-20 is not a combination of basic structures: that is, exit of iteration is at two places. Therefore, this algorithm is rewritten in Figure 7-21.



In Figure 7-20 and Figure 7-21, two conditions of “ $I > N$ ” and “ $T(I) = X$ ” are compared for every one round of iteration. There is a concept called **sentinel method** that reduces this comparison of conditions (i.e., decision process) and improves process efficiency. In the sentinel method, prior to searching, in the last of all elements, the target value is stored as “sentry”. If “sentry” is stored, the target value is always found. Therefore, it is sufficient to just keep the ending condition of iteration as “ $T(I) = X$ ” where the target value is found”.

Figure 7-22 shows the algorithm of the sentry method that uses the following array T .

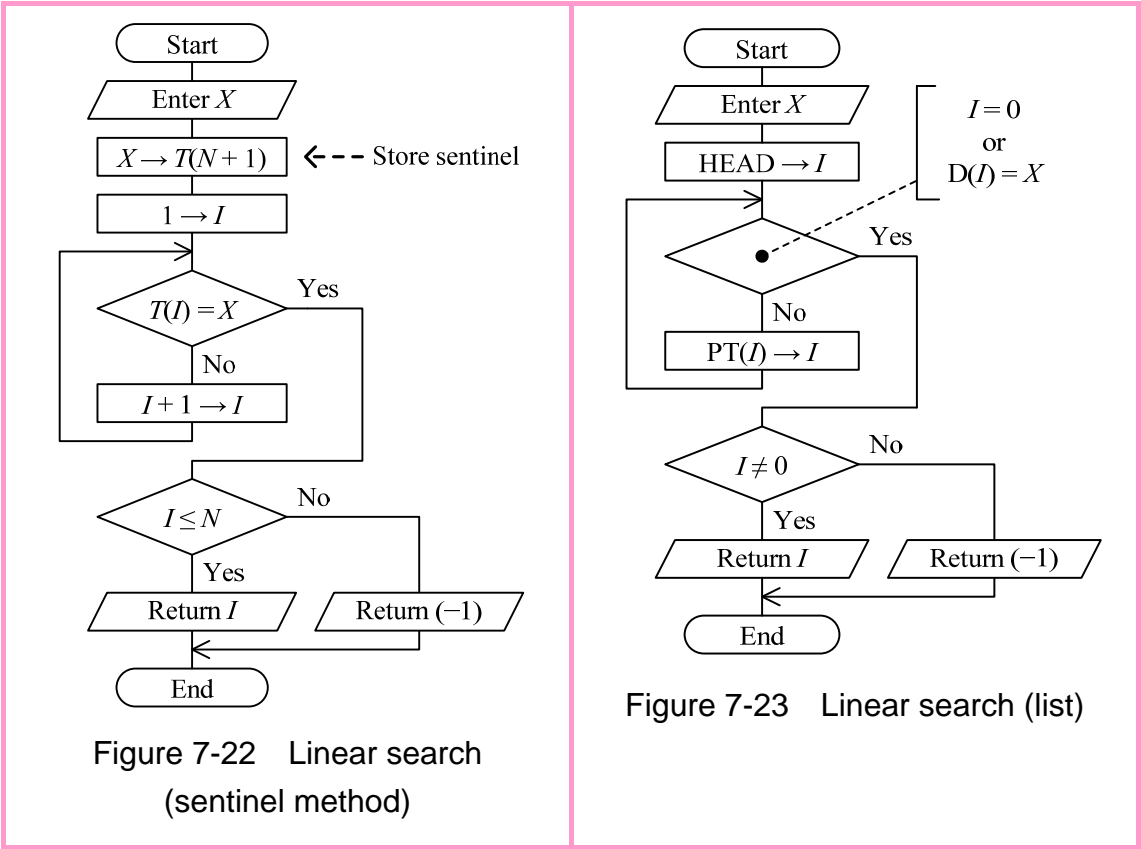
	1	2	3	...	N	$N+1$	
T	35	67	21	...	54		... $T(N+1)$: Element where sentry is stored

In addition, linear search can also be used in the list. When a list is used, elements are searched for in sequence while the list is traced by using a pointer from the first element.

In Figure 7-23, an algorithm searches for the entered target value X from the data that is recorded in the list using the following array, and returns the position (i.e., index) where the data is recorded.

HEAD		D	PT	
		1		
		2		
		3		
		⋮	⋮	
		N		

- HEAD: Pointer (i.e., index) to the head element
- D: Data to be searched
- PT: Pointer (i.e., index) to the next element
(0 when there is no next element)



2-2-2 Binary Search

Binary search is a search algorithm that can only be used when data is recorded in ascending (or descending) order. This algorithm compares the target value and the data that is located in the center of the search range, and narrows down the search range on the basis of the magnitude relation.

A binary search algorithm that searches for the entered target value X from array T that has nine elements is considered. In the array T , data is recorded in ascending order. The binary search algorithm returns the position (i.e., index) where the data is recorded. When an element that matches with the target value X does not exist, “-1” is returned instead of the index.

	1	2	3	4	5	6	7	8	9	
T	12	28	34	45	57	60	71	83	96	X <input type="text"/>

When 34 is entered as target value X , the search procedure is as follows:

- 1) Specify the entire array as the initial search range. Store a smaller index of the search range in L and a larger index in H .

	L								H	
T	12	28	34	45	57	60	71	83	96	

- 2) Compare element $T(5)$ that is located in the center ($M=5$) of a search range with the target value X .

→ Since $T(M) > X$, X is not located after $T(M)$. (Move H to one position before M .)

	L			H	M				
T	12	28	34	45	57	60	71	83	96

- 3) Compare element $T(2)$ that is located in the center ($M=2$) of the search range with the target value X .

→ Since $T(M) < X$, X is not located before $T(M)$. (Move L to one position after M .)

		M	L	H					
T	12	28	34	45	57	60	71	83	96

- 4) Compare element $T(3)$ that is located in the center ($M=3$) of the search range with the target value X .

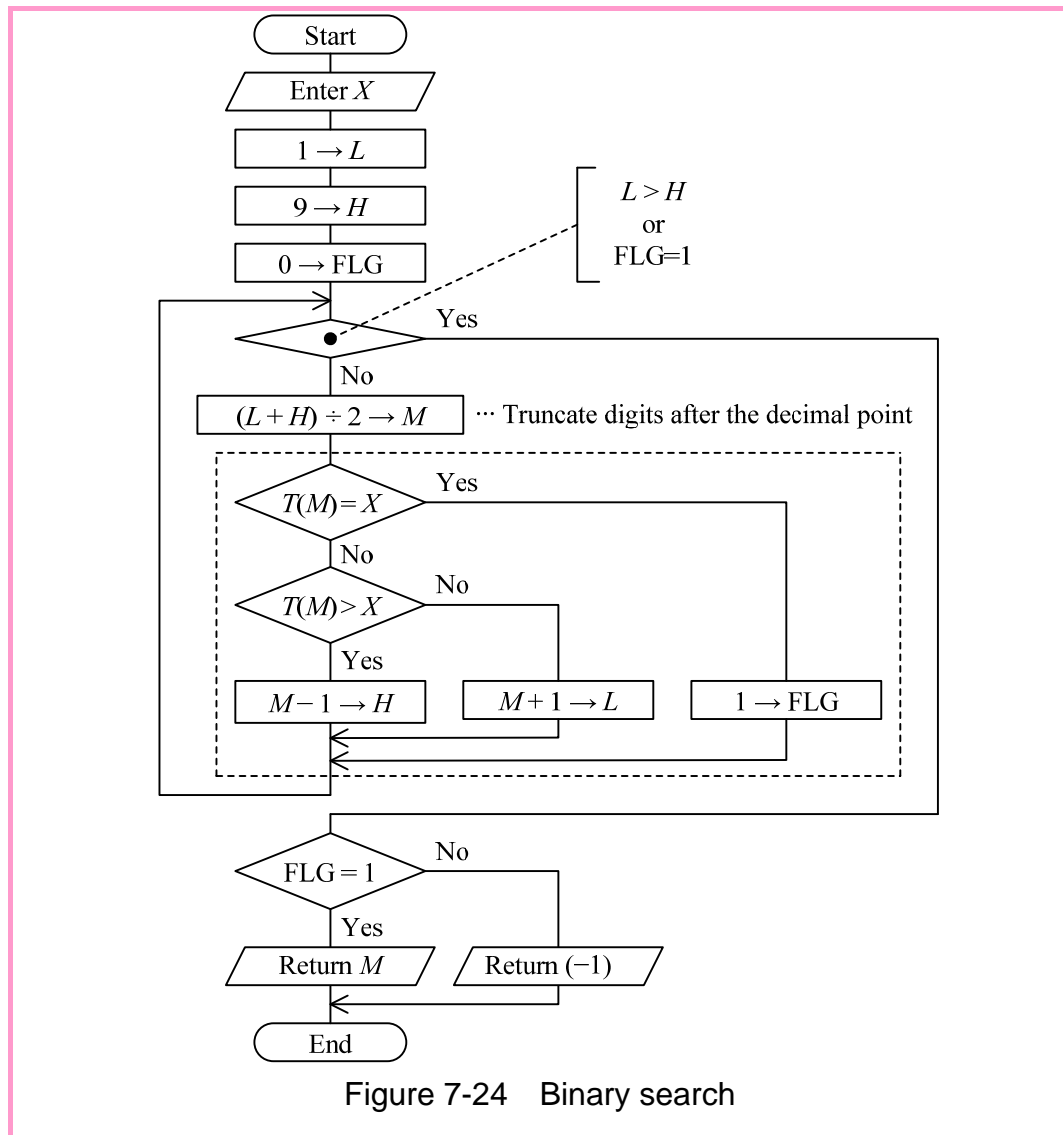
→ Since $T(M) = X$, M is returned and the search process is terminated.

			L, M	H					
T	12	28	34	45	57	60	71	83	96

Notes: 1. Index M of the element that is located in the center of the search range is determined as " $(L+H) \div 2$ ". Here, if the calculation result is a decimal fraction, truncate the digits after the decimal point.

2. If the search range is exhausted in process of repeating the process (that is, when $L > H$), it means there is no element that matches with the target value X .

A flowchart based on this concept is shown in Figure 7-24. In this flowchart, the state during the search is checked with the value of variable FLG (0: Not found, 1: Found). (A variable that represents the state is called a **flag**.) The nested structure part that is covered with a dotted line can be rewritten with CASE selection.



A search that uses a **binary search tree** can also be called a type of binary search. In a binary search tree, “Left subtree < Node < Right subtree” holds true in all nodes. Therefore, the next node to be searched for is decided according to the magnitude relation between the node value and the target value.

[Search using binary search tree (procedure when 41 is the target value)]

	Left PT	Node value	Right PT
1	2	50	3
2	4	34	5
3	0	72	6
4	0	18	0
5	0	41	0
6	0	83	0

- 1) Store index (i.e., 1) of root in I
- 2) Node value(I) > Target value
 \Rightarrow Left PT(I) $\rightarrow I$
- 3) Node value(I) < Target value
 \Rightarrow Right PT(I) $\rightarrow I$
- 4) Node value(I) = Target value
 \Rightarrow Return I and end the search

2-2-3 Hash Search Algorithm

Hash search algorithm is a search algorithm that uses **hash table** which determines the storage location (i.e., index) from the data to be stored. From the target value, the storage location of data is determined by calculation (**hash function**). Therefore, it can basically find the target value with one round of searching. The main point is the process when **synonym** has occurred.

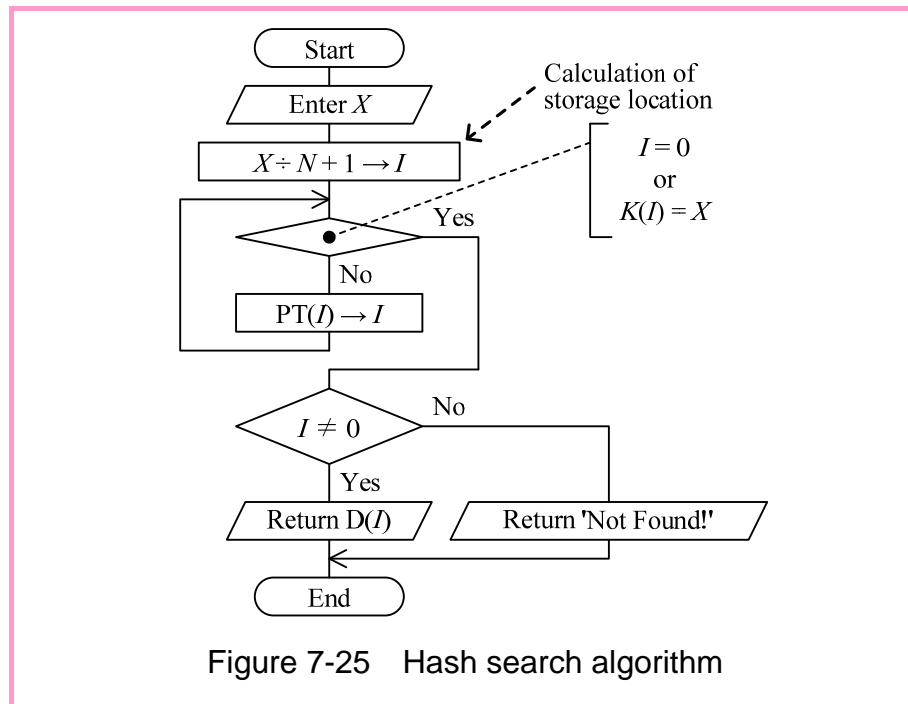
An algorithm is considered. Here, the algorithm searches for the key that matches with the entered target value X from the recorded data in the following hash table (i.e., array) and returns the data that corresponds to this key. When a key that matches with the target value X does not exist, the message “Not Found!” is returned.

	K	D	PT
1			
2			
3			
\vdots	\vdots	\vdots	\vdots
N			
\vdots	\vdots	\vdots	\vdots
M			

- K : Key that decides the storage location (i.e., index)
 $\text{Storage location (i.e., index)} = K \div N + 1$
- D : Data that corresponds to the key
- PT : Pointer (i.e., index) to synonym records
 (0 when there is no synonym record)

} Synonym area (stores synonym records)

In this hash table, **chain method** is used, which reserves the synonym area to store synonym records, and connects them with pointers. Therefore, when the key of the storage location that is determined from the target value does not match with the target value, synonym records are traced with the help of pointers. Figure 7-25 shows this flowchart.



2-2-4 Computational Complexity

Computational complexity is one of the indicators for evaluating algorithms. There is **time complexity** which shows the increase in processing time according to the amount of data to be processed, and **space complexity** which shows the increase in the use of memory space. The notation system of computational complexity includes **big O notation** which shows the upper limit of computational complexity of an algorithm with O (i.e., order). In the big O notation, if computational complexity of the part that has the maximum impact on the overall is proportional to data number N , it is represented as $O(N)$. If it is proportional to the square of N , it is represented as $O(N^2)$.

