October 2022

Fundamental IT Engineer Examination (Afternoon)

Questions must be answered in accordance with the following:

Question Nos.	Q1	Q2 – Q5	Q6	Q7, Q8
Question Selection	Compulsory	Select 2 of 4	Compulsory	Select 1 of 2
Examination Time	13:30 – 16:00 (150 minutes)			

Instructions:

1. Use a pencil. If you need to change an answer, erase your previous answer completely and neatly. Wipe away any eraser debris.

2. Mark your examinee information and test answers in accordance with the instructions below. Your answer will not be graded if you do not mark properly. Do not mark or write on the answer sheet outside of the prescribed places.
   (1) **Examinee Number**
   Write your examinee number in the space provided, and mark the appropriate space below each digit.
   (2) **Date of Birth**
   Write your date of birth (in numbers) exactly as it is printed on your examination admission card, and mark the appropriate space below each digit.
   (3) **Question Selection**
   For questions **Q2** through **Q5**, and **Q7** and **Q8**, mark the Ⓢ of the questions you select to answer in the "Selection Column" on your answer sheet.
   (4) **Answers**
   Mark your answers as shown in the sample question below.

   [Sample Question]
   Which of the following should be used for marking your answer on the answer sheet?

   Answer group
   a) Ballpoint pen b) Crayon c) Fountain pen d) Pencil

   Since the correct answer is "d) Pencil", mark the answer as below:

   [Sample Answer]

Do not open the exam booklet until instructed to do so.
Inquiries about the exam questions will not be answered.

Notations used in the pseudo-language

In questions that use pseudo-language, the following notations are used unless otherwise stated:

[Declaration, comment, and process]

Notation		Description
type: *var1*, ... , *array1*[], ...		Declares variables *var1*, ... , and/or arrays *array1*[], ... , by data *type* such as INT and CHAR.
FUNCTION: *function*(*type*: *arg1*, ...)		Declares a *function* and its arguments *arg1*, ...
/* comment */		Describes a comment.
Process	*variable* ← *expression*;	Assigns the value of the *expression* to the *variable*.
	function (*arg1*, ...);	Calls the *function* by passing / receiving the arguments *arg1*, ...

	IF (*condition*) {
	  process1
	}
	ELSE {
	  process2
	}
		Indicates the selection process.
		If the *condition* is true, then *process1* is executed.
		If the *condition* is false, then *process2* is executed, when the optional ELSE clause is present.

	WHILE (*condition*) {
	  process
	}
		Indicates the "WHILE" iteration process.
		While the *condition* is true, the *process* is executed repeatedly.

	DO { *process* } WHILE (*condition*);	Indicates the "DO-WHILE" iteration process. The *process* is executed once, and then while the *condition* is true, the *process* is executed repeatedly.

	FOR (*init*; *condition*; *incr*) {
	  process
	}
		Indicates the "FOR" iteration process.
		While the *condition* is true, the *process* is executed repeatedly.
		At the start of the first iteration, the process *init* is executed before testing the *condition*.
		At the end of each iteration, the process *incr* is executed before testing the *condition*.

[Logical constants]

true, false

[Operators and their precedence]

Type of operation	Unary	Arithmetic		Relational	Logical	
Operators	+, –, not	×, ÷, %	+, –	>, <, ≥, ≤, =, ≠	and	or
Precedence	High ◄					► Low

Note: With division of integers, an integer quotient is returned as a result.
The "%" operator indicates a remainder operation.

Question **Q1** is **compulsory**.

Q1. Read the following description of security in an Internet-based order management system, and then answer Subquestions 1 through 4.

A manufacturer named Company M develops an Internet-based order management system. Company M sells its products to wholesalers. Since wholesalers also use the order management system, any security defects may result in damage to not only Company M but also the wholesalers. Therefore, Company M requested Company S, which provides security analytic services, to perform a vulnerability analysis of the order management system.

[Order management system]

The order management system's application (hereinafter, the application) runs on a Web server. Orders, shipments, and other information are stored in an order information database that runs on the database server. The application is used to reference and update the information. Files that can be downloaded to or files that were uploaded from wholesaler's PCs are stored on a disk that is connected to the Web server. Figure 1 shows the configuration of the order management system.

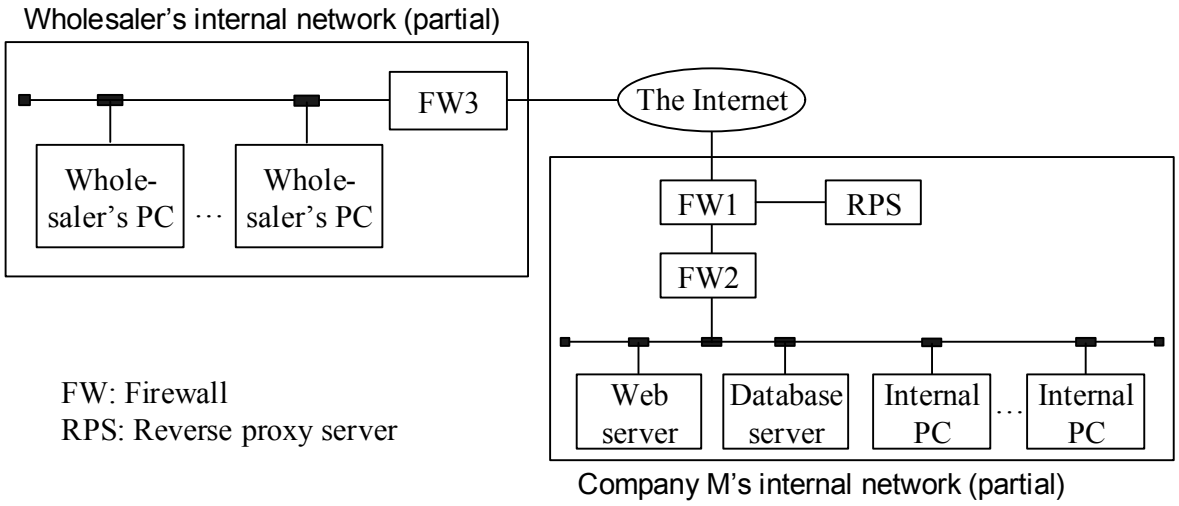


Figure 1 Configuration of the order management system

A digital certificate is configured on the RPS. The operator of the wholesaler who utilizes the order management system accesses the application via RPS from a browser on the wholesaler's PC, and logs in using a user ID and password on the login screen. At that point, the browser on the wholesaler's PC uses HTTP over TLS (hereinafter, HTTPS) for communication, and the RPS uses the digital certificate to convert the protocol from HTTPS to HTTP.

[Result of the vulnerability analysis by Company S]

In the application, measures have been taken against [A] that makes the database server execute unexpected operations to gain unauthorized access to the database. However, the vulnerability analysis by Company S reveals multiple security vulnerabilities that require measures in the application. Table 1 shows an excerpt of the findings by Company S.

Table 1 Findings by Company S (excerpt)

Finding	Cause
An operator of a wholesaler can access another wholesaler's order and shipping information.	
An operator of a wholesaler can download arbitrary files on the Web server.	(i) An issue exists in the download processing of files in the application.
Since there is a [B] vulnerability by which a script can be embedded in a Web page by an attacker, there is a possibility that an operator of a wholesaler can be led to another Web site that steals the user ID and password.	
	(ii) When an operator of a wholesaler logs in, the user ID is not locked even when the operator repeatedly enters wrong passwords.