The computational complexity (e.g., time complexity) for three search algorithms is compared. In a search algorithm, a comparison process (decision process) is repeated until the target value is found. Therefore, the number of comparisons (i.e., number of searches) decides the processing time of the algorithm (or the implemented program).

The following is the number of comparisons made in each search algorithm for N data items that are covered in the search.

	Minimum number of comparisons	Maximum number of comparisons	Average number of comparisons
Linear search	1 time	N times	" $N \div 2$ " times
Binary search	1 time	" $\lceil \log_2 N \rceil + 1$ " times	$\lceil \log_2 N \rceil$ times
Hash search	1 time	" $1 + \alpha$ " times	1 time (See note below.)

- $[x]$ that is referred to as Gauss symbol shows means the greatest integer that is less

than or equal to x .

- $\log_2 N$ shows x for which $2^x = N$ (e.g., $\log_2 16 = 4$).

Note: This is the case the storage location (i.e., hash value) is approximated with uniform distribution, and in addition, the probability of occurrence of a synonym is so small that it can be ignored.

Example: When the number of data has increased from 100 to 200, by how many times does the average number of comparisons increase for a linear search and a binary search, respectively?

- Increase in the average number of comparisons of linear search
= Average number of comparisons of 200 data items
– Average number of comparisons of 100 data items
= $(200 \div 2)$ times – $(100 \div 2)$ times = 100 times – 50 times = 50 times
- Increase in the average number of comparisons of binary search
= Average number of comparisons of 200 data items
– Average number of comparisons of 100 data items
= $[\log_2 200]$ times – $[\log_2 100]$ times = 7 times – 6 times = 1 time

Therefore, the computational complexity of linear search is $O(N)$, computational the complexity of binary search is $O(\log_2 N)$, and the computation complexity of hash search is $O(1)$.

The computational complexity of big O notation is an indicator (or rough guide), and it may not always match with the actual processing time. For example, when the target value is not found, the maximum number of comparisons is made. Therefore, the average search time varies depending on the probability of search success and search failure. If an improvement is made so that the search ends as soon as it is known that the target value does not exist by using the data recorded in ascending order (or descending order) in linear search, the number of comparisons can be reduced in the event of search failure. However, the order does not change.

2 - 3 Data Sorting Process

The process of rearranging the recorded data in ascending (or descending) order of the specified items is called **sorting**. Sorting is broadly classified into two.

- **Internal sorting**

Internal sorting is performed when the data to be sorted is recorded in main memory. Selection sort, bubble sort, insertion sort, quick sort, heap sort, shaker sort, shell

sort

- **External sorting**

External sorting is performed when the data to be sorted is recorded in auxiliary storage.

Merge sort

2-3-1 Selection Sort

Selection sort is a sorting algorithm that decides one element at a time in sequence from the first element.

An algorithm of the selection sort is considered. The algorithm sorts array S with N elements where data is recorded in ascending order. For simplifying the question, the following array S with 5 elements (i.e., $N=5$) is used.

	1	2	3	4	5	
S	28	84	73	16	51	N 5 ... Number of elements

The following is the procedure that sorts array S in ascending order with the selection sort.

- 1) Select the minimum value from all elements, and let it be the first element $S(1)$.

S	16	84	73	28	51
-----	----	----	----	----	----

- 2) Select the minimum value from the remaining elements, and let it be the 2nd element $S(2)$.

S	16	28	73	84	51
-----	----	----	----	----	----

- 3) Select the minimum value from the remaining elements, and let it be the 3rd element $S(3)$.

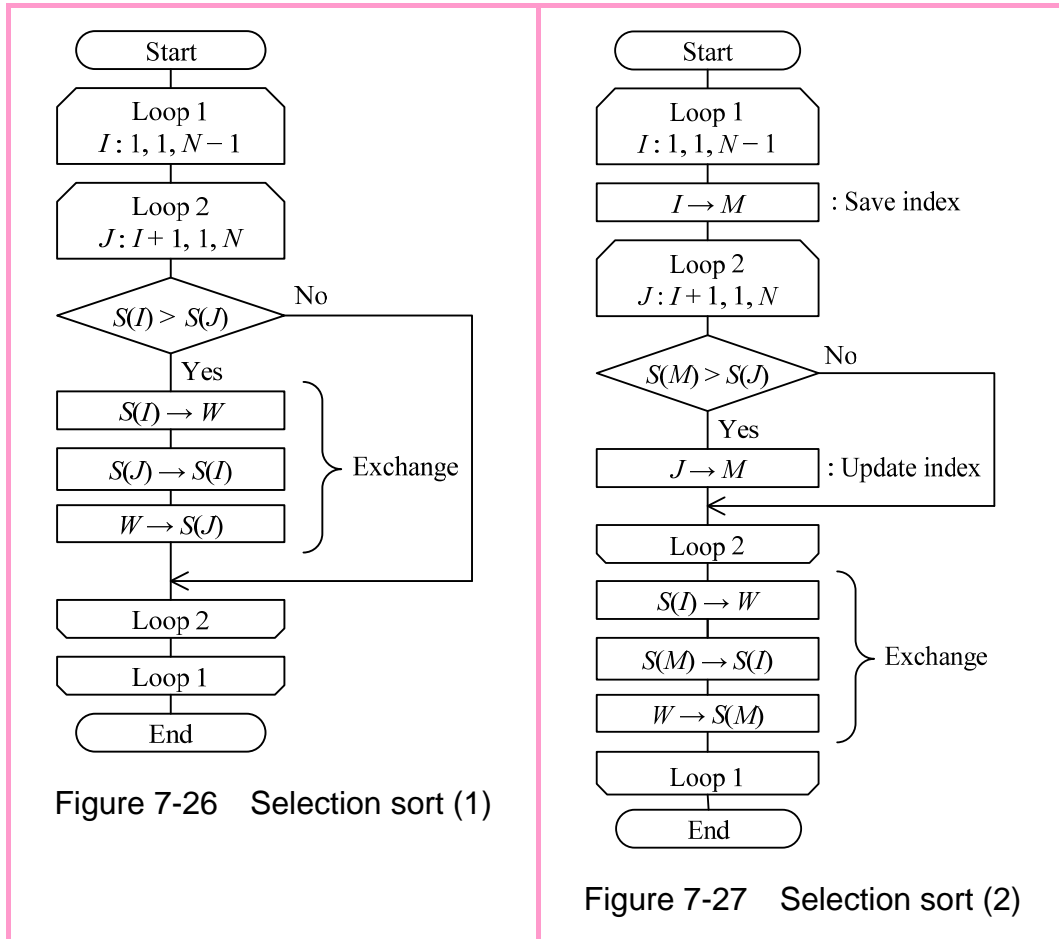
S	16	28	51	84	73
-----	----	----	----	----	----

- 4) Select the minimum value from the remaining elements, and let it be the 4th element $S(4)$.

→ The 5th element $S(5)$ is automatically decided. (Sorting ends.)

S	16	28	51	73	84
-----	----	----	----	----	----

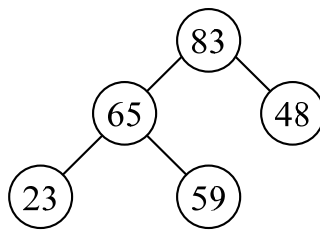
Figure 7-26 and Figure 7-27 show the flowcharts that are prepared on the basis of this concept. In Figure 7-26, elements are exchanged each time comparison is made. However, in Figure 7-27, exchange is performed only once after deciding the element to be exchanged at the completion of comparison. (The procedure that is mentioned above corresponds to Figure 7-27.) In all methods, for deciding the elements to be stored in $S(I)$, comparison is repeated with $S(I+1)$ through $S(N)$.



[Heap sort]

Heap sort is a sorting algorithm that is an improved selection sort. It represents the data to be sorted with a heap, and then sequentially extracts the data of root. (Refer to p.414 for the details of heap and heap sort.)

In the array representation of a heap, the array of the root is placed at the 1st position. The left child node of the I -th element is placed in $(2 \times I)$ -th position, while the right child node is placed at $(2 \times I + 1)$ -th position. By using this relation, a heap is reorganized while a parent node and a child node are compared and exchanged.



	1	2	3	4	5	6
H	83	65	48	23	59	
	↓ Extract root (83) and reorganize.					
H	65	59	48	23		
	↓ Extract root (65) and reorganize.					
H	59	23	48			

2-3-2 Bubble Sort

Bubble sort is a sorting algorithm that compares adjacent elements. If the magnitude relation is reverse, it exchanges the elements to correct the relation. This operation is performed in sequence from the first element. The name “bubble sort” was chosen because the action of gradually moving the maximum value (or the minimum value) in the rear direction resembles the movement of bubbles that are generated in water.

By using array S ($N=5$) that is used in selection sort, the algorithm of bubble sort is considered. Here, this algorithm sorts array S in ascending order where array S has N elements where the data is recorded.

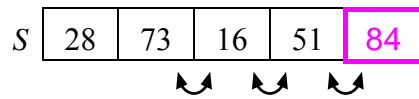
	1	2	3	4	5	
S	28	84	73	16	51	

N	5	...	Number of elements
-----	---	-----	--------------------

The procedure of sorting array S in ascending order with bubble sort is as shown below. The process of repeating comparison/exchange from the first element up to the last element is called one round of path.

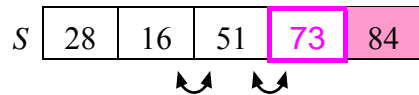
1) Make the 1st comparison/exchange in order from the first element.

→ Decide the 5th element $S(5)$.



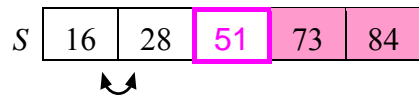
2) Make the 2nd comparison/exchange in order from the first element.

→ Decide the 4th element $S(4)$.



3) Make the 3rd comparison/exchange in order from the first element.

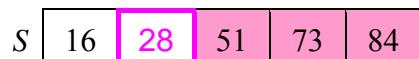
→ Decide the 3rd element $S(3)$.



4) Make the 4th comparison/exchange in order from the first element.

→ Decide the 2nd element $S(2)$.

→ The 1st element $S(1)$ is automatically decided. (Sorting ends.)



In this explanation, last elements are decided one by one and are removed from the scope of sorting. The path is repeated until positions of all elements are decided. In Figure 7-28, this algorithm is implemented by reducing N elements one by one for each round of a path.

On the other hand, if exchange does not occur even once in one round of a path, it means that all elements are arranged in the correct order and sorting ends. Figure 7-29 shows the method that uses the value of variable FLG (0: initial state, 1: exchange occurred) and terminates the process if exchange does not occur.

When efficiency is important, these two methods can also be used in combination. Please think about what kind of flowchart it turns into.

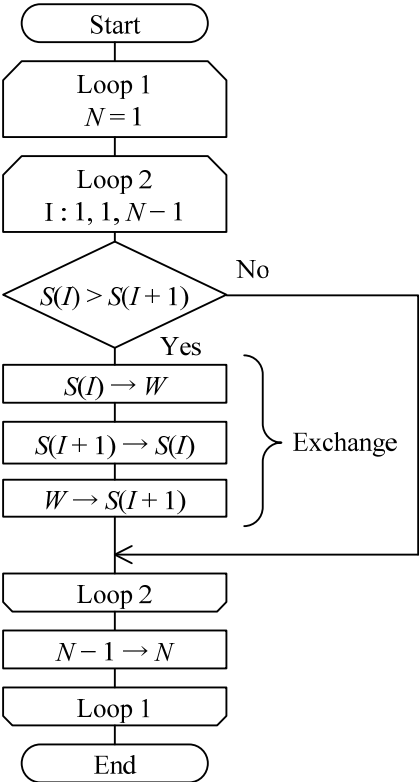


Figure 7-28 Bubble sort (1)

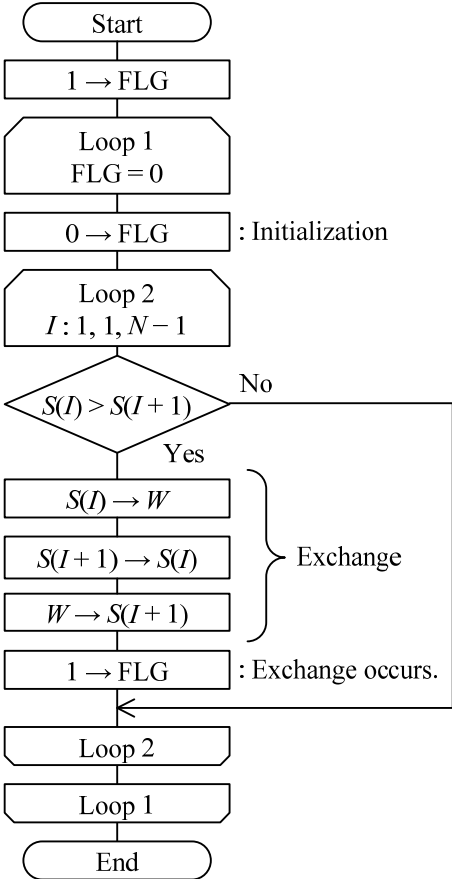
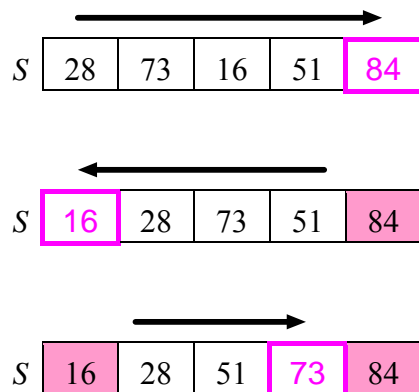


Figure 7-29 Bubble sort (2)

[Shaker sort]

Shaker sort is a sorting method that is obtained by improving bubble sort. Bubble sort compares/exchanges adjacent elements in order from the smaller element number, while shaker sort compares/exchanges adjacent elements in order from the smaller element number and then compares/exchanges in order from the larger element number. Since elements move alternately from a smaller number to a larger number and a larger number to a smaller number, it is called shaker sort. However, efficiency does not improve that much.



2-3-3 Insertion Sort

Insertion sort is a sorting algorithm that divides the data into the sorted part and elements before sorting, and then inserts data so that the order of the sorted part is not disturbed. By using array S ($N=5$) that is used in selection sort, the algorithm of insertion sort is considered. This algorithm sorts array S in ascending order where array S has N elements where data is recorded.



The procedure of sorting array S in ascending order with insertion sort is as follows:

- 1) Assume that the 1st element has already been sorted.

S

28

84	73	16	51
----	----	----	----

- 2) In the sorted part (1-1), insert “84” from the elements before sorting.

S

28	84
----	----

73	16	51
----	----	----

- 3) In the sorted part (1-2), insert “73” from the elements before sorting.

S

28	73	84
----	----	----

16	51
----	----

- 4) In the sorted part (1-3), insert “16” from the elements before sorting.

S

16	28	73	84
----	----	----	----

51

- 5) In the sorted part (1-4), insert “51” from the elements before sorting.

→ There is no element before sorting. (Sorting ends.)

S

16	28	51	73	84
----	----	----	----	----

In insertion sort, for inserting data in the sorted part in ascending order, comparison is made in order from the last element of the sorted part, and exchange is repeated until a value smaller than the inserted data is found or no comparison element is left (or exchange is performed up to the first element). In step 5) of the above example, the process of inserting element “51” is shown as follows:

- * In the sorted part (1-4), insert “51” from the elements before sorting.

S

16	28	73	84	51
----	----	----	----	----

- 1) Since “84 > 51”, exchange two elements.

S

16	28	73	51	84
----	----	----	----	----

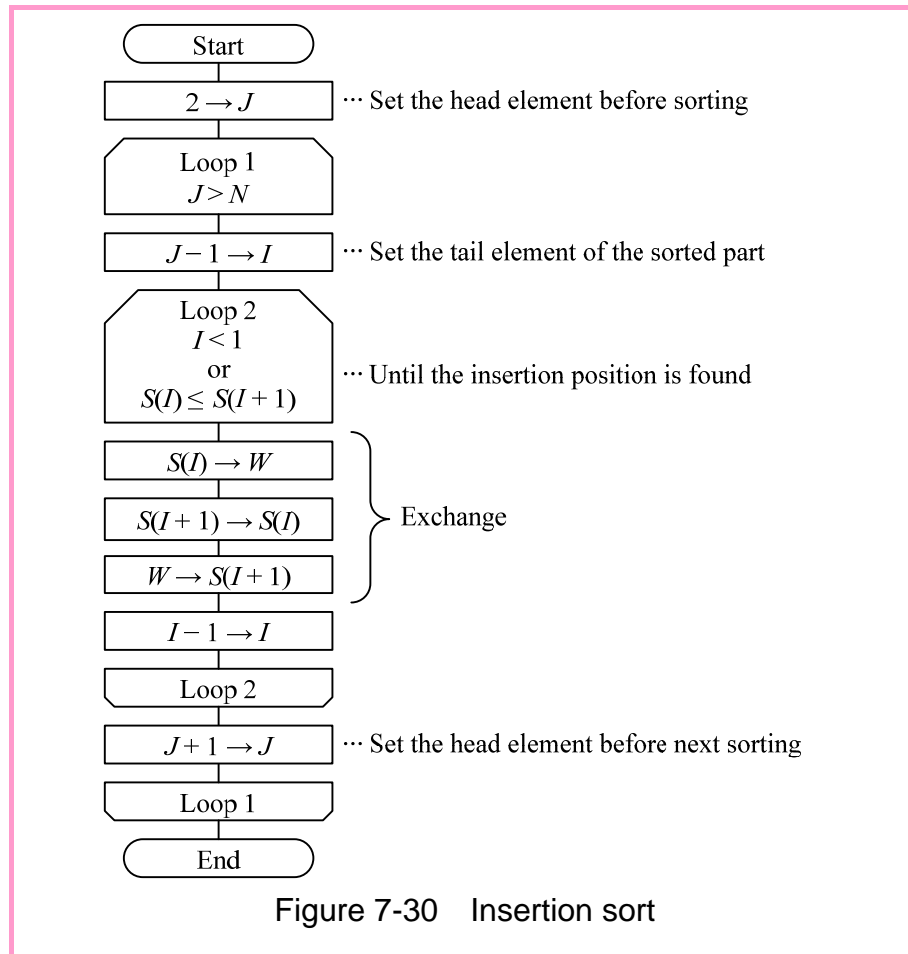
- 2) Since “73 > 51”, exchange two elements.

S

16	28	51	73	84
----	----	----	----	----

- 3) Since “28 < 51”, the insertion position is decided.

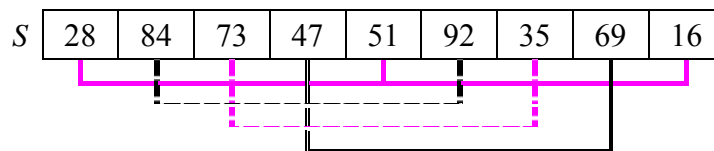
Figure 7-30 shows the flowchart that is prepared on the basis of this concept. In Figure 7-30, the first element (i.e., data to be inserted) before sorting is managed with variable J .



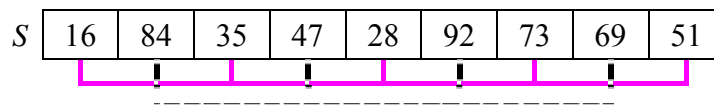
[Shell sort]

Shell sort is a sorting technique that is an improved insertion sort. Data to be sorted is extracted at regular intervals (i.e., gaps), and the data is sorted by using insertion sort. The gap size is gradually reduced, and the process is repeated until it becomes 1. It is a very efficient sorting technique where no large movement of data is required.

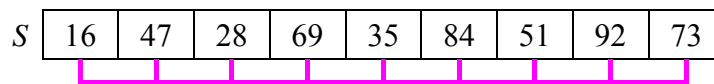
1) Gap = 4



2) Gap = 2



3) Gap = 1

**2-3-4 Quick Sort**

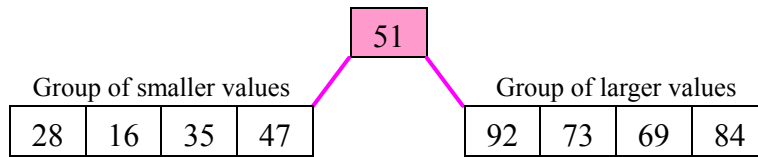
Quick sort is a high-speed sorting algorithm. The reference value is selected from the data group to be sorted. Next, data is divided into the group that has values smaller than the reference value and the group that has values larger than the reference value. After that, the reference value is selected for each group, and they are divided in a similar manner. It is a **recursive algorithm** that repeatedly calls itself until the number of elements in each group becomes 1.

By using array S ($N=9$) below, the algorithm of quick sort is considered.

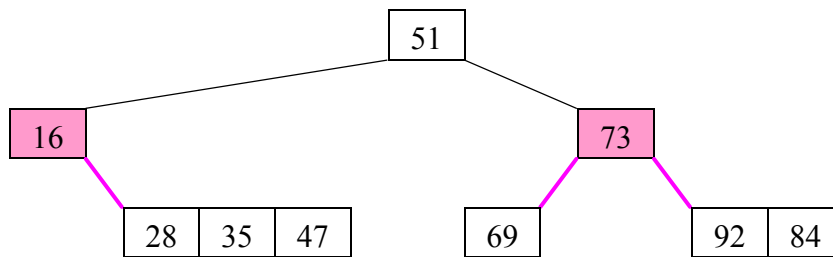
	1	2	3	4	5	6	7	8	9	
S	28	84	73	47	51	92	35	69	16	N 9

The following is the image of a procedure of sorting array S in ascending order with quick sort. The element in the middle of the target data is used as the reference value.

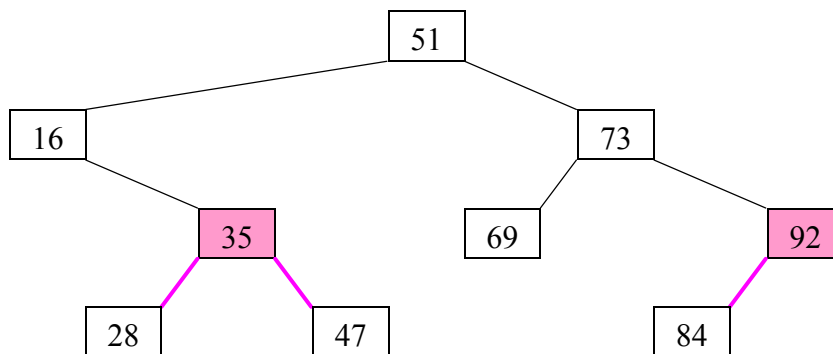
- 1) Divide $S(1)$ through $S(9)$ into the group that has values smaller than reference value $S(5)$ and the group that has values larger than reference value $S(5)$.



- 2) In the group of smaller values $\{S(1)$ through $S(4)\}$ and the group of larger values $\{S(6)$ through $S(9)\}$, decide the reference value for each group and split the groups.



- 3) In groups $\{S(2)$ through $S(4), S(8)$ through $S(9)\}$ where there are more than one elements, decide the reference value for each group and split the groups.



- 4) Sorting is complete when all groups have only one element.

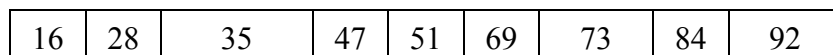


Figure 7-31 shows the flowchart that is prepared on the basis of this concept. In Figure 7-31, quick sort process is recursively called with “QSORT (index of leftmost element, index of rightmost element)”.

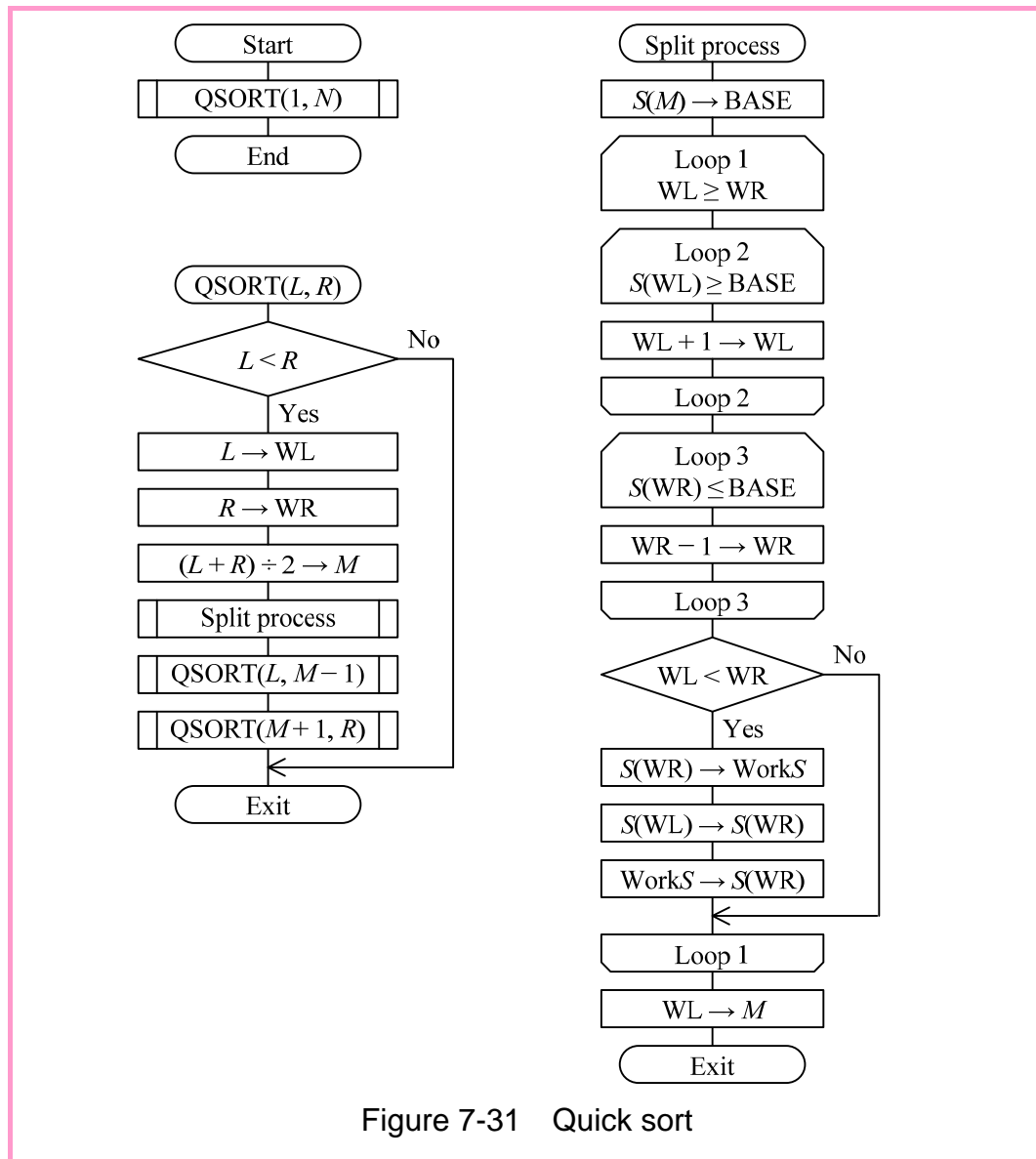
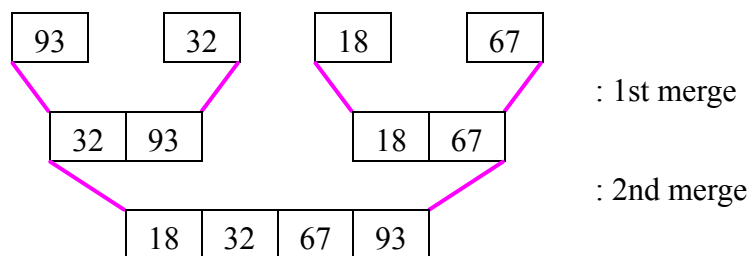


Figure 7-31 Quick sort

[Merge sort]

Merge sort is a sorting algorithm that uses **merge**, which merges two sorted data items into one without disrupting the order. This algorithm recursively performs a series of processes of splitting and merging data.



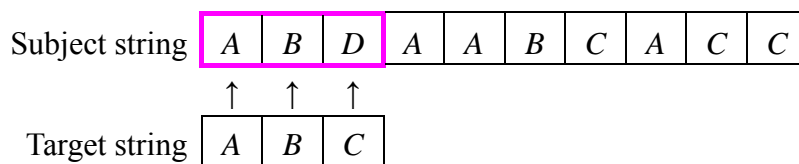
2 - 4 Other Algorithms

2-4-1 Character String Processing

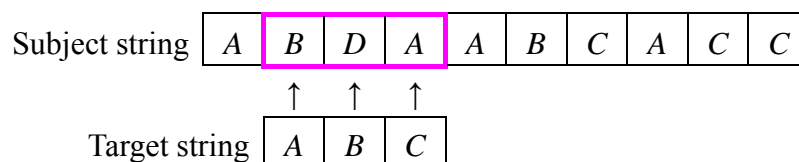
Character string processing is an algorithm that uses characters as the elements of an array. **String pattern matching** is the main string processing that searches for the target character string from the character strings that are recorded in the array. The basic concept is the same as when numerical values are recorded in the elements of an array. However, since the target value is a character string, it must be confirmed whether multiple elements simultaneously match or not.

The easiest concept of string pattern matching is the method of matching a substring (i.e., string that has the same number of characters as the target string) in array elements while one character is shifted from the first at a time.