Note: Shaded parts are not shown.

Company M took the measures to address the underlined sections (i) and (ii) in Table 1. Furthermore, Company M took measures to address the other findings by Company S, including security education for operators to prevent [C] incidents that take advantage of human errors to gain unauthorized access, and decided to launch the order management system.

Subquestion 1

The sections of the communication routes in Figure 1 are numbered 1 through 5 as shown in Table 2. From the answer group below, select the correct answer that represents all of the communication routes that use HTTPS for communication when the wholesaler's PC accesses the Web server.

Table 2 Communication routes between devices

Route number	Communication route
1	Between each wholesaler's PC and FW3
2	Between FW3 and FW1
3	Between FW1 and RPS
4	Between FW1 and FW2
5	Between FW2 and the Web server

Answer group

a) 1
b) 1, 2, 3
c) 1, 2, 3, 4
d) 1, 2, 3, 4, 5
e) 2, 3, 4
f) 2, 3, 4, 5
g) 3, 4

Subquestion 2

From the answer group below, select the appropriate answer to be inserted in each blank [ ] in the description.

Answer group for A through C

a) brute force attacks
b) cross-site scripting
c) dictionary attacks
d) directory traversal
e) DoS attacks
f) port scanning
g) social engineering
h) SQL injection

Subquestion 3

From the answer group below, select the answer that is appropriate as a measure to address the underlined section (i) in Table 1.

Answer group

a) Making the user specify only the name of the file to be downloaded, and if the file exists in the folder set for each wholesaler, performing download processing.
b) Making the user specify the file to be downloaded by using a relative path, and if the file exists, performing download processing.
c) Making the user specify the file to be downloaded by using an absolute path, and if the file exists, performing download processing.
d) On the browser on the wholesaler's PC, displaying the structure of all folders and the files on the Web server, making the user specify the file to be downloaded, and performing download processing.

Subquestion 4

From the answer group below, select the answer that is appropriate as a security incident that can be caused by the vulnerability in the underlined section (ii) in Table 1.

Answer group

a) A person could view the login operation over the operator's shoulder, and later log in.
b) A person who uses a program that automatically enters possible passwords in succession may be able to log in.
c) If an operator of a wholesaler loses a memo on which the password is written, a person who finds the memo may be able to log in.
d) If an operator of a wholesaler uses his or her birthday as a password, a person who knows the birthday can log in.

Concerning questions **Q2** through **Q5**, **select two** of the four questions. For each selected question, mark the Ⓢ in the selection area on the answer sheet, and answer the question. If three or more questions are selected, only the first two questions will be graded.

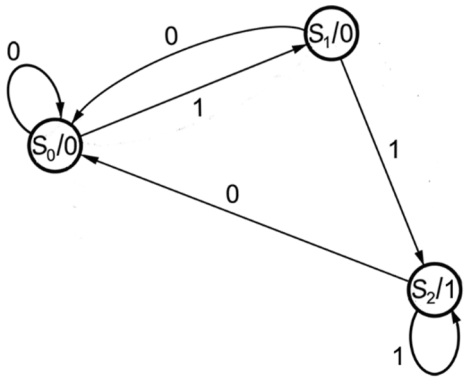
Q2. Read the following description of a logic circuit design, and then answer Subquestion. In this question, the symbols "•", "+", and "¯" indicate the logical operators AND, OR, and NOT respectively.

Engineers are planning to design a logic circuit that detects a sequence of two or more consecutive 1's in a bit string coming through an input line.

The first step in the design of a circuit is to provide a state diagram.
Figure 1 shows a state diagram for the circuit. The diagram flows as follows:

- Initially, the circuit stays at state S_0, the reset state.
- At S_0: If the input is 0, the circuit stays at S_0.
  If the input is 1, the circuit goes to S_1; the circuit receives a 1.
- At S_1: If the input is 0, the circuit goes to S_0.
  If the input is 1, the circuit goes to S_2; the circuit receives two consecutive 1s.
- At S_2: If the input is 0, the circuit goes to S_0.
  If the input is 1, the circuit stays at S_2; the circuit receives more consecutive 1s.

The state transitions in the state diagram are synchronized by the clock signal. In a single clock cycle, the circuit receives an input signal, transits the state, and puts an output signal that is defined at the destination state.



Mark	Description
○	A state
I →	Transition between states triggered by the input signal. I: Input signal (0 or 1)
Sn/Q	Sn: State name Q: Output signal (0 or 1)

Figure 1 State diagram for the circuit

Starting at the reset state S_0, for example, if the circuit gets an 8-bit sequence of input signals 0→1→0→1→1→1→0→1 during the first 8 clock cycles, then the circuit puts an 8-bit sequence of output signals 0→0→0→0→ [A], and the final state is [B].

The next step in the design of the circuit is to create a binary-coded state transition table. Table 1 shows the state names and their 2-bit state codes. The first(leftmost) bit of the state code is name V bit, and the last(rightmost) bit is named Z bit.

Table 1 State names and 2-bit state codes

State name	State code	V bit	Z bit
S_0	00	0	0
S_1	01	0	1
S_2	11	1	1

Table 2 shows the binary-coded state transition table. This table is derived from the state diagram in Figure 1.
The bits named V_P and Z_P represent the present state before the transition, and V_N and Z_N represent the next state after the transition. The bits named I and Q indicate the input and output signals, respectively.
The next state becomes the present state in the next clock cycle.

Table 2 Binary-coded state transition table

Present state		Input	Next state		Output
V_P	Z_P	I	V_N	Z_N	Q
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	1	1
1	1	0	0	0	C
1	1	1	1	1	

The next step in the design of the circuit is to determine logical expressions that derive the next state V_N and Z_N, and output Q from the present state V_P and Z_P, and input I.

(1) Determine the value of V_N from V_P, Z_P, and I

Table 2 shows that the value of V_N becomes 1 only when the set of values (V_P, Z_P, I) is (0, 1, 1) and (1, 1, 1). Therefore, the value of V_N can be determined by the following logical expression:

$V_N =$ [D] $+ V_P \bullet Z_P \bullet I$

Based on the property of logical operations, the above logical expression can be simplified as follows:

$V_N =$ [E]

(2) Determine the value of Z_N from V_P, Z_P, and I

From Table 2, the value of Z_N can be determined by the following logical expression:

$Z_N =$ [F]

(3) Determine the value of Q

Using V_N and Z_N that have been determined in (1) and (2), the value of Q can be determined by the following logical expression:

$Q =$ [ ] Note: Shaded part is not shown.

Subquestion

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the description.

Answer group for A

a) 0→1→1→0 b) 1→0→0→0 c) 1→1→0→0 d) 1→1→1→0

Answer group for B

a) S_0 b) S_1 c) S_2

Answer group for C

a)

0
0

b)

0
1

c)

1
0

d)

1
1

Answer group for D

a) $V_P \bullet \overline{Z_P} \bullet I$ b) $V_P \bullet \overline{Z_P} \bullet \overline{I}$ c) $\overline{V_P} \bullet Z_P \bullet I$ d) $\overline{V_P} \bullet Z_P \bullet \overline{I}$

Answer group for E and F

a) I b) $V_P \bullet I$ c) $\overline{V_P} \bullet I$

d) $V_P \bullet Z_P$ e) $Z_P \bullet I$ f) $\overline{Z_P} \bullet I$

Concerning questions **Q2** through **Q5**, **select two** of the four questions.