- 1) Compare the target string with the substring that is located at the first in array elements.

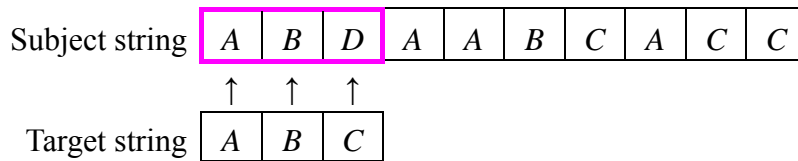


- 2) When the substring does not match, compare after the starting position is shifted by one character.

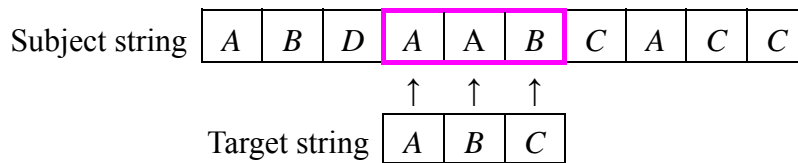


While this procedure is correct, efficiency is not very good. Therefore, the following **Boyer-Moore method** was proposed as an effective matching technique of character strings.

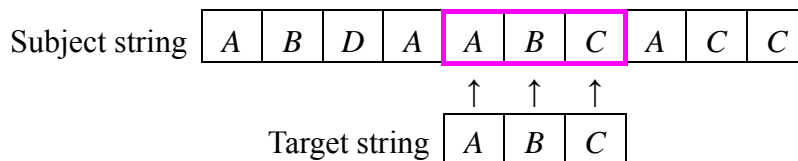
- 1) Compare the target string with the substring that is located at the first in array elements.



- 2) When the tail character of the substring does not exist in the target string, make a comparison after the starting position is shifted by the number of characters in the target string.



- 3) When the tail character of the substring exists in the target string, make a comparison by shifting the starting position so that the position of the tail character is matched with the position of the corresponding character in the target string.



Other than this string pattern matching, the string process includes compression/decompression algorithms of strings, such as the **run length method**.

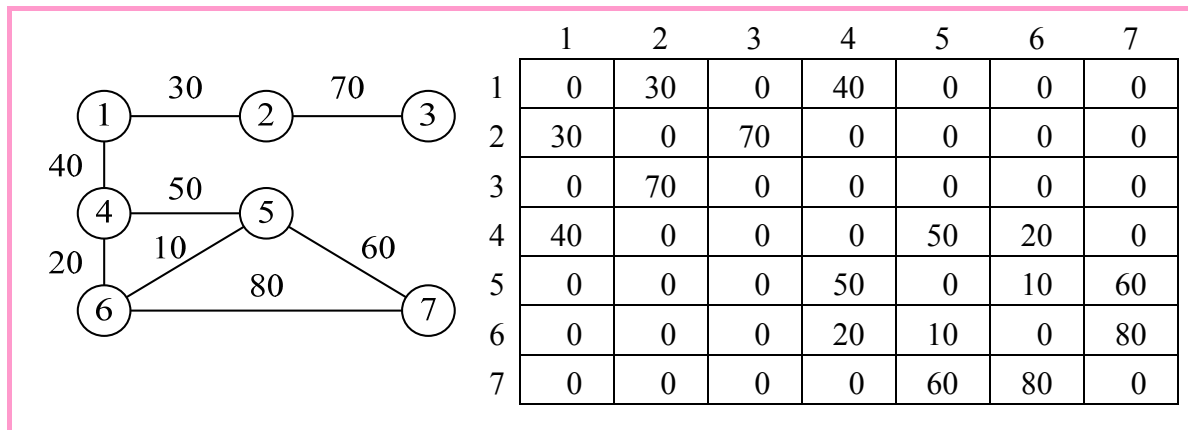
2-4-2 Graph Processing

Graph processing is an algorithm that uses a graph. **Graph** is a pictorial figure that is made of nodes and branches (or edges). (A tree structure is also a kind of graph.)

Graphs include **directed graph** where branches have a direction like a state transition diagram, and **undirected graph** where branches have no direction like a tree structure or a network diagram.

A graph can be represented with a matrix, array, list, and so on. In algorithms, mostly an **adjacency matrix** (i.e., array) is used that shows the adjacent (or linked) status of nodes. The following is an example of an adjacency matrix that shows an undirected graph (i.e., network diagram). In this example, cost (e.g., distance) of branches that connect nodes is recorded to

show that there is a branch. (0 if there is no branch.)



Representative graph processing is **route searching** that searches for the route from the origin point (i.e., node) up to the target point (i.e., node). The following are the two main methods of route searching.

- **Depth-first search**

Depth-first search searches one path until the end (i.e., dead-end or target point). When it is found that the route is not the solution (e.g., ①→②→③), it returns to the earlier point and searches for a different route. In this manner, it is a **backtrack method (backtracking)**.

- **Breadth-first search**

Breadth-first search searches the route in a hierarchical manner. It repeats the process of determining the node at one layer below adjacent to each node until the target node is found.

However, in these two methods, the route that was found may not be always the shortest route (i.e., route with minimum cost: ①→④→⑥→⑤→⑦). Therefore, as the method of **shortest route search** (or shortest path search), the **Dijkstra method** was proposed so that the lowest cost node can be finalized one by one in order.

[General procedure of Dijkstra method]

- 1) At the node adjacent to origin node, finalize the lowest cost node X.
- 2) Including the adjacent nodes to the finalized node X, finalize one lowest cost node among them.
- 3) Repeat the process of step 2) until the lowest cost of all nodes is finalized.
⇒ Decide the lowest cost (i.e., route) up to the target node.

2-4-3 Numerical Processing

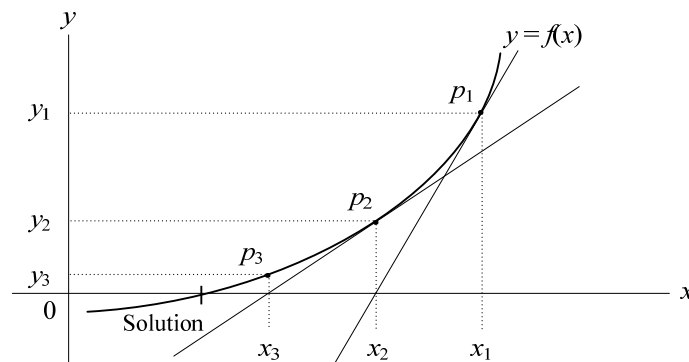
Numerical processing is an algorithm that uses numerical calculation for determining an approximate solution.

(1) Newton's method

When the approximate solution of n order equation is known where n is large, **Newton's method** determines the approximate value of a real solution while this approximate solution is modified.

[Algorithm of Newton's method]

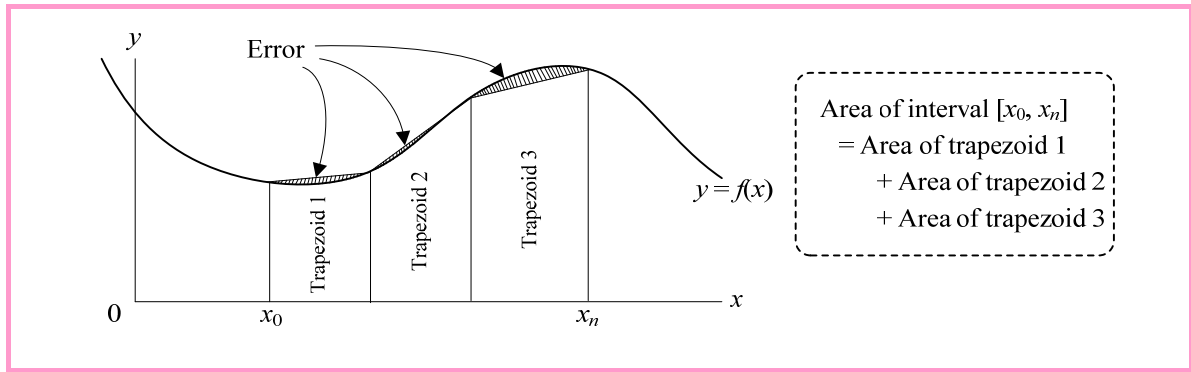
- 1) Let tangent of $y=f(x)$ and point of intersection with x axis in point $p_1(x_1, y_1)$ be x_2 .
- 2) Let tangent of $y=f(x)$ and point of intersection with x axis in point $p_2(x_2, y_2)$ be x_3 .
- 3) Similarly, determine x_4, x_5, \dots , and when the difference between x_n and x_{n+1} converges below a certain value, let x_{n+1} be the solution.



Newton's method cannot be used unless function $f(x)$ is differentiable, because it is necessary to determine the tangent of function $f(x)$. In addition, x coordinate of one point on the curve can be assigned as the initial value. However, a solution may not converge depending on how the initial value is assigned: that is, approximate value of solution might not be obtained.

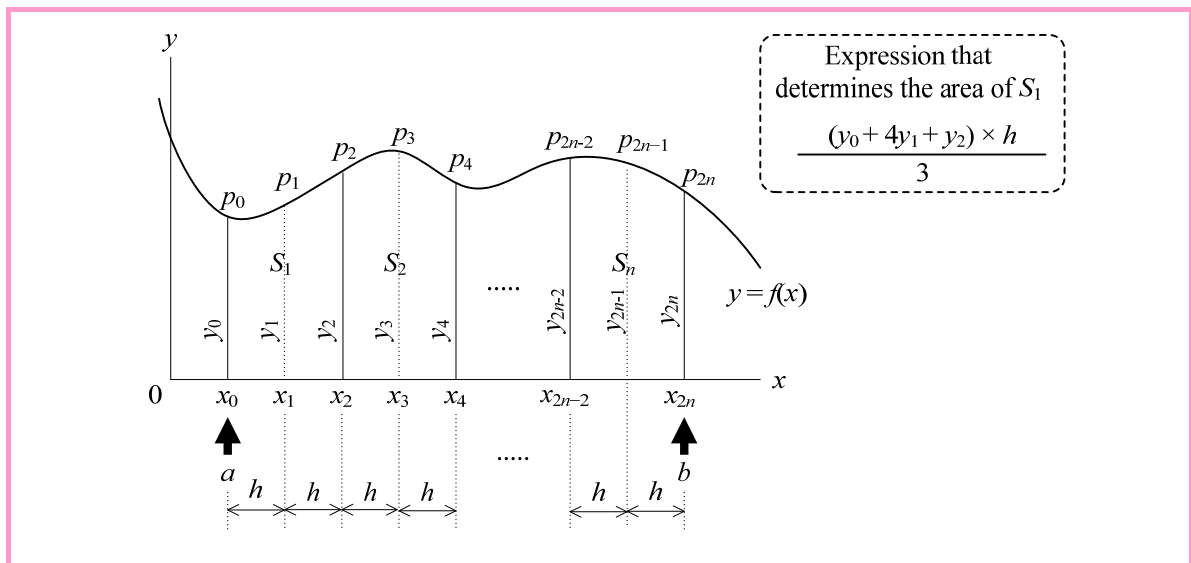
(2) Simpson's method

Trapezoidal rule is one of the methods of calculating the approximate value of an area between the x -axis and the curve over the given interval. In the trapezoidal rule, part of the area to be calculated is split into trapezoids, and the total is determined.



However, the trapezoidal rule approximates a curve with a straight line. Therefore, it suffers from the disadvantage that errors become large. The only way of reducing the error is to increase the number of trapezoids to be split. However, if the number of trapezoids to be split is large, the algorithm becomes complex and computational complexity increases.

For this reason, in **Simpson's method**, when the area after splitting is determined, the portion that corresponds to the side of $y=f(x)$ is approximated with a parabola. In specific terms, the area of S_1 portion that is split as shown below is determined with the expression by considering a parabola that passes through point p_0 , p_1 , and p_2 instead of connecting p_0 and p_2 with a straight line.



In the trapezoidal rule, the portion of the area to be determined is divided into n equal parts. However, the characteristic of Simpson's method is that it is divided into $2n$ (i.e., even number) equal parts.

2-4-4 File Processing

File processing is an algorithm that uses files where records are stored. In the basic file processing, a series of flows like “reading records from the input file”, “processing (e.g., calculation process, editing process) input records”, and “writing records in the output file” is repeated until no record is left in the input file.

- **Sorting process/merging process**

Sorting process/merging process is a process of sorting records recorded in a file and merging two sorted files into one file without disturbing the sorting order. One of the methods implements this with internal processing after the contents of records are sorted in an array, and another method uses a service program or instructions (e.g., SORT instruction, MERGE instruction) of programming languages.

- **Control break process**

When the records of a file are sorted with key items, the control break process handles the records as one group until the value of the key items change. It is used in the process (i.e., group total process) that determines the total for each group.

2 - 5 Algorithm Design

Algorithm design refers to conducting problem analysis and creating an algorithm.

In problem analysis, type and data structure of input data/output data, relation between data items are defined, and complex process conditions are summarized in a **decision table**, or other method.

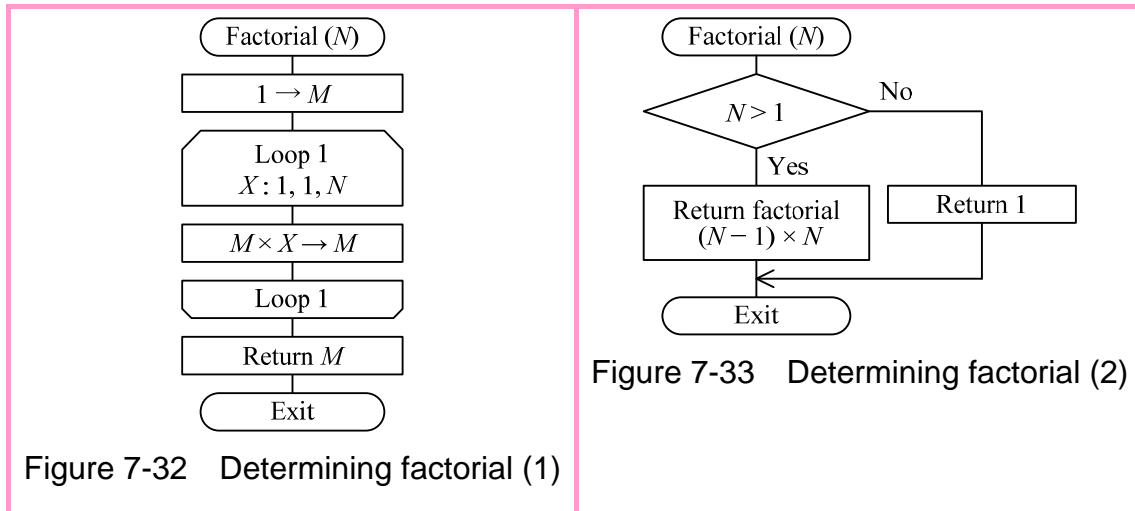
- **Decision table**

A decision table summarizes an action or process according to conditions in a table form.