Q3. Read the following description of a relational database for a dairy store, and then answer Subquestions 1 and 2.

Store U sells dairy products such as cow's milk, skim milk, and cheese. Store U is open from 10:00 to 20:00 every day. It sells products at the store, by Web order, and by telephone order. Products are delivered free at a customer's request.

Store U builds a relational database to manage the store's business. The database comprises five tables named Product, Customer, Employee, Order, and OrderDetail.
The table structures and sample data are shown below.
Note: Underline indicates the primary key, and dashed underline indicates the foreign key.

Product table

Product table contains information on product ID, product name, quantity, and price.
The quantity indicates the stocked quantity that is updated once a day after closing the store.

ProductID	ProductName	Quantity	Price
LC050	L's cow milk (500ml)	250	150
LC100	L's cow milk (1000ml)	800	300
MP032	M's powdered milk (80g x 4)	40	800
MP080	M's powdered milk (800g)	20	2000

Customer table

Customer table contains information on customer ID, customer name, and address.

CustomerID	CustomerName	Address
C001	Alice Arnold	12 North-hill, North town
C002	Barbara Brown	345 East-lake, East town
C003	Carol Crystal	67 South-road, South town

Employee table

Employee table contains information on employee ID, employee name, and salary.

EmployeeID	EmployeeName	Salary
E001	Dan Denny	50,000
E002	Elvis Erickson	60,000
E003	Frank Fairchild	40,000

Order table

Order table contains information on order ID, customer ID, employee ID, order date, and delivery date. When customers make purchases at the store, today's date is set to the order date and delivery date.

OrderID	CustomerID	EmployeeID	OrderDate	DeliveryDate
101605	C003	E002	2022-10-16	2022-10-20
101809	C001	E001	2022-10-18	2022-10-18
102102	C002	E003	2022-10-21	2022-10-26

OrderDetail table

OrderDetail table contains information on order detail ID, order ID, product ID, quantity, and amount.

OrderDetailID	OrderID	ProductID	Quantity	Amount
101605-01	101605	LC050	2	300
101605-02	101605	LC100	10	3000
101809-01	101809	MP032	2	1600
102102-01	102102	LC100	10	3000
102102-02	102102	MP080	4	8000

Subquestion 1

From the answer group below, select the correct answer to be inserted in each blank [ ] in the SQL statement SQL1.

The store manager wants to know if the currently stocked quantities are enough or not to satisfy all the delivery requests scheduled on and after tomorrow.

At the manager's request, a database engineer plans to execute the following process as the night batch after updating the stocked quantities in the Product table.

First, extract the rows of the OrderDetail table that contain the products to be delivered on and after tomorrow, and create the ExtractedOrderDetail table.

Assuming that today is 2022-10-23. From the sample data shown above, the contents of the ExtractedOrderDetail table are as follows:

ExtractedOrderDetail table

OrderDetailID	OrderID	ProductID	Quantity	Amount
102102-01	102102	LC100	10	3000
102102-02	102102	MP080	4	8000

And then, execute the following SQL statement SQL1.

```
-- SQL1 -
SELECT P.ProductID, P.ProductName, [ A ] -
       CASE
          WHEN [ B ] IS NULL THEN 0
          ELSE [ B ]
       END AS EstimateQuantity
FROM Product P [ C ] ExtractedOrderDetail D
ON P.ProductID = D.ProductID
GROUP BY P.ProductID, P.ProductName, P.Quantity
```

When SQL1 is executed with the sample data shown above, it outputs the following result:

ProductID	ProductName	EstimateQuantity
LC050	L's cow milk (500ml)	250
LC100	L's cow milk (1000ml)	790
MP032	M's powdered milk (80g x 4)	40
MP080	M's powdered milk (800g)	16

Note:

`INNER JOIN` returns the values on the matched rows from the left and right tables.

`LEFT OUTER JOIN` returns the values on all rows from the left table and the matched rows from the right table. Columns of unmatched rows are set to `NULL`.

`RIGHT OUTER JOIN` returns the values on all rows from the right table and the matched rows from the left table. Columns of unmatched rows are set to `NULL`.

`CASE WHEN` *a* `THEN` *b* `ELSE` *c* `END` returns *b* if condition *a* is true; otherwise returns *c*.

Answer group for A and B

a) `D.Quantity`
b) `D.Quantity - P.Quantity`
c) `P.Quantity`
d) `P.Quantity - D.Quantity`
e) `SUM(D.Quantity)`

Answer group for C

a) `INNER JOIN`
b) `LEFT OUTER JOIN`
c) `RIGHT OUTER JOIN`

Subquestion 2

From the answer group below, select the correct answer to be inserted in each blank [ ] in the SQL statement SQL2.

The following SQL statement increases salary by 10% for employees who contributed to getting orders from their customers of more than total amount 10,000 in 2022. When SQL2 is executed with the sample data shown above, it increases the salary of Frank Fairchild.

```
-- SQL2 -
[ D ]
SET Salary = Salary * 1.10
WHERE EmployeeID =
      ( SELECT O.EmployeeID
        FROM Order O INNER JOIN OrderDetail D
        ON O.EmployeeID = Employee.EmployeeID
        WHERE [ E ]
          AND YEAR(O.OrderDate) = 2022
        [       F       ]
      )
```

Note: The function YEAR(*date*) returns the four-digit year of the *date* as a number.

Answer group for D

a) `CREATE Employee`
b) `CREATE Salary`
c) `INSERT INTO Employee`
d) `INSERT INTO Salary`
e) `UPDATE Employee`
f) `UPDATE Salary`

Answer group for E

a) `O.CustomerID = Customer.CustomerID`
b) `O.EmployeeID = Employee.EmployeeID`
c) `O.OrderDate IS NOT NULL`
d) `O.OrderID = D.OrderID`

Answer group for F

a)
```
  AND SUM(D.Amount) > 10000
GROUP BY O.EmployeeID
```

b)
```
  AND SUM(D.Amount) > 10000
GROUP BY O.OrderID
```

c)
```
GROUP BY O.EmployeeID
HAVING SUM(D.Amount) > 10000
```

Concerning questions **Q2** through **Q5**, **select two** of the four questions.

Q4. Read the following description of a network in an institute, and then answer Subquestions 1 and 2.

A training institute V has a computer lab named Computer Lab 1. It has 8 PCs for trainees, a PC for a trainer, a file server, a printer, a 16-port manageable 10/100 switch, and a 10/100/1000 router.

The institute has several public IP addresses that allow LAN users to access the Internet. A network administrator uses a private IP address 192.168.1.0/28 for configuring PCs, printers, and servers in Computer lab 1. The institute subscribes to SaaS for efficient training. The SaaS provides contents such as instructional videos for trainers, textbooks, material files for trainees, and online applications. Generally, there is no major issue with the total network system.

Figure 1 shows the institute network and Table 1 shows some of the IP addresses.