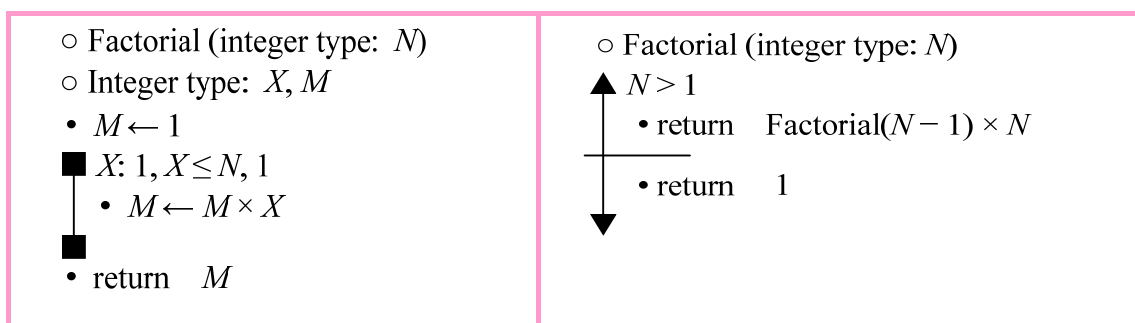
Condition title field	Condition input field (Y/N/−)				
Action title field	Action input field (X/−)	It is raining	Y	N	N
		Probability of rain is 30% or more	−	Y	N
		Carry long umbrella	X	−	−
		Carry folding umbrella	−	X	−
		Do not carry umbrella	−	−	X

For creating an algorithm, when the problem is large and complex, the **divide-and-conquer approach** is also used that splits the problems into smaller problems, and then, each of the smaller problems is solved. When the divide-and-conquer approach is used, the algorithm

becomes **recursive** in most of the cases (e.g., quick sort). For example, when the algorithm in Figure 7-32, which determines factorial of 1 through N ($N > 2$), is prepared on the basis of the concept of the divide-and-conquer approach (with a focus on what to do with respect to N), it becomes recursive as shown in Figure 7-33.



In addition, the algorithm in the figure above can also be represented by using **pseudo-language**.



Chapter 7 Exercises

Q1

The figure shows a singly-linked list. “Tokyo” is at the first element of the list, and this pointer contains the address of the next element. In addition, “Nagoya” is at the last element of the list, and this pointer contains 0. Which of the following is the process that inserts “Shizuoka” between “Atami” and “Hamamatsu”? Here, “Shizuoka” is located at address 150.

Pointer to the first element	Address	Element	Pointer
10	10	Tokyo	50
	30	Nagoya	0
	50	Shin Yokohama	90
	70	Hamamatsu	30
	90	Atami	70
	150	Shizuoka	

- Setting the pointer of Shizuoka to be 50 and the pointer of Hamamatsu to be 150
- Setting the pointer of Shizuoka to be 70 and the pointer of Atami to be 150
- Setting the pointer of Shizuoka to be 90 and the pointer of Hamamatsu to be 150
- Setting the pointer of Shizuoka to be 150 and the pointer of Atami to be 90

Q2

Which of the following is an appropriate description of a stack?

- Data that is stored last can be extracted first.
- Data that is stored first can be extracted first.
- By tracing from the first element, data can be extracted in sequence.
- By converting a search key into an address, data can be extracted.

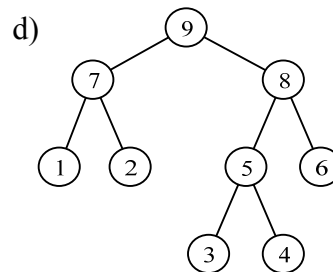
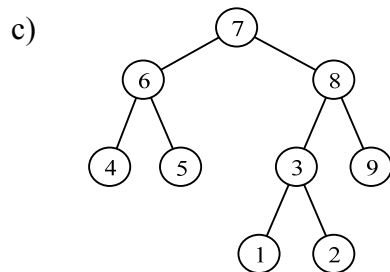
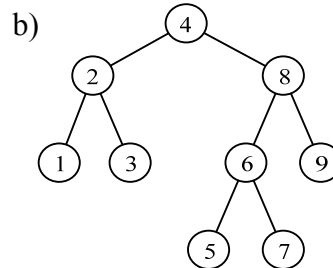
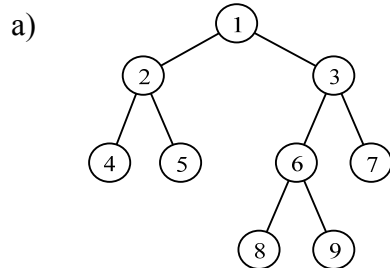
Q3

In the reverse Polish notation, expression “ $X = (A - B) \times C$ ” is represented as “ $XAB - C \times =$ ”. Which of the following is the representation of expression “ $X = (A + B) \times (C - D \div E)$ ” in the reverse Polish notation?

- | | |
|--|--|
| <ol style="list-style-type: none"> $XAB + CDE \div - \times =$ $XAB + EDC \div - \times =$ | <ol style="list-style-type: none"> $XAB + C - DE \div \times =$ $XBA + CD - E \div \times =$ |
|--|--|

Q4

Which of the following is an appropriate binary search tree? Here, numbers 1 through 9 show the value of each node.

**Q5**

Which of the following is the programming technique that is based on the principle of using only Sequence, Selection, and Iteration as the basic structure for developing an easy-to-understand program?

- | | |
|---------------------------|---------------------------|
| a) Functional programming | b) Structured programming |
| c) Parallel programming | d) Logic programming |

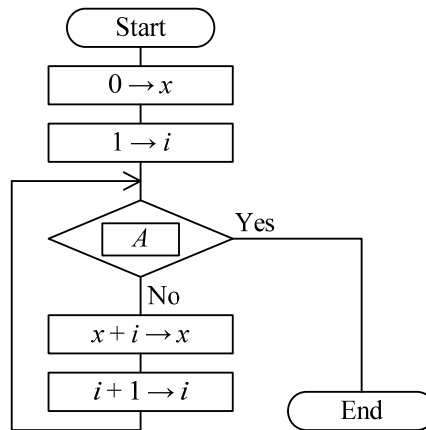
Q6

Which of the following is an appropriate explanation of the basic structure of an algorithm?

- a) “REPEAT UNTIL iteration” is used for representing an iteration process with a nested structure.
- b) “IF THEN ELSE selection” executes two processes in parallel.
- c) “CASE selection” is used to return to the process just prior to the selection.
- d) “DO WHILE iteration” may not execute the iteration process even once.

Q7

The following flowchart shows an algorithm that determines the sum (i.e. “ $1+2+\dots+N$ ”) of integer numbers from 1 through N ($N \geq 1$) and inserts the results into variable x . Which of the following is an expression that fits into blank A in the flowchart?



- a) $i = N$ b) $i < N$ c) $i > N$ d) $x > N$

Q8

Which of the following is an appropriate description of binary search?

- a) Data is correctly searched only when it is sorted in the ascending order.
 b) Data is correctly searched only when it is sorted in the ascending or descending order.
 c) Data is efficiently searched when data is sorted in the ascending or descending order.
 d) Data is correctly searched only when the number of data items is an even number.

Q12

Which of the following is an appropriate explanation of the Boyer-Moore method?

- a) It is an aggregation algorithm that reads the records that are sorted with key items in order, categorize the records that have the same value of key items as a same group, and calculates the sum of the records.
- b) It is an n -th order solution algorithm that is used when an approximate solution is found, approximates a true solution by calculating a tangential equation in an approximate solution.
- c) It is a shortest path search algorithm that decides the minimum cost up to all nodes by finalizing the node that has the minimum cost among the nodes whose paths are clarified.
- d) It is a string pattern matching algorithm that decides the amount of movement of a partial string on the basis of whether the last character of the partial string that is extracted from the search target is present in the string to be searched for or not.

Index

Numbers

— 0 —

0-address instruction 53

— 1 —

1's complementer 69
 1000BASE-FX 327
 1000BASE-T 326
 100BASE-FX 326
 100BASE-TX 326
 10BASE-T 326
 1-address instruction 53
 1-word instructions 53

— 2 —

2's complement 39
 2's complementer 70
 2-address instruction 53
 2-word instructions 53

— 3 —

3-address instruction 53
 3D 177
 3DCG 177

Alphabets

— A —

A/D conversion 170
 A/D converter 97
 Absolute addressing 57
 Absolute error 42
 Absolute path 236
 Absorption laws 65
 abstract tree structure 221
 Access 83

access control 210
 access log analysis 167, 394
 access method 230, 208
 access mode 48
 Access operation 51
 access right 209, 265
 access time 83
 Accessibility 154
 Accident 360
 account function 209
 accountability 210, 352
 Accounting management 342
 Accumulator 47
 ACID characteristics 122, 123, 261
 ActiveX 188
 acupointer 95
 adder 46
 addition 36
 additional memory 52
 Address bus 48
 address modification 56, 57
 address part 53
 address scanning 394
 Address selection feature 51
 address selection operation 52
 Addressing mode 57
 adjacency matrix 439
 administrator 210
 ADPCM (Adaptive Differential Pulse
 Code Modulation) 172
 ADSL (Asymmetric Digital Subscriber Line) 310
 ADSL modem 293
 AES (Advanced Encryption Standard) 362
 agent 342
 aggregate functions 270
 aging 201
 Ajax (Asynchronous JavaScript + XML) 217, 339
 Algorithm design 443
 alias 237
 ALU (Arithmetic and Logical Unit) 46, 61
 AM (Amplitude Modulation) 293
 Amdahl's law 134
 analog input/output board 105

batch processing system	122
bathtub curve	150
baud	294
BCA (Bridge Certification Authority)	374
BCD code (Binary-Coded Decimal code)	31
behavior method	385
benchmark program	142
benchmark test	141
benchmarking	142
best effort-type	291
bias value	39
big data	281
big O notation	425
Binary numbers	23
binary search	419, 421
binary search tree	413, 423
binary semaphore	202
binary tree	410
BIND	193
Biometric authentication	367
Biometric authentication device	97
BIOS	111
Bipolar type	49
bit	20, 227
Bit error rate	290
Bit synchronization method	295
bitmap fonts	189
Blacklist	385
Blending	176
block	81, 228
block cipher	365
Block cipher mode of operation	365
Block code	160
blocking	81
blocking factor	81
Bluetooth	109
Blu-ray disc	88
Blu-ray drives	88
BMP (Bit MaP)	172
BNF (Backus Naur Form) notation	220
BNF notation	220
boot	185
Boot sector virus	359

bootloader	185
BOOTP	317
bootstrap	185
Bot	359
Boyer-Moore method	438
bps (bit per second)	290
Branch	410
Branch instruction	53
Breadth-first search	411, 440
Bridge	325
broadcast	330
browser	338
Brute force attack	358
BSD-family OSs	193
BSDL (BSD License)	194
Bubble sort	429
Buffer memory	70
Buffer overflow	358
buffering	198
burst error	298
Bus	48, 322
Bus power method	108
bus width	48
BYOD (Bring Your Own Device)	392
byte	20, 227

— C —

C	213
C++	213
CA (Certification Authority)	371
CA (Control Area)	235
Cache memory	70
Cache poisoning	357
CAD (Computer Aided Design)	178
Cafeteria method	122
Call back	393
Cancellation of significant digit	42
Capacitance method	96
capacitor	50
capacity planning	138, 290
CAPTCHA	373
Card reader	96
cardinality	252

cascade connection.....	106, 322, 325	CIDR (Classless Inter-Domain Routing).....	331
CASE selection.....	417	CIL (Common Intermediate Language).....	222
CASL II.....	212	circuit design.....	64
CATV.....	312	circuit switched network.....	301
CCD (Charge Coupled Device).....	97	circuit switching method.....	300
CCU (Communication Control Unit).....	291	circuit switching service.....	301
CD (Compact Disc).....	87	Circular list.....	407
CDM (Code Division Multiplexing).....	305	CISC.....	75
CD-R (CD-Recordable).....	87	CISC (Complex Instruction Set Computer).....	76
CD-R/RW drive.....	88	CISO (Chief Information Security Officer).....	383
CD-ROM (CD-Read Only Memory).....	87	class library.....	187
CD-ROM drive.....	88	Classification code.....	160
CD-RW (CD-ReWritable).....	88	CLI (Common Language Infrastructure).....	215
cell.....	302	Clickjacking.....	357
Cell phone.....	387	client.....	125
Cell relay.....	302	Client/server system.....	125
center batch processing.....	122	Clipping.....	176
centralized processing system.....	124	clock frequency.....	47, 139
Centronics.....	108	Clock generator.....	47
CG (Computer Graphics).....	176	Closed batch processing.....	122
CGI (Common Gateway Interface).....	214, 339	Cloud computing.....	128
chain method.....	234, 424	cluster.....	82, 129, 134
Challenge-response authentication.....	366	Cluster analysis.....	280
channel.....	105	cluster system.....	129
Channel coding.....	296	clustering.....	129
Channel control.....	105	clusters.....	235
channel program.....	105	CMOS (Complementary MOS).....	49
character codes.....	29	CMS (Contents Management System).....	169
character corruption.....	29	CMY.....	100, 172
Character string processing.....	438	Coaxial cable.....	320
Character synchronization method.....	295	COBOL.....	212
chattering.....	92, 104	code design.....	159
Check box.....	156	Code generation.....	218, 222
check character.....	159	Code128.....	94
check constraint.....	263	coding theory.....	296
check digit.....	161	Cold site.....	129
Check digit check.....	159	Cold standby method.....	132
checkpoint.....	259	Cold start method.....	261
checkpoint file.....	259	column constraint.....	263
Checksum.....	298	Combination check.....	159
Chrome.....	193	Combination circuit.....	62
chunk.....	153	command driven method.....	255
CI (Control Interval).....	235	command interpreter.....	187