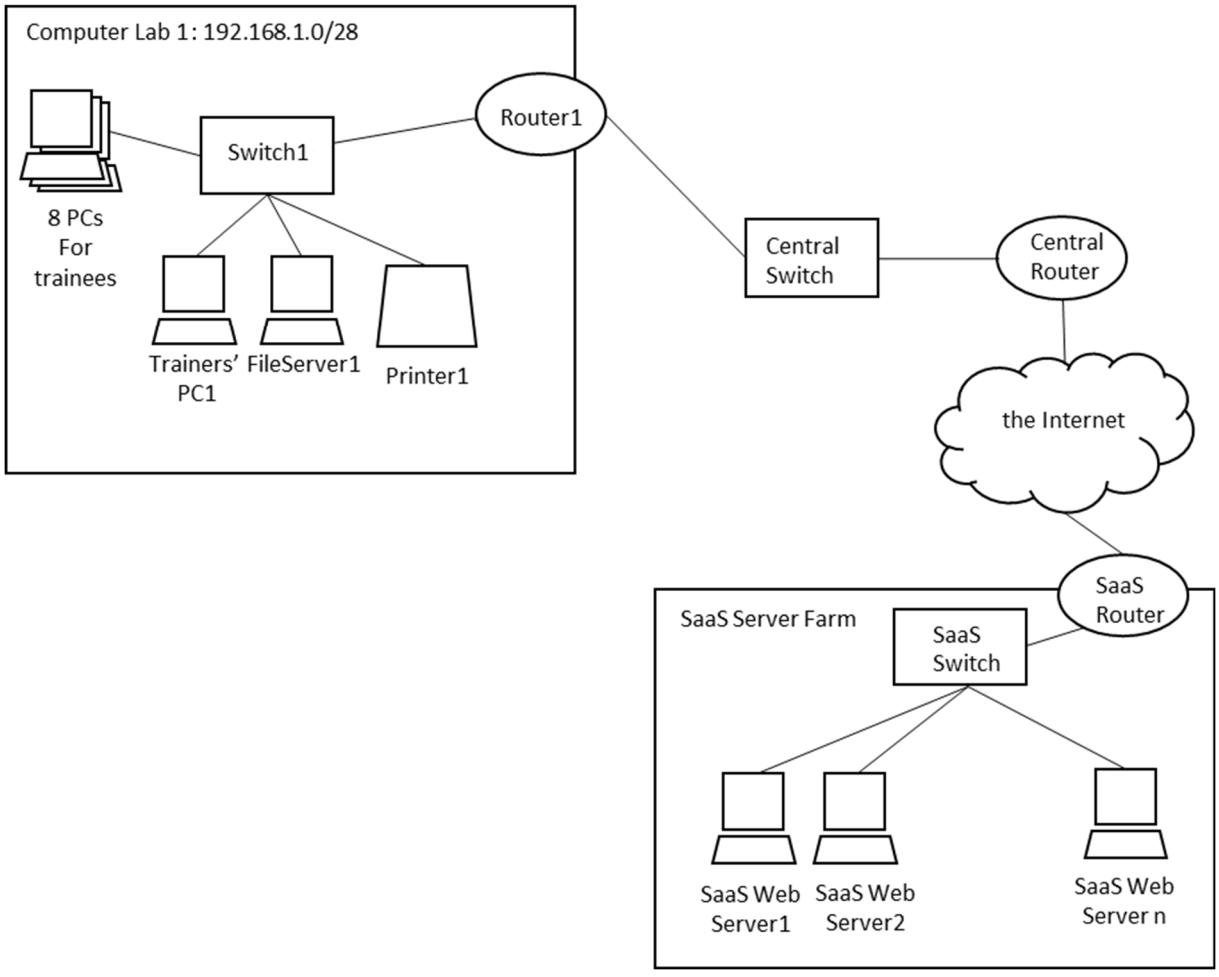


Figure 1 The institute network

Table 1 Some of the IP addresses

Location	IP address (range)	Comment
Computer Lab 1	192.168.1.0/28	Private network (Used for local communication)
Central Router	198.51.100.20	Used for NAPT
SaaS Web Server1	203.0.113.55	A public Web server on the Internet

The central router uses public IP address 198.51.100.20 to enable LAN users to browse materials on the Internet. The network administrator configured Network Address Port Translation (NAPT), also known as NAT overload, in the central router. The NAPT enables inside local IP addresses (private addresses on the LAN) to hide from the Internet and allows local users to access the Internet. When PCs on the LAN send requests/packets to SaaS Web server1, the packets pass through the central router. The central router replaces the inside local IP addresses of the packets with the public IP address of the router. It also attaches individual port numbers with each IP address to map the inside local IP address of the sender PC with the public IP address of the router. The router receives a response from the Internet along with the same port number it attached when sending the request/packet. Using this port number, the router translates the public IP address back to the inside local IP address and delivers the responded message to the right destination on the LAN.

Figure 2 shows the connection diagram and communication path. Table 2 shows port numbers and IP addresses at a different stage of communication when the packet goes from PCs to SaaS Web Server1.

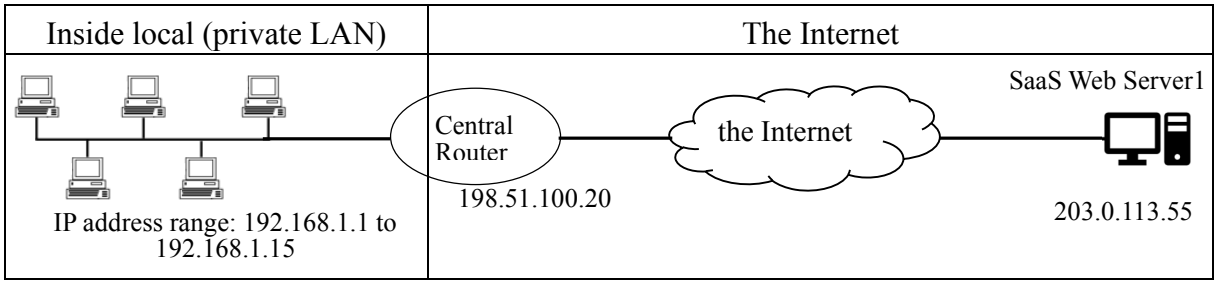


Figure 2 Connection diagram and communication path

Table 2 Port numbers and IP addresses at a different stage of communication

Inside local (address:port)		The Internet (address:port)	
Source	Destination	Source	Destination
192.168.1.5:3312	203.0.113.55:80	198.51.100.20:5589	203.0.113.55:80
192.168.1.6:4524	203.0.113.55:80	198.51.100.20:3312	203.0.113.55:80
192.168.1.7:8923	203.0.113.55:80	198.51.100.20:8923	203.0.113.55:80
192.168.1.8:12870	203.0.113.55:80	198.51.100.20:8924	203.0.113.55:80
192.168.1.7:3312	203.0.113.55:80	198.51.100.20:3313	203.0.113.55:80

When SaaS Web Server1 responds to a request of a PC on the LAN, the translation process goes in reverse order. For example, when SaaS Web Server1 wants to send a response to one of a request in Table-2, it will create a packet with source and destination address along with port number as follows:

Source address:port	Destination address:port
203.0.113.55:80	198.51.100.20:3312

In this case, the packet will finally be received by the PC whose IP address is [A].

[Establishment of Computer Lab 2]

Training institute V has established Computer Lab 2 for a larger class. The network configuration of Computer Lab 2 will be the same as Computer Lab 1, except that there will be 20 PCs for trainees. Figure 3 shows the new institute network.