Commercial database	281	Contention	308
commercial mix	140	context-free grammar	218, 219
commit	259	Contingency plan	375
Common application software	188	continuing condition	418
Common Criteria	382	Control break process	443
common key	362	Control bus	48
Common key cryptography	362	control computers	14
Communication control unit	103	control program	185
Communication line	292	Control unit	17, 46
Communication management system	186	cookie	339
Communication server	126	co-processors	19
communication service	310	copyleft	194
communications protocols	209	CORBA	318
Commutative laws	65	CORBA (COMmon Request Broker Architecture)	188
company regulation	383	correlation name	274
Comparison instruction	53	Correlation subquery	276
compilation method	212	CPAN	193
compiler	186, 218	CPI (Cycles Per Instruction)	139
compiling	218	CPU (Central Processing Unit)	18
complement	34	CPU external bus	48
Complement notation	34	CPU internal bus	48
complementer	46, 69	cracker	356
complete binary tree	411	Cracking	356
compliance program	375	crawler	340
Componentware	188	CRC (Cyclic Redundancy Check)	298
Composite key	252	CRL (Certificate Revocation List)	372
compression	174	cross assembler	223
compression rate	175	cross browser	163
compromise	356	Cross compiler	223
Computational complexity	425	cross join	273, 274
computer crime	356	cross-checked	133
Computer virus	359	CRT display	99
Conceptual data model	248	Cryptography	361
Conceptual design of databases	251	CRYPTREC (CRYPTography Research	
Conceptual schema	255	and Evaluation Committees)	381
condenser	50	CSIRT (Computer Security Incident Response Team)	381
confidentiality	352	CSMA/CA (CSMA with Collision Avoidance)	323
Configuration management	341	CSMA/CD (Carrier Sense Multiple Access with Collision	
connection method	306	Detection)	323
connection-less method	306	CSRF (Cross Site Request Forgery)	357
Consistency	122, 261	CSS (Cascading Style Sheets)	163, 215
consistency constraint	253	CSV format	174
Content filtering	386	CTI (Computer Telephony Integration)	292

current directory	236
currently used system	132
Cursor	278
customization	189
Cyber mall	178
cycle time	83, 139
cylinder	79

— D —

D/A conversion	171
D/A converter	102
DaaS (Desktop as a Service)	129
Daisy chain connection	106
Damage	355
DAO (Disk-At-Once)	88
DAT	206
DAT (Digital Audio Tape)	90
data administration	281
data analysis	251
Data bus	48
data cache	71
Data cleaning	280
Data cleansing	280
data compression	296
data dictionary	251, 281
Data link control	305
data link layer	315, 318
Data management	208
Data mapping	282
data mart	281
Data mining	280
Data model	248
data normalization	253
Data transfer	84
data transfer rate	85
data transfer time	84, 85
Data warehouse	280
database	247
Database access layer	126
Database control function	256
Database definition function	255
Database manipulation function	255
Database security	394

Database server	126
Database software	190
DBMS (DataBase Management System)	186, 254
DCE (Data Circuit terminating Equipment)	292
DD/D	281
DD/D (Data Dictionary/Directory)	251
DDL (Data Definition Language)	255
DDoS (Distributed DoS)	356
DDR (Double Data Rate)	47
DDR SDRAM (Double Data Rate SDRAM)	50
DDS (Digital Data Storage)	90
DDS (Digital Data Streamer)	238
DDX (Digital Data eXchange)	311
de facto standard	30, 316
De Morgan's laws	65
deadlock	257
Decimal numbers	23
Decision	417
decision table	443
decoder	46
decompression	174
dedicated processors	19
default gateway	326
defragmentation	85
degraded operation	143
Delete	251
demodulation	293
Dendrogram	281
denial of service	356
Depth-first search	411, 440
dequeue (deq)	408
desktop directory	237
Destruction	355, 360
Development framework	188
device driver	105, 110
device manager	111
DHCP	316, 333
DHCP (Dynamic Host Configuration Protocol)	
server	336
DHTML (Dynamic HTML)	216
dial-up router	325
Dictionary attack	358
Difference	250

Differential backup.....	238	DMZ (DeMilitarized Zone).....	336, 391
Digital camera.....	97	DNS.....	316, 333
digital certificate.....	371	DNS (Domain Name System) server.....	336
Digital forensics.....	387	DNS cache poisoning.....	357
Digital line.....	292	DNS server.....	333
Digital signature.....	370	DNSSEC (DNS SECurity Extensions).....	390
digital video camera.....	97	DOM (Document Object Model).....	216
Digital watermarking.....	386	domain.....	249
Digitizer.....	96	DoS attack (Denial of Service).....	356
Dijkstra method.....	440	Dot impact printer.....	100
DIMM (Dual In-line Memory Module).....	52	dots.....	97, 100
diode.....	14	double precision.....	41
Direct access.....	230	double update.....	256
Direct addressing.....	59, 232	double-precision floating point number.....	38
direct control.....	104	Doubly-linked list.....	406
direct cost.....	152	DO-WHILE iteration.....	418
direct organization file.....	232	downloads.....	340
direct product.....	273, 249, 250	dpi (dots per inch).....	97, 100, 172
directed graph.....	439	DRAM (Dynamic RAM).....	50
directory.....	232, 236	drawing software.....	176, 190
directory service.....	210	Drive unit.....	103
Directory traversal.....	357	Drive-by download.....	357
Directory type.....	339	drum plotter.....	102
Disaster.....	360	DSA (Digital Signature Algorithm).....	370
disaster recovery.....	375	DSDL (Data Storage Definition Language).....	255
disk cache.....	72, 86	DSU (Digital Service Unit).....	292
Disk encryption.....	388	DTD (Document Type Definition).....	215, 216
disk mirroring.....	135	DTE (Data Terminal Equipment).....	291
disk striping.....	135	DTP (DeskTop Publishing) software.....	190
dispatcher.....	201	dual license.....	194
dispatches.....	201	dual system.....	133
Display.....	97	dumpster diving.....	356
Disposition.....	388	duplex system.....	132
Distributed database.....	279	Duplication check.....	158
distributed processing system.....	124	Durability.....	123, 261
Distributive laws.....	65	DVD (Digital Versatile Disc).....	87
divide-and-conquer approach.....	443	DVD-R.....	87
division.....	36, 233, 250	DVD-R/RW drive.....	88
division method.....	405	DVD-RAM (DVD-Random Access Memory).....	88
DKIM (Domain Keys Identified Mail).....	386	DVD-ROM.....	87
DLL (Dynamic Link Library).....	187, 224	DVD-ROM drive.....	88
DMA method (Direct Memory Access method).....	104	DVD-RW.....	88
DML (Data Manipulation Language).....	255	DVI (Digital Visual Interface).....	110

Dye sublimation thermal transfer printer	101
Dynamic access	230
Dynamic Address Translator	206
dynamic array	405
dynamic linking	224
dynamic priority scheduling	201
dynamic relocatable programs	226
Dynamic SQL	278

— E —

EAL (Evaluation Assurance Level)	382
EAP (PPP Extensible Authentication Protocol)	390
EAP-TLS (EAP Transport Layer Security)	390
EBCDIC	30
EC (Electronic Commerce)	340
ECC (Error Correcting Code)	299
ECMAScript	214
EDI (Electronic Data Interchange)	312, 340
editor	225
EDSAC	13
EEPROM (Electrically EPROM)	50
effective access time	71, 86
effective address	56
effective address calculation	56
EIA (Electronic Industries Association)	106
electronic mail	337
Electronic paper	102
electronic signatures	370
ElGamal encryption	363
Elliptic Curve Cryptography	363
e-mail address	333
embedded SQLs	255
emulator	225
Encoding	170, 294
encryption control	210
end tag	215
ENIAC	13
enqueue (enq)	408
Entity	251
Entrance and exit control	387
Environmental aspect	156
EOR	61
equijoin	273

E-R diagram	251
E-R model	251
erlang	290
Error	41, 355
Error control	295, 305
ESC/Page	102, 190
escape processing	395
ESSID (Extended SSID)	321
Ethernet	319
Ethernet header	329
ETL (Extract/Transform/Load)	280
EUC (Extended Unix Code)	31
Even parity	297
event-driven	201
Excess method	39
Exclusive control	256
Exclusive lock	257
Exclusive logical sum (Exclusive OR) operation	61
execution cycle	54
Exif (Exchangeable Image File Format)	173
Expansion bus	48
expert reviews	167
Exponent	38, 39
Extended memory	52
External bus	48
External interrupt	60
external model	248
External schema	255
External sorting	427
Extranet	340

— F —

Fail-safe	143
Fail-soft	143
fail-soft structure	134
failure	360
Failure rate	149
Failure recovery function	259
Falsification	354
Fast Ethernet	326
FAT (File Allocation Tables) file system	235
Fat client	127
FAT32	235

Fault avoidance	143	FLOPS	141
Fault management	210, 341	flow	42
Fault tolerant	143	flow control	305
fault tolerant system	133, 143	Flowchart	416
FCFS (First-Come First-Served)	198	FM (Frequency Modulation)	294
FD (Floppy Disk)	79	folder	237
FDDI	326	foolproof	144, 158
FDM (Frequency Division Multiplexing)	305	foot printing	356
feature extraction	155	foreign key	252
feed	339	Form design	159
fetch cycle	54	form overlay	159
fields	249	formal language	218, 219
FIFO	207	Format check	158
FIFO (First-In First-Out)	408	Fortran	212
file	227, 247	FPGA (Field Programmable Gate Array)	15
file access rights	210	fps (frames per second)	173
file organization format	208	FQDN	333
File processing	443	Fraction (mantissa)	38, 39
file search function	237	fragmentation	85, 204
File server	126	frame	163, 173, 302
File sharing	237	frame rate	173
file system	235	Frame relay	302
file transfer service	340	Frame relay service	302
finite automaton	218, 219	Frame synchronization method	295
finite fraction	28	fraudulent behavior	355
Firefox	193	free software	192
Firewall	336, 391	FreeBSD	193
First normalization	253	Freeware	192
First-come first-served	202	FTP	317
five major units	16	FTP (File Transfer Protocol)	340
Fixed length record	228	FTP server	340
fixed partitioning method	203	FTTH (Fiber To The Home)	311
fixed point numbers	33, 34	Full adder	69
fixed-width fonts	189	Full backup	238
flag	422	Full duplex communication	304
Flag register	47	full functional dependency	253
Flag synchronization method	295	Full text search	339
flash bootloader	185	Fully Qualified Domain Name	333
Flash memory	51	function distribution	124
flatbed plotter	102	Function layer	126
Flat-rate system	290	Functional dependency	253
flip-flop circuit	50, 62	Functional language	211
floating point numbers	33, 37	Fundamental information security policy	374

— G —

garbage collection	204
Gateway	326
Gateway server	126
GCC (GNU Compile Collection)	194
general purpose computer OS	191
general purpose processors	19
General register	47
general-purpose computer	14
generator	223
Gibson mix	140
GIF (Graphic Interchange Format)	173
Gigabit Ethernet	326
GIPS (Giga Instructions Per Second)	141
global IP address	332
GNU	194
GNU projects	194
GPIB	109
GPKI (Government PKI)	373
GPL (GNU General Public License)	194
gradation	172
Graph	439
Graph processing	439
graphic accelerator	98
graphical editors	225
Graphics software	190
Grid computing	130
Group classification code	160
Group commitment function	127
Grouping	270
Groupware	190
guarantee-type	291
guest	210
GUI	156
Gumblar	358
GZIP (GNU ZIP)	175

— H —

half adder	67
Half duplex communication	303
hamming code	135, 299
Hamming distance	300
hard real-time system	191, 123

Hardware	7
hardware faults	210
Hardware monitoring	143
hard-wired logic control	76
Harvard architecture	48, 71
hash function	233, 364, 369, 405, 424
hash organization file	233
Hash search algorithm	424
hash table	405, 424
hashing	233
hazards	379
HDB (Hierarchical DataBase)	248
HDD (Hard Disk Drive)	79
HDLC (High-level Data Link Control)	308
HDMI	110
header	328
Heap	414
heap sort	414, 429
Heuristic evaluation	167
heuristic method	385
Hexadecimal numbers	24
HFS (Hierarchical File System)	236
Hierarchical model	248
high-level languages	211
High-speed Ethernet	326
Historical file	229
hit ratio	71, 86
home directory	237
home record	234
Honey pot	386
horizontal distributed system	124
Horizontal parity	297
Horizontal/Vertical parity	297
host address	330
Host language system	255
Host-based IDS	392
hot plug	108, 111
Hot site	129
Hot standby method	132
HPC (High Performance Computing)	130
HTML (HyperText Markup Language)	128, 215
HTTP	317
HTTP (HyperText Transfer Protocol)	338

hub	106, 325
Huffman coding	296
human interface	154
Human security.....	383
Human-centered design	164
hybrid cryptography	364
hyperlink	338
hypermedia.....	169
hypertext.....	169
Hypertext database.....	282



IaaS (Infrastructure as a Service)	128
IBG (InterBlock Gap)	81
IBM/360	14
IC (Integrated Circuit).....	14
IC card	366
IC card reader.....	97
IC memory	49, 89
ICE (In-Circuit Emulator).....	225
ICMP.....	317, 333
Icon.....	121, 156
IDE (Integrated Device Electronics)	109
IDEA (International Data Encryption Algorithm)	362
Idempotent laws	65
IDF (Intermediate Distribution Frame)	292
IDS (Intrusion Detection System)	392
IEEE (Institute of Electrical and Electronics Engineers).....	108
IEEE 1394.....	108
IEEE 802.11n.....	321
IEEE 802.1X.....	373
IEEE 802.3.....	319
IF-THEN-ELSE selection.....	417
image recognition	155
Image scanner	95
image sensor	95
Imaging device	104
IMAP	316
IMAP4 (Internet Message Access Protocol version 4)...	337
Immediate addressing	58
Impact printer.....	100
Incorrect operation	355

Incremental backup	238
Independent language system	255
index	254, 403
Index addressing.....	57
Index area	231
Index register	47
indexed sequential organization file	231
Indirect addressing	59, 233
indirect cost	152
Individual application software	188
infinite fraction	28
information amount	21
Information architecture.....	153
Information aspect	155
information assets	354
information barrier free.....	166
Information leakage.....	355
Information security management	374
Information security measures criteria	374
Information security measures implementation procedures, etc.	374
Information security measures procedures.....	374
information security policy	374, 383
Infrared shielding method.....	96
Initial cost	151
initiator	197
Inkjet printer	101
Inner join	274
In-order	412
in-order execution	77
input check	158
Input unit	17, 92
Input/output bus	48
input/output channel	105
input/output control method	104, 209
Input/output instruction.....	53
Input/output interfaces.....	105
input/output interrupt	60, 104, 209
Input/output management	209
Insert	251
Insertion sort	432
In-site search	164
install.....	111

JCMVP (Japan Cryptographic Module Validation Program)	382
JDBC (Java DataBase Connectivity)	282
JEIDA (Japan Electronic Industry Development Association)	52
JIS (Japanese Industrial Standards)	29
JIS 7-bit codes	29
JIS 8-bit codes	29
JIS code	29
JIS kanji code	30
JIS X 3016	215
JIS X 8341	166
JIS Z 8530	164
JISC (Japanese Industrial Standards Committee)	29
JISEC (Japan Information technology Security Evaluation and Certification scheme)	382
JIT compiler	213
JIT compiler (Just-In-Time compiler)	223
Job management	197
job scheduler	198
job scheduling	198
job steps	197
jobs	197
Join	250
Joining	273
journal file	259
Joystick	95
JPCERT/CC (JaPan Computer Emergency Response Team / Coordination Center)	381
JPEG (Joint Photographic Experts Group)	172, 175
JPNIC (Japan Network Information Center)	329
jQuery	193
JVM (Java Virtual Machine)	213