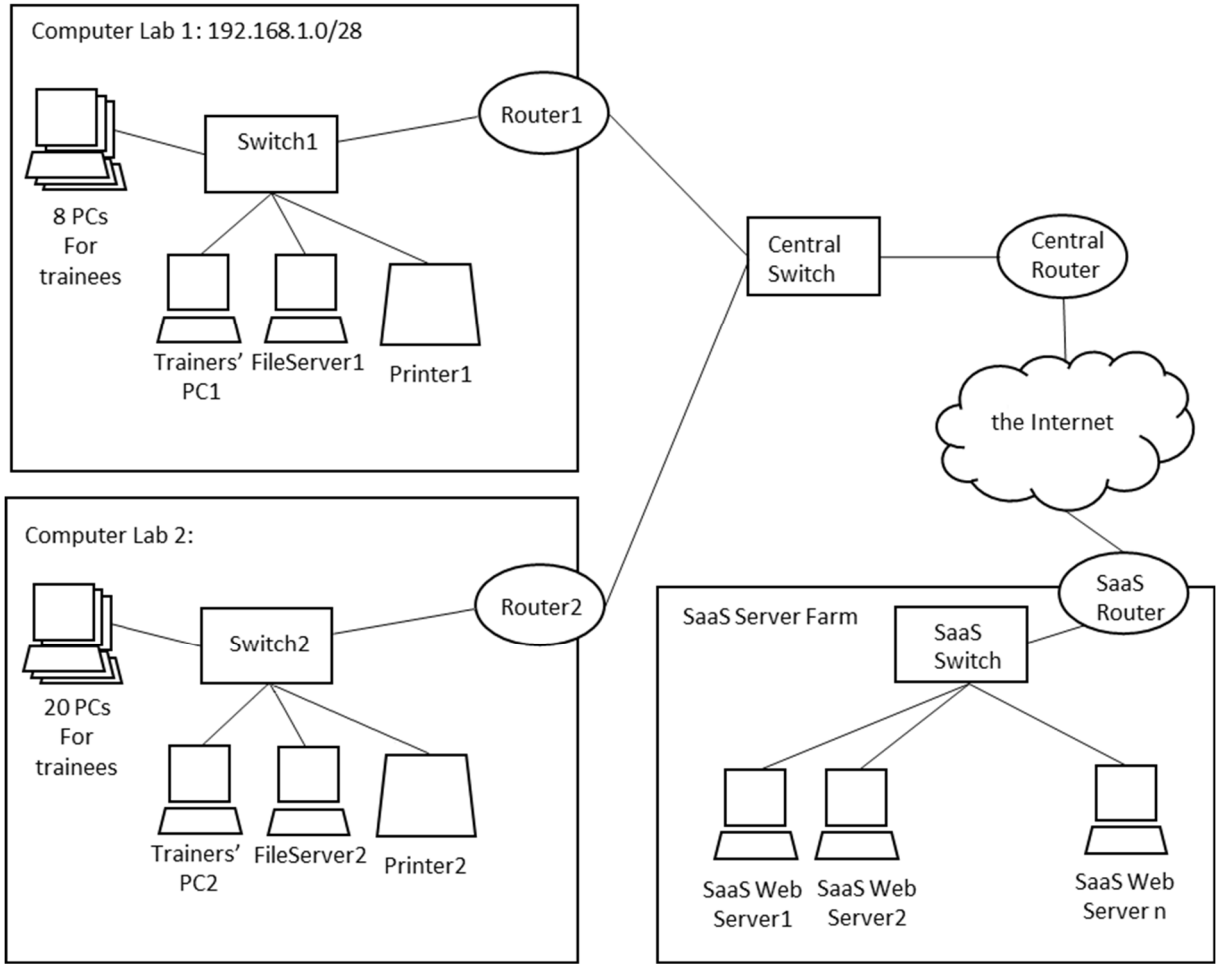


Figure 3 New institute network

The network administrator assigns the next assignable address block so that the new subnet accommodates the minimum number of IP addresses that can cover all network devices in Computer Lab 2.

The network administrator assigns the IP addresses within the IP address block to the PCs, printer, server, and router in Computer Lab 2.
The new subnet mask is [B], and the IP address range is [C]. The network administrator uses the last assignable IP address of the second subnet as the default gateway address of that network. For Computer Lab 2, [D] is used as the default gateway.

Subquestion 1

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the above description.

Answer group for A

a) 192.168.1.5
b) 192.168.1.6
c) 192.168.1.7
d) 192.168.1.8

Answer group for B

a) 255.255.255.0
b) 255.255.255.128
c) 255.255.255.192
d) 255.255.255.224
e) 255.255.255.240

Answer group for C

a) 192.168.1.16 – 31/28
b) 192.168.1.16 – 47/27
c) 192.168.1.32 – 63/27
d) 192.168.1.64 – 127/26
e) 192.168.1.128 – 255/25

Answer group for D

a) 192.168.1.30
b) 192.168.1.31
c) 192.168.1.62
d) 192.168.1.63
e) 192.168.1.126
f) 192.168.1.127

Subquestion 2

From the answer group below, select the correct answer to be inserted in each blank [ ] in the following description.

The contents and online applications provided by SaaS have recently been renewed, and the current Internet connection 1Gbps does not have enough bandwidth to use the new contents. Therefore, the decision was made to upgrade the Internet connection and replace the switches and routers.

Table 3 shows the interface specifications of the devices before and after replacement.

Table 3 Interface specifications of the devices

Device	Before replacement	After replacement
Central Router	1Gbps: 4 ports	10Gbps: 4 ports
Central Switch	1Gbps: 2 ports 100Mbps: 16 ports	10Gbps: 16 ports
Router 1, Router 2	1Gbps: 2 ports	10Gbps: 2 ports
Switch 1, Switch 2	1Gbps: 2 ports 100Mbps: 24 ports	10Gbps: 2 ports 1Gbps: 24 ports
PCs for trainees and trainer	100Mbps: 1 port	1Gbps: 1 port
File server 1, File server 2	1Gbps: 1 port	1Gbps: 1 port
Printer1, Printer 2	100Mbps: 1 port	100Mbps: 1 port

After replacement, all the hosts are connected to the 1Gbps ports of the switches, and the switches are connected to the routers via a 10Gbps port. All the devices are interconnected by UTP cat5 cables. No cables are shorter than 3 meters and longer than 20 meters.
There are some training classes where the trainers conduct classes on 3D modeling, video editing, and rendering using online applications or tools provided by the SaaS. If the training class wants to work with those online applications in real time, they need at least 1Gbps even in some cases 3 to 4Gbps bandwidth between the classroom and the SaaS server farm. The performance of the PCs is well capable of handling all the tasks. However, when the entire class wants to follow the trainer using the online applications, the bandwidth significantly drops and becomes insufficient for trainees to work smoothly. The network administrator checks all the devices, connections, and configurations, and suggests to [E] to increase the bandwidth per computer.

Answer group for E

a) replace all the CAT-5 cables with CAT-6a cables
b) replace all the crossover cables with straight-through cables
c) replace all the PCs with a PC that has higher memory and processing power
d) replace the switch with a hub to create a large collision domain

Concerning questions **Q2** through **Q5**, **select two** of the four questions.

Q5. Read the following description of an asset management system, and then answer Subquestion.

Company W is a medium-sized company in the automotive industry. It produces and sells components of cars, such as sheet metals and axles.
The accounting department is responsible for registering all assets of Company W. The accountants require up-to-date asset information to calculate asset depreciation to prepare company financial statements. Currently, there are many non-current assets, and they are long-term operating assets such as machinery and office equipment. In some cases, assets are damaged or moved to another department. However, the departments in charge of such assets did not inform the accounting department to update the information.

In the past, each asset had a sticker that indicated only the asset ID. The asset information was kept in spreadsheets. For updating asset information, the accounting department and the departments in charge of the assets must work together in finding, referencing, and verifying those assets. This was quite a time-consuming process.
Therefore, the asset management system is developed for retrieving and verifying asset information. The system manages the history of asset information. Even if an asset disappears, the record of that asset is not deleted.

Four main functions of the asset management system are shown below. To access these functions, an accountant or a staff logs in to the system by entering a user name and password.

(1) Register assets

An accountant registers newly-purchased assets.

(i) After login to the system, the accountant creates asset records for the new assets. The attributes of the asset record include asset ID, asset description, purchase date, seller, depreciation, cost, asset condition, location, department, and checking flag. The asset condition is set to Normal and the checking flag is set to 0 (not checked) by the system.

(ii) Based on the accountant's request, the system generates QR code stickers for the assets. The accountant prints the QR code stickers and attaches them to the assets.

(2) Check assets

An accountant activates the asset checking event. Once a year, the accounting department asks staff in the departments in charge of the assets to check the assets.

(i) After login to the system, the accountant activates the asset checking event.

(ii) The system resets the checking flag of all assets to 0 (not checked).

(3) Report assets

Staff reports the result of asset checking. The staff in the department in charge of the assets checks and reports asset conditions using a mobile terminal.

(i) After login to the system, the staff performs a) to d) for each asset.

a) Scan the QR code and display the asset information on the screen.

b) Select the asset condition: Normal, Need repair, or Discard.

c) Select the location of the asset from the list displayed by the system.

d) After finishing checking, set the checking flag of the asset to 1 (checked).

(4) Update assets

An accountant updates the asset records after the reporting deadline.

(i) After login to the system, the accountant displays the list of assets whose checking flag is 1 (checked).

(ii) The accountant confirms the check result of each asset and sets the checking flag to 2 (updated). If the asset information is incomplete, state the reason and set the checking flag to 0 (not checked).

(iii) Finally, if there are assets whose checking flag is [A], the accountant requests the concerned staff to check and report again for those assets.

Figure 1 shows the use case diagram of the asset management system.

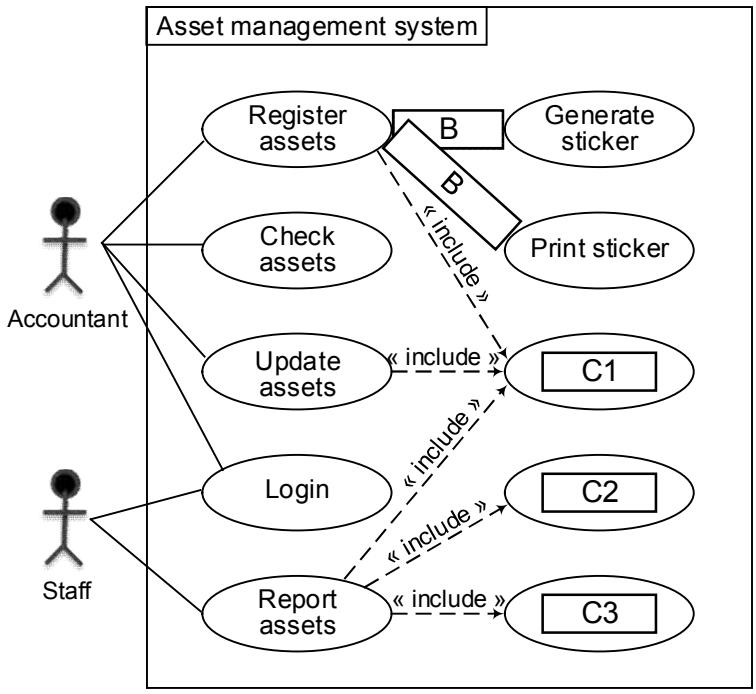


Figure 1 Use case diagram of the asset management system

Note:

X ← « extend » - Y: Y is an additional (optional) function of X.

X ← « include » - Y: Y has X, i.e., X is included in Y.

Table 1 shows the use case description of Register assets.

Table 1 Use case description of Register assets

Use case name	Register assets	
Primary actor	Accountant	
Secondary actor	-	
Input	asset description, purchase date, seller, depreciation, cost, D	
Output	The summary of asset information, QR code (optional)	
Trigger	The accountant clicks the "Register assets" button on the screen	
Availability	Always	
Pre-condition	Successful login	
Post-condition	The summary of asset information is displayed	
Basic flow	Actor	System
	1. Displays the asset filling form.	
	2. Selects the asset type, department in charge, and location.	
	3 Fills in the asset description, purchase date, seller, depreciation, and cost.	
	4. Clicks the "Register" button.	
		5. Assigns the unique asset ID,sets E, then displays the summary information.
	6. Clicks "Print QR and Logoff" or "Logoff".	
		7. If "Print QR and Logoff" is clicked, calls functions Generate QR and Print QR. Then terminate the process.

Subquestion

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the above description.

Here, the answers to be inserted in C1, C2 and C3 should be selected as the correct combination from the answer group for C.

Answer group for A

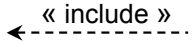
a) 0 b) 0 or 1 c) 0 or 2

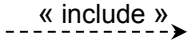
d) 1 e) 1 or 2 f) 2

Answer group for B

a)

b)

c) 

d) 

Answer group for C

	C1	C2	C3
a)	Scan QR code	View location	View assets
b)	View assets	Scan QR code	View location
c)	View location	View assets	Scan QR code

Answer group for D

a) asset ID, checking flag, department, location
b) asset ID, department, location
c) checking flag, department, location
d) department, location

Answer group for E

a) the asset condition to Normal
b) the asset condition to Normal and the checking flag to 0
c) the asset condition to Normal and the checking flag to 1
d) the checking flag to 0
e) the checking flag to 1

Question **Q6** is **compulsory**.

Q6. Read the following description of programs, and then answer Subquestions 1 and 2.

Counting sort is one of the algorithms for sorting, and it is most suitable for sorting data where the data values exist within a narrow range of non-negative integers starting at 0, such as age and exam score.
Counting sort proceeds as follows:
Step 1: Count the number of occurrences of each data value in an input sequence.
Step 2: Determine the starting position of each data value in an output sequence.
Step 3: Copy each data in the input sequence to the output sequence.

[Program Description]
The program sorts exam scores of 10 students in ascending order.
The input array `StuNo` contains the student numbers, and `Score` contains the corresponding exam scores. Exam scores range between 0 and 10.
The sorted results are stored in the output arrays `SortedStuNo` and `SortedScore`.