— K —

kernel	185
kernel mode	185
Kernel program test	142
Key logger	357
Keyboard	92

— L —

L2 switch	325
-----------------	-----

L3 switch	325
label	153
LAMP	194
LAN (Local Area Network)	209, 289
LAN analyzer	342
LAN card	320
language processor	186, 217
LAPP	194
large-scale parallel processing	130
Laser printer	101
layer 2 switch	325
layer 3 switch	325
LCD (Liquid Crystal Display)	99
LDAP (Lightweight Directory Access Protocol)	210
Leaf	410
Learning functions	156
leased line service	300, 312
leased lines	300
Least privilege	395
LED (Light Emitting Diode)	14
Level	410
Lexical analysis	218
LFU	207
LGPL (GNU Lesser GPL)	194
Library	187
library modules	224
LIFO (Last-In First-Out)	407
Light pen	96
Limit check	158
line capacity	290
Line concentration method	304
Line control	305
Line printer	101
Line speed	290
line utilization rate	290
linear list	406
linear search	419
linkage editor	186, 224
linked list	405
Linker	224
linking loader	225
Linking software between application programs	186
Linux	191, 192

Linux kernel	192
List	405
List box	156
lm	102
load distribution	124
load forecast	139
load library	187, 224
load module	186, 224
Load sharing system	135
loader	225
lock system	256
Locking management	388
log data analysis method	167
log file	259
log management	383
logging function	210
Logic circuit	61
logic gates	13
Logic language	211
Logical check	158
Logical data model	248
logical design	64
Logical design of databases	252
logical expressions	64
logical functions	64
logical laws	64
Logical operation instruction	53
logical operations	61
Logical product operation	61
logical record	81, 228
Logical shift	44
Logical sum operation	61
Loop	418
Loosely coupled multiprocessor system	134
Loss	355
Loss of trailing digits	41
lossless compression	173, 175
lossy compression	173, 175
Lost-call rate	290
low-level languages	211
LRU	207
LSI (Large Scale Integration)	14
LTE (Long Term Evolution)	311

lumen	102, 190
LZH	175

— M —

MAC (Mandatory Access Control)	395
MAC (Media Access Control)	322
MAC (Message Authentication Code)	370
MAC address	322, 329, 335
MAC address filtering function	393
Mac OS	191
Machine check interrupt	60
machine language	52, 186, 211
machine language instructions	52
Macro	214
macro virus	359
magnetic card reader	96
Magnetic disk	78
Magnetic head	78
magnetic tape	90
Magnetic tape unit	90
mail client	337
mail server	337
main memory	18, 70
Maintenance function	256
malware	358
manager	342
Man-made vulnerability	361
Markup languages	215
Mask ROM	50
Master file	229
master scheduler	199
Matching check	158
Matrix switching method	96
MD5	369
MDF (Main Distribution Frame)	292
measure against computer virus	384
measure against information leakage	384
Measure against mail header injection	386
Measure against malware	384
Measure against spam	385
measure against unauthorized access	384
members	232
Memory	70

Memory bus.....	48	MODEM (MOdulator/DEModulator).....	292
memory card.....	52, 91	Modulation.....	293
memory compaction.....	204	modulation speed.....	294
Memory devices.....	49	module language system.....	255
memory hierarchy.....	73	Modulus 10.....	162
Memory interleave.....	73	monitoring.....	142, 209
memory leak.....	204	Monitoring camera.....	388
Memory management.....	203	Monolithic kernel.....	185
Memory unit.....	51	moral hazards.....	379
MEMS (Micro-Electro-Mechanical Systems).....	102	Morphing.....	176
Menu bar.....	157	MOS (Metal Oxide Semiconductor) type.....	49
merge.....	437	motion capture.....	177
merge program.....	225	motion JPEG.....	174
Merge sort.....	437	Mouse.....	95
merging process.....	443	Moving image processing.....	173
Message authentication.....	369	MP3 (MPEG1 Audio Layer3).....	171
message control.....	209	MPEG (Moving Picture Experts Group).....	173, 175
metadata.....	251, 281	MPL (Mozilla Public License).....	194
metalanguage.....	220	MPU (Micro Processing Unit).....	19
Metered-rate system.....	289	MR (Modified Read).....	175
MHz (Mega Hertz).....	47	MTBF (Mean Time Between Failures).....	144
MIB (Management Information Base).....	343	MTTR (Mean Time To Repair).....	144
MICR (Magnetic Ink Character Reader).....	93	multiboot.....	185
microcomputers.....	14	multicast.....	330
Microkernel.....	185	multi-core processor.....	74
microprocessors.....	14, 19	Multidimensional array.....	404
Microprogram control.....	76	multilink procedure.....	306
Middleware.....	186	Multimedia.....	169
MIDI (Musical Instruments Digital Interface).....	171	multimedia authoring tool.....	169
MIL (Military Specifications and Standards)		Multimedia database.....	281
symbols.....	62	multimedia system.....	177
MIMD (Multiple Instruction stream Multiple		multi-platform.....	213
Data stream).....	77	multiple backup.....	239
MIME.....	317	Multiple fixed partitioning method.....	203
MIME (Multipurpose Internet Mail Extensions).....	337	multiple selection.....	417
MIPS.....	139	multiplexing system.....	133
mirroring.....	135	multiplication.....	36
MISD (Multiple Instruction stream Single		Multi-point (or multi-drop) method.....	304
Data stream).....	77	multiprocessor system.....	129, 134
MMR (Modified Modified Read).....	175	Multiprocessors.....	77
Mnemonic code.....	160	multiprogramming.....	196, 199
MO (Magneto-Optical disc).....	90	multiscan method.....	98
Mobile communication.....	311	Multi-session.....	88

multitasking	199
multi-tenant method	128
MVS (Multiple Virtual Storage)	191
MySQL	193

— N —

n-adic numbers	23
NAND	63
NAND circuit	63
NAPT	394
NAPT (Network Address Port Translation)	332
n-ary tree	410
NAS (Network Attached Storage)	136
NAT	394
NAT (Network Address Translation)	332
Natural-language interface	155
navigation	153, 164
NCU (Network Control Unit)	292
NDB (Network DataBase)	248
need-to-know	384
Negation operation	61
negative logical product (negative AND) operation	63
negative logical sum operation	63
nested structure	419
NetBSD	193
netstat	342
NetWare	343
Network	8
network address	330
network architecture	314
network boot	185
network control program	209
network interface layer	318
Network layer	315
Network management	209
network management tool	342
Network model	248
network operations management	341
network OS	343
Network security	391
network storage	127, 136
Network-based IDS	392
Neumann architecture	48
Neumann computers	13
Newton's method	441
NFP	86
NFP (Not Found Probability)	71
NIC (Network Information Center)	329
NIC (Network Interface Card)	320
NISC (National Information Security Center)	381
NNTP	317
Node	410
non-availability	145
Non-impact printer	101
non-interactive processing system	121
non-interlaced mode	98
Non-linear image editing system	178
non-NULL constraint	252, 263
Non-preemptive	202
non-procedural language	223
Non-procedure	306
non-repudiation	352
Non-verbal interface	155
non-volatility	18, 50
NOR	63
NOR circuit	63
normalization	39
NOT	61
NOT circuit	63
NS chart	417
NTP	317
NULL	252
number of significant digits	42
Numeric check	158
Numerical processing	441
N-version programming	133

— O —

OAuth	390
Object library	187
Object orientation	213
object program	186, 217
Object-oriented language	211, 213
occurrence	249
occurrence probability	22
OCR (Optical Character Reader)	93

OCS (On-line Certificate Status Protocol)	372
Octal numbers	24
Odd parity	297
OECD (Organization for Economic Cooperation and Development)	353
OLAP (OnLine Analytical Processing)	280
OLED (Organic Light Emitting Diode) display	99
OLTP (OnLine Transaction Processing)	280
OMR (Optical Mark Reader)	93
one chip microcomputer	15, 19
One-click fraud	360
One-dimensional array	404
One-dimensional bar code	94
one-phase commitment	279
online real time processing	123
online transaction processing system	123
OODB (Object-Oriented DataBase)	281
OP25B (Outbound Port 25 Blocking)	385
Open batch processing	122
open OS	191
open source communities	195
open source libraries	193
OpenBSD	193
operand	56
operand fetching	56
operand part	53
Operational cost	152
Operations management	209
Operations management tool	186
Optical disc	86
Optical fiber cable	320
Optimization	218, 222
Optimization compiler	222
OR	61
OR circuit	62
ordered tree	414
organization format	231
Organizational aspect	156
OS	191
OS (Operating System)	184
OS for PCs	191
OS updating	385
OSD (The Open Source Definition)	192

OSI (Open Source Initiative)	192
OSI basic reference model	209, 314
OSS (Open Source Software (OSS))	192
OSS licenses	194
OTP (One-Time Password)	366
Outer join	275
outline fonts	189
out-of-order execution	77
Output unit	17, 97
overflow	42
Overflow area	232
Overlay method	205
Overlay network	340

— P —

P2P (Peer to Peer)	124
PaaS (Platform as a Service)	128
Packaged software	192
Packed decimal number	32
Packet filtering	391
Packet switching	302
Packet switching service	302
Packet writing	88
packets	302, 328
PAD	417
page description language	102
page fault	207
page printer	101
page replacement algorithm	207
Paged segment	206
page-in	206
page-out	206
Paging	206
painting software	176, 190
parallel bus	48
parallel interface	105
parallel processing system	129
parallel system	132, 147
Parallel transmission	303
Parity check	297
parity code	135
partial functional dependency	253
partitioned organization file	232

Partitioning method	203	Physical design of databases	254
Pascal	212	Physical layer	316
password	365	physical record	81, 228
Password cracking	358	Physical security	387
patch program	385	physical threat	355, 360
path	236	PIAFS (PHS Internet Access Forum Standard)	312
Pattern matching method	384	PIN code (Personal Identification Number)	366
PBX (Private Branch eXchange)	291	ping	342
PC (Personal Computers)	15	Pipeline control	74
PC card	91	pixels	172
PCI (Peripheral Component Interconnect)	49	PKI (Public Key Infrastructure)	373
PCI Express	49	PL/I	212
PCIDSS (Payment Card Industry Data Security Standard)	382	PLC (Power Line Communications)	289
PCM (Pulse Code Modulation)	170, 294	Plotter	102
PCMCIA (Personal Computer Memory Card International Association)	52, 91	Plug-in software	190
PDA (Personal Digital Assistant)	15	PM (Phase Modulation)	293, 294
PDCA cycle (Plan, Do, Check, Act)	376	PNG (Portable Network Graphics)	173
PDF (Portable Document Format)	169	PnP (Plug and Play)	111
PDL (Page Description Language)	190	pointing devices	92
PDP (Plasma Display)	99	Point-to-point method	304
PDS (Public Domain Software)	192	Polish notation	221, 412
PEAP (Protected EAP)	390	Polling/selection	308
PEAR	193	Polygon	177
Peeping	356	Polyprocessor system	135
Penetration test	394	POP	316, 407
penetration testing	382	POP3 (Post Office Protocol version 3)	337
Performance management	341	Pop-up menu	157
perils	379	port multiplier	109
peripheral devices	19	port number	328, 329
Perl	193, 214	port scanning	394
Personal Digital Assistance (PDA) security	386	POS system	4
personal digital assistants	15	PostgreSQL	193
personal information protection policies	375	Post-order	412
personal threat	355	PostScript	102, 190, 214
PGP (Pretty Good Privacy)	389	post-test iteration	418
Phishing	357	Power consumption	16
Phishing fraud	360	ppm (pages per minute)	102
photo printer	101	PPP	318
PHP	193, 214	PPPoE	318
Physical aspect	155	Precompiler	222
Physical data model	248	Preemptive	201
		prefix	22
		prefix value	332

Pre-order	411
preprocessor	222
Presentation layer	126, 315
Presentation software	190
pre-test iteration	418
primary cache	72
primary key	252
Primary key constraint	263
primary system	132
Prime area	231
Print server	126
Printer	100
Priority scheduling	198, 201
Privacy Mark (P Mark)	384
privacy policies	375
private CA	372
private IP address	332
private key	362
privilege mode	59
problem resolution data structures	407
Procedural language	211, 212
process	199
process management	199
processing privilege	265
processor	18
Product	250
profile function	209
Program control	104
program counter	47
Program file virus	359
Program interrupt	60
program library	224
program register	47
programming language	186
Progress bar	157
progressive enhancement	163, 164
Projection	250
projector	102, 190
PROM (Programmable ROM)	50
proportional fonts	189
protocol control	209
protocols	314
PROXY server	126, 326, 336

pseudo-language	444
PSW (Program Status Word)	47
public CA	372
public key	362
public key certificate	371
Public key cryptography	362, 368
public line	300
public line service	310
Pull-down menu	157
pure risk	378
PUSH	407
Python	194, 214

— Q —

QoS (Quality of Service)	291, 340
QR code	94
quadruple format	221
Quantization	170, 294
quantization bit rate	171, 294
quarantine network	392
Questionnaire survey	168
queue	407, 408
Quick sort	435
QuickTime	174

— R —

RA (Registration Authority)	371
Radio box	156
Radio button	156
RADIUS (Remote Authentication Dial-In User Service) server	336, 393
RADIUS server	373
radix	23, 38
Radix conversion	26, 233
radix conversion method	405
RAID	135
RAID0	135
RAID1	135
RAID2	135
RAID3	135
RAID4	135
RAID5	136
RAID6	136

RAM	49	Relative addressing	58
RAM disk	89	Relative error	43
RAM file	89	Relative organization	231
RAM file system	89	Relative path	236
random access	230	reliability	144, 352
RARP	317	reliability process	210
RAS	144	relocatable program	226
RAS (Remote Access Service) server	337	relocation program	226
RAS technology	387	Remote backup	388
RASIS	144	remote batch processing	122
Ray-tracing	176	removable media	80, 86
RDA (Remote Database Access)	279	Rendering	177
RDB (Relational DataBase)	248	reorganization	261
RDRAM (Rambus DRAM)	50	reorganized	414, 415
Read/write feature	51	Repeater	325
reader	197	repeater hub	325
Read-only	87	REPEAT-UNTIL iteration	418
ready state	199	Replication	280
real memory management	203	repository	281
real-time control processing system	123	residual risk	380
real-time OS	191	resolution	97, 100, 172
real-time processing system	123	response time	137
Re-authentication	395	Restart interrupt	60
record	81, 227	reusable program	226
record array	404	reverse Polish notation	221, 412
Recording area	81	reverse proxy	392
recurring fraction	28, 41	rewritable	88
recursive	444	RGB	172
recursive algorithm	435	RGB (Red, Green, Blue)	97
recursive call	408	Rich client	127
recursive program	226	RIFF	174
Redirect function	187	right to access a file	237
reentrant program	226	right to use a system	209
reference information	236	rights to use a terminal	210
referential constraint	253, 263	Ring	321
refresh	50	RIP	317, 334
Register	47	RISC	75
regular expression	218, 219	RISC (Reduced Instruction Set Computer)	75
relation	249	Risk	378
relational algebra	249	risk assessment	379
Relational model	248, 249	Risk avoidance	380
relational operations	249, 250	Risk concentration	380
Relationship	251	Risk control	380