Input arrays:

	0	1	2	3	4	5	6	7	8	9
StuNo[]	11	12	13	14	15	16	17	18	19	20
Score[]	7	8	5	9	8	4	8	6	9	7

Output arrays:

	0	1	2	3	4	5	6	7	8	9
SortedStuNo[]	16	13	18	11	20	12	15	17	14	19
SortedScore[]	4	5	6	7	7	8	8	8	9	9

[Program]

```
FUNCTION: CountingSort() {
   INT: StuNo[10] ← {11,12,13,14,15,16,17,18,19,20},
        Score[10] ← { 7, 8, 5, 9, 8, 4, 8, 6, 9, 7},
        SortedStuNo[10],
        SortedScore[10],
        Count[11] ← { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
        Index[11] ← { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
   INT: i, n

   Execute the statements in [Program - Step 1]
   Execute the statements in [Program - Step 2]
   Execute the statements in [Program - Step 3]
}
```

[Program - Step 1]

```
FOR (i ← 0; i < 10; i ← i + 1) {
    [ A ] ← [ A ] + 1;
}
```

[Description of Program - Step 1]

It counts the number of occurrences of each data value in the input array `Score` and stores the result in the array `Count` as shown below. The index of the array `Count` represents the data value.

	0	1	2	3	4	5	6	7	8	9	10
Count[]	0	0	0	0	1	1	1	2	3	2	0

[Program - Step 2]

```
n ← 0;
FOR (i ← 0; i < 11; i ← i + 1) {    /*** α ***/
   IF (Count[i] > 0) {
      Index[i] ← [ B ];
      n ← n + [ C ];
   }
}
```

[Description of Program - Step 2]

It calculates the lowest storing position of each data value in the output arrays and stores the result in the array `Index` as shown below. The index of the array `Index` represents the data value.

	0	1	2	3	4	5	6	7	8	9	10
Index[]	0	0	0	0	0	1	2	3	5	8	0

[Program - Step 3]

```
FOR (i ← 0; i < 10; i ← i + 1) {
   n ← [ D ];                /*** β ***/
   SortedStuNo[n] ← StuNo[i];
   SortedScore[n] ← Score[i];
   [ E ] ← [ E ] + 1;
}
```

[Description of Program - Step 3]

It moves the data in the input arrays to the output arrays.

[Characteristics of the program]

- The sort algorithm used in this program is stable. Sort algorithms are classified into two types: stable and unstable. A stable sort algorithm sorts the elements with the same value in the same order they appear in the input.
- The calculation performance of the program is expressed as [F] in *O*-notation.

Subquestion 1

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the above program and description.

Answer group for A

a) `Count[i - 1]`
b) `Count[i]`
c) `Count[Score[i - 1]]`
d) `Count[Score[i]]`

Answer group for B

a) `i`
b) `i + Count[i - 1]`
c) `n`
d) `n + Count[i - 1]`

Answer group for C

a) `i`
b) `Count[i]`
c) `Index[i]`
d) `Score[i]`

Answer group for D

a) `Index[i]`
b) `Index[Score[i]]`
c) `n + 1`
d) `Score[i]`

Answer group for E

a) `Index[i]`
b) `Index[n]`
c) `Index[Score[i]]`
d) `Index[Score[n]]`

Answer group for F

a) $O(1)$
b) $O(n)$
c) $O(n \log n)$
d) $O(n^2)$

Subquestion 2

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the following description.

The program will be modified to sort the input data in descending order of exam scores. Two methods, method 1 and method 2, are considered.

Method 1: In [Program - Step 2], replace the line marked `/*** α ***/` with the following line.

```
FOR (i ← 10; i >= 0; i ← i - 1) {
```

Method 2: In [Program - Step 3], add the following line immediately after the line marked `/*** β ***/`.

```
n ← [ G ];
```

When the programs modified by method 1 and method 2 are executed, the output results will be either of the following. Concretely, [H].

Output arrays: Stable

	0	1	2	3	4	5	6	7	8	9
SortedStuNo[]	14	19	12	15	17	11	20	18	13	16
SortedScore[]	9	9	8	8	8	7	7	6	5	4

Output arrays: Unstable

	0	1	2	3	4	5	6	7	8	9
SortedStuNo[]	19	14	17	15	12	20	11	18	13	16
SortedScore[]	9	9	8	8	8	7	7	6	5	4

Answer group for G

a) `9 - n`
b) `10 - n`
c) `11 - n`
d) `i + 10 - n`

Answer group for H

a) both method 1 and method 2 are stable
b) both method 1 and method 2 are unstable
c) method 1 is stable and method 2 is unstable
d) method 1 is unstable and method 2 is stable

Concerning questions **Q7** and **Q8**, **select one** of the two questions.
Then, mark the Ⓢ in the selection area on the answer sheet, and answer the question.
If two questions are selected, only the first question will be graded.

Q7. Read the following description of a C program, and then answer Subquestions 1 and 2.

Heap is a data structure using a complete binary tree. The complete binary tree is a tree satisfying the following conditions:

- Each node in the tree may have at most two child nodes.
- The number of nodes at depth N is 2^N except at the bottom of the tree.
- The nodes at the bottom of the tree are filled from left to right.

Heap satisfies the following condition in addition to the above:

- Any parent node in a complete binary tree has a greater value than or equal to both of its child nodes have.

Figure 1 shows an example of a heap.

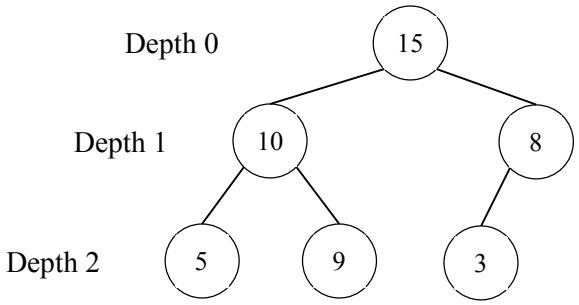


Figure 1 Example of a heap

A heap can be built onto an array by regarding i-th element of the array as the parent node of the $(2 \times i + 1)$-th and $(2 \times i + 2)$-th elements. Figure 2 shows how the heap in Figure 1 is built onto an array.

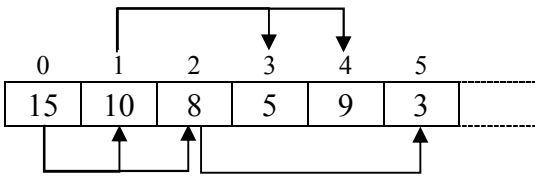


Figure 2 How the heap in Figure 1 is built onto an array

Heap sort is a sorting technique based on the heap. The following steps show how elements in an array are sorted by heap sort.

(1) For each element of a given array, apply (i) through (iii) to create the heap.
   (i) Include an element to the heap at the rightmost node at the bottom of the heap.
   (ii) Check if the element has a greater value than its parent has. If it does, swap them.
   (iii) Fix the heap by recursively applying (ii) upward while swapping occurs and the element is not the first of the array.

(2) While the number of nodes in a heap is greater than 1, apply (i) through (iv).
   (i) Swap the root node with the rightmost node at the bottom of the heap.
   (ii) Exclude the rightmost node at the bottom of the heap from the heap.
   (iii) Check if the parent node has less value than either of the child nodes has. If it does, swap the root node with the child node with the greatest value.
   (iv) Fix the heap by recursively applying (iii) downward while swapping occurs and the node has a child node.

As a result of the steps above, the elements are sorted in ascending order within the array. Figure 3 shows how the heap sort is applied to the heap in Figure 2.