Risk finance	380
Risk isolation	380
risk management	378
Risk optimization	380
Risk prevention	380
Risk retention	380
Risk transfer	380
RJE (Remote Job Entry) terminal	122
Robot type	340
Rollback	260
Rollforward	260
roll-in	203
roll-out	203
ROM	50
root	210, 410
root certificates	371
root directory	236
rootkit	356
Rotation shift	45
rotational latency	83
Round robin	201
Rounding error	41
route searching	440
Router	325
routing	334
Routing control	305
RPC (Remote Procedure Call)	127
RS-232C	106
RSA	363
RSS (RDF Site Summary)	339
RSS reader	339
RTP	317
Ruby	194, 214
Rule-based method	384
Run length coding	296
run length method	439
running cost	152
running state	199
runtime compiler	213, 223

— S —

SaaS (Software as a Service)	128
Salami technique (Salami slicing)	360

Sampling	170, 294
sampling frequency	170, 294
SAN (Storage Area Network)	136
Sanitizing	395
SAO (Session At Once)	88
Satellite communication service	311
SAX (Simple API for XML)	217
scavenging	356
scheduling	209
schema	255
Screen design	157
screen saver	99
Script language	211, 214
script virus	359
Scroll bar	157
SCSI	107
SD card	89
SD memory card	89
SDL (Service Description Language)	217
SDRAM (Synchronous DRAM)	50
search	83, 419
search engine	339
search time	83
search tree	413
Seat reservation system	5
Second normalization	253
secondary cache	72
secondary system	132
sector	78
Sector method	81
Secure OS	395
Secure protocols	389
Security	144
Security cable	388
Security hole	361
Security management	210, 341
security protection function	258, 394
Seek	83
seek time	83
Segment	206
Segment method	205
Selection	250, 417
Selection sort	427

selective perception	156	Shannon's sampling theorem	170
Semantic analysis	218, 221	Shared lock	257
Semaphore	202	Shareware	192
semaphore system	258	Shell	187
semiconductor memory	49, 89	Shell sort	435
sendmail	193	Shift instruction	53
sensor network	289	Shift JIS code	30
Sensors	103	Shift operation	43
sentinel method	420	shortcut	156, 237
Sequence	417	shortcut keys	158
Sequence check	159	Shortest processing time first	201
Sequence code	160	shortest route search	440
Sequence number	328	shoulder hacking	356
Sequential access	230	Side channel attack	358
Sequential circuit	62	Sign	38
Sequential method	234	signal speed	294
Sequential organization	231	Signed absolute value notation	34
sequential search	419	SIMD (Single Instruction stream Multiple Data stream)	77
Serial ATA	109	SIMM (Single In-line Memory Module)	52
serial bus	48	Simplex communication	303
serial interface	105	simplex system	131
serial printer	100, 101	simulator	178, 225
Serial transmission	303	single chip microcomputer	15, 19
series system	131, 146	Single fixed partitioning method	203
server	15, 125	single link procedure	306
server virtualization	125	single precision	41
service program	185, 224	single sign-on	393
service quality	291	single-core processor	74
Serviceability	144	single-precision floating point number	38
servlet	213	single-user mode	191
Session hijacking	357	Singly-linked list	406
Session key cryptography	364	SiP (System in a Package)	19
Session layer	315	SISD (Single Instruction stream Single Data stream)	77
SET (Secure Electronic Transactions)	389	Site map	164
set functions	270	Sizing	139
set operations	249, 250	slot-in type	86
SGML (Standard Generalized Markup Language)	215	smart media	97
SHA-1	369	smartphone	15, 387
SHA-2	369	SMTP	316
SHA-256	369	SMTP (Simple Mail Transfer Protocol)	337
SHA-512	369	SMTP-AUTH	390
Shading	176	SNMP	317, 342
Shaker sort	432		

SNMP management station	342	Star	321
SOAP	317	Star connection	106
SOAP (Simple Object Access Protocol)	217	Star schema	280
SoC (System on a Chip)	15, 19	start tag	215
Social engineering	356	Start-stop synchronization method	295
Soft real-time system	123	starvation	201
Software	8	state transition diagram	199, 219
Software development tool	186	state transition table	219
software faults	210	static array	405
software license	192	static linking	224
Software monitoring	142	static priority scheduling	201
sort program	225	Stealth function	393
sorting	272, 426	Still image processing	172
Sorting process	443	STN liquid crystal display	99
Sound input device	97	Storage capacity	80
Sound processing	170	Storage schema	255
Source coding	296	Storage unit	17
Source library	187	store-and-forward switching method	301
source program	186, 217	stored logic control	76
SP (Stack Pointer)	408	Stored procedure function	127
space complexity	425	stored-program system	13
spanning tree	325	stream cipher	365
SPEC benchmark	142	streamer	90
SPECfp	142	streaming	169
SPECint	142	strict routing	325
speculative risk	378	String pattern matching	438
SPF (Sender Policy Framework)	385	striping	135
splitter	293	structure editors	225
Spoofing	354, 356	Structured array	404
spool	198	structured database	249
Spooling	198	structured programming	417
Spreadsheet software	189	structured theorem	417
Spyware	359	Style sheet	163
SQL	255	Sub reference	275
SQL injection	357	subdirectories	236
SRAM (Static RAM)	50	Subnet	331
SSD (Solid State Drive)	89	subnet addresses	331
SSH (Secure SHell)	389	subnet mask	331
SSI (Server Side Include)	339	subquery	275
SSID (Service Set Identifier)	321	Subschema	255
SSL (Secure Sockets Layer)	373, 389	subscript	403
SSL accelerator	389	Subset	249
Stack	407	subtraction	36

super computers	14
super-pipeline	76
Superposition	233
superposition method	405
superscalar	76
superuser	210
supervisor	60
supervisor mode	185
supplicant	373
SVC (SuperVisor Call) interrupt	60
SVC interrupt	104, 209
SVG (Scalable Vector Graphics)	217
SVGA (Super VGA)	98
swap-in	205
swap-out	205
Swapping	205
switches	103
switching hub	325
switching method	300
SXGA (Super XGA)	98
symbolic languages	211
symbolic link	237
SYN synchronization method	295
synchronization	134, 294
Synchronous control	305
synonym	234, 424
synonym record	234
Syntax analysis	218, 220
syntax diagram	220
syntax tree	220, 221
Synthetic code	161
System bus	48
System file	229
system software	184

— T —

TA (Terminal Adapter)	293
table constraint	263
table design	252
tablet	96
Tablet device	387
tablet terminals	15
tags	215

tamper resistance	366
tandem system	131
Tangible information assets	354
TAO (Track At Once)	88
Tapping	354
Targeted attack	356
Task management	199
tasks	199
TCO (Total Cost of Ownership)	151
TCP	317
TCP (Transmission Control Protocol)	329
TCP header	328
TCP layer	317
TCP/IP	209
TDM (Time Division Multiplexing)	305
TDMA (Time Division Multiple Access)	324
Technical security	384
technological threat	355, 356
telecommunications carrier	289
TELNET	317
Temporal aspect	156
Temporary file	229
terminal interface	293
terminating condition	418
terminator	106, 197, 322
tethering	311
Text box	156
text editor	225
Texture mapping	176
TFLOPS (Tera FLOPS)	141
TFT liquid crystal display	99
The Internet	5
the number of significant digits	40
The Open Group	193
Theft	354, 355
Thermal printer	101
Thermal wax transfer printer	101
thesaurus search	281
thin client	127
thin client agent	127
Thin client system	127
thin client terminal	127
Third normalization	253

Third-party relay	358
thrashing	207
thread	203
Threats	354
Three basic structures	417
three primary colors of color	172
three primary colors of light	172
three-address statement	221
three-dimensional picture	177
three-schema architecture	255
three-tier client/server system	126
Throughput	138
TIFF (Tagged Image File Format)	172
Tightly coupled multiprocessor system	134
Time authentication	372
time complexity	425
time quantum	201
time slice	124, 198, 202
Time synchronous authentication	367
Timer interrupt	60
timestamp authentication	372
TLS (Transport Layer Security)	389
token bus	324
Token passing	324
token ring	324
Tomcat	194
topology	321
touch panel	95
Touch screen	95
TP (Transaction Processing) monitor	186
TPC benchmark	142
TPS (Transactions Per Second)	138
track	78
trackball	95
Traffic density	290
Transaction file	229
Transaction processing	122
Transceiver	322
Transfer instruction	53
transfer speed	290
transistors	14
transitive functional dependency	253
translator	223

Transmission control	305
Transmission control procedures	306
transparency	279
transport layer	315, 317
Trapezoidal rule	441
Trashing	356
tray type	87
Tree	322
tree connection	106
tree traversal	411
TripleDES	362
Trojan horse	359
truth table	61
TSS (Time Sharing System)	123
TTL (Transistor-Transistor Logic)	49
tuples	249
turnaround time	137
Twisted pair cable	320
Two-dimensional bar code	94
Two-part tariff system	290
Two-phase commitment	279
two-tier client/server systems	126
two-way selection	417

— U —

ubiquitous computing	289
Ubiquitous network	313
UCS-2	31
UCS-4	31
UDP	317
UFS (Unix File System)	236
unauthorized access	356
Unauthorized intrusion	360
Undefined length record	228
underflow	42
undirected graph	439
Undo	158
unicast	330
Unicode	31
uniform distribution	234
Union	250
unique constraint	252, 263
UNIVAC I	14

universal design	165
UNIX	31, 191
un-normalized form	253
unpacked decimal numbers	33
unpacking	33
Update	251
Updating	385
uploads	340
URL (Uniform Resource Locator)	333, 338
URL filtering	386
Usability	154
Usability evaluation	167
Usability test	167
usage licensing agreement	192
USB	107
USB key	388
USB memory	89
USB1.1	108
USB2.0	108
USB3.0	108
user account	210
User authentication	365
User file	230
user ID	365
user interface	121
user management	209
user mode	59, 185
User programmable ROM	50
utility program	185
UTM (Unified Threat Management)	394
UV-EPROM (UltraViolet-Erasable PROM)	50

— V —

VA (Validation Authority)	371
vaccine software	384
vacuum tubes	13
VAN (Value Added Network)	312
Variable length record	228
Variable method	81
variable partitioning method	203, 204
VB (Visual BASIC)	213
VBA (VB for Application)	214
VBScript	214

Vector computers	77
vector processor	77
version management	341
vertical distributed system	124, 125
Vertical parity	297
VGA (Video Graphics Array)	98
Video game	178
Video on demand	178
video recognition	155
View	264
Virtual mall	178
virtual memory management	203, 205
virtual surround	177
virtual table	264
VLAN (Virtual LAN)	325
VLW	76, 77
VLSI (Very Large Scale Integration)	15
VLSM (Variable Length Subnet Mask)	332
vocaloid	103
voice recognition	155
Voice synthesizer	102
VoIP (Voice over IP)	312
volatility	18, 49
volume	235
VPM (Virtual Private Network)	312
VPN (Virtual Private Network)	340, 394
VR (Virtual Reality)	177
VRAM (Video RAM)	98
VRML (Virtual Reality Modeling Language)	174
VSAM organization files	234
Vulnerabilities	360

— W —

W3C	163
WAF (Web Application Firewall)	395
WAI (Web Accessibility Initiative)	166
WAI-ARIA (Web Accessibility Initiative-Accessible Rich Internet Applications)	166
waiting line	408
waiting state	199
WAN (Wide Area Network)	209, 289
Warm site	129
Warm start method	261

WAV (RIFF Waveform Audio Format).....	172
WCAG 1.0	166
WCAG 2.0	166
WCAG 2.0 (Web Content Accessibility Guidelines 2.0)	163
WDM (Wavelength Division Multiplexing).....	305
Web	338
Web accessibility	163
Web API	187
Web beacon	339
web browser	128
web client	338
web content	169
Web design.....	162
web server.....	128, 338
Web system	128
Web usability	162
weighted average.....	71
well-known port number	329
WEP (Wired Equivalent Privacy).....	393
whitelist.....	385
Wide-area Ethernet.....	312
Wiki	169
window	121, 156
Windows	191
Windows CE.....	191
wired	289
wired LAN	319
Wired LAN interface card	103
Wired logic control	76
wireless	289
wireless LAN	321
wireless LAN access point.....	321
wireless LAN interface card	103
WMA (Windows Media Audio).....	172
word.....	20
Word processing software.....	189
work stations.....	15
workflow management	190
Working file.....	229
Worm	359
WPA (Wi-Fi Protected Access).....	393
Write-back method.....	72

write-once	87
writer.....	198
Write-through method	72
WSDL (Web SDL)	217
WWW	338
WYSIWYG (What You See Is What You Get).....	190

— X —

XGA (eXtended Graphics Array)	98
XHTML (EXtensible HyperText Markup Language)	217
XHTML Basic	217
XML (Extensible Markup Language)	216
XML database	282
XML parser	216
XML Schema	216
XML signature.....	371
XOR.....	61
XOR circuit	64
XSL (Extensible Stylesheet Language)	216
XSS (Cross Site Scripting).....	357
XY plotter	102

— Z —

Zero-day attack	358
ZIP	80, 175
zoned decimal number.....	31, 32
Zoning	387

New FE Textbook Vol.1

IT Fundamentals

First Edition: April, 2015

Photo credit

I-O DATA Device, Inc.

AIMEX Corporation Co., Ltd.

Intel K.K.

Uchida Yoko Co., Ltd.

Canon Sales Co., Inc.

Sharp Corporation

Sumitomo Electric Industries, Ltd.

Seiko Instruments Inc.

Sekonic Corporation

Sony Corporation

IBM Japan, Ltd.

NEC Corporation

Hal Corporation

Fujitsu Ltd.

Microsoft Japan Co., Ltd.

The Ricoh Company, Ltd.

* Microsoft® and Microsoft® Windows® are registered trademarks of Microsoft Corporation in the United States and other countries.

* Product names noted in this book are the trademarks or registered trademarks of their respective companies.

* Every precaution has been taken in the preparation of this book. However, the information contained in this book is provided without any express, statutory, or implied warranties. Neither the author, translator, nor publishers will be held liable for any damages caused or alleged to be caused either directly or indirectly by the book.

Original Japanese edition published by Infotech Serve Inc.

ITワールド

(ISBN978-4-906859-06-1)

Copyright © 2013 by Infotech Serve Inc.

Translation rights arranged with Infotech Serve Inc.

Translation copyright © 2015 by Information-technology Promotion Agency, Japan

Information-technology Promotion Agency, Japan

Center Office 16F, Bunkyo Green Court, 2-28-8, Hon-Komagome, Bunkyo-ku, Tokyo,
113-6591 JAPAN