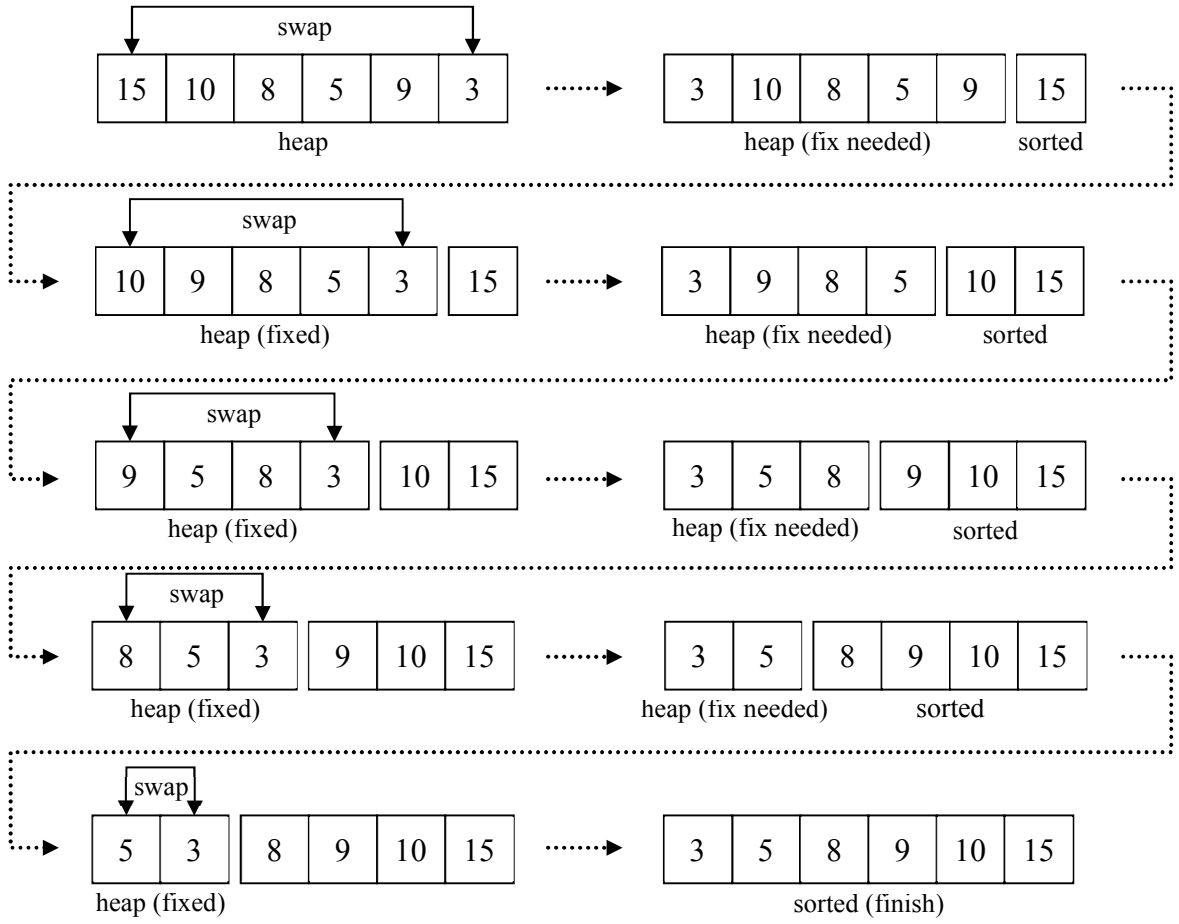


Figure 3 How heap sort is applied to the heap in Figure 2

[Program Description]

This program sorts students by their exam score using the heap sort. Each student is assigned an ID, a serial number starting at 0.

The program uses the user-defined data type, user-defined functions, and the compile-time operator.

User-defined data type:

Name	Description
`Data_t`	A user-defined data type represents the ID and value of the data.

User-defined functions:

Name	Description
`void swap(Data_t *a, Data_t *b)`	Swaps the values pointed by `a` and `b`.
`void heap_up(Data_t *data, int child)`	Recursively fixes the heap. The processing starts from the `child` and proceeds upward.
`Void heap_down(Data_t *data, int heap_size, int parent)`	Recursively fixes the heap. The processing starts from the `parent` and proceeds downward.
`void rank_by_score(Data_t *data, int n)`	Sorts `n` elements of `data` in ascending order using functions `heap_up` and `heap_down`.
`void print_data(Data_t *data, int n)`	Prints `n` elements of `data`.

Compile-time operator:

Name	Description
`sizeof(`*data*`)`	During the compile-time, determines the size, in bytes, of *data*.

When this program is executed, the following results will be printed out.

```
Students' scores (unsorted)
ID:     0,   1,   2,   3,   4,   5,   6,   7,   8,   9
Value:  75,  80,  82,  68,  56,  97,  65,  80,  81,  92
-----------------
Students' scores (sorted)
ID:     4,   6,   3,   0,   1,   7,   8,   2,   9,   5
Value:  56,  65,  68,  75,  80,  80,  81,  82,  92,  97
-----------------
```

[Program]

```
#include <stdio.h>

typedef struct Data {
    int id;
    int value;
} Data_t;

void swap(Data_t *a, Data_t *b);
void heap_up(Data_t *data, int child);
void heap_down(Data_t *data, int heap_size, int parent);
void rank_by_score(Data_t *data, int n);
void print_data(Data_t *data, int n);

void swap(Data_t *a, Data_t *b) {
    Data_t t = *a;
    *a = *b;
    [    A    ];
}

void heap_up(Data_t *data, int child) {
    int parent = [    B    ];
    if (data[child].value > data[parent].value) {
        swap(&data[child], &data[parent]);
        if ([    C    ]) {
            heap_up(data, parent);
        }
    }
}

void heap_down(Data_t *data, int heap_size, int parent) {
    int l_child = parent * 2 + 1;
    int r_child = parent * 2 + 2;
    int child = l_child;
    if ([    D    ]) {
        if (r_child < heap_size &&
            data[l_child].value < data[r_child].value) {
            child = r_child;
        }
        if (data[parent].value < data[child].value) {
            swap([    E    ]);
            heap_down(data, heap_size, child);
        }
    }
}
```

```
void rank_by_score(Data_t *data, int n) {
    int i, heap_size = 0;
    printf("Students' scores (unsorted)");
    print_data(data, n);
    for (i = 0; i < n; i++) {
        heap_up(data, i);
        heap_size++;
    }
    for (i = n - 1; i > 0; i--) {
        swap(&data[0], &data[i]);
        heap_size--;
        heap_down(data, [ F ]);
    }
    printf("Students' scores (sorted)");
    print_data(data, n);
}

void print_data(Data_t *data, int n) {
    int i;
    for (i = 0; i < n; i++) {
        if (i == 0)
            printf("\nID:     %d", data[i].id);
        else
            printf(",\t%d", data[i].id);
    }
    for (i = 0; i < n; i++) {
        if (i == 0)
            printf("\nValue:  %d", data[i].value);
        else
            printf(",\t%d", data[i].value);
    }
    printf("\n------------------\n");
}

int main() {
    Data_t data[] = {
        {0, 75}, {1, 80}, {2, 82}, {3, 68}, {4, 56},
        {5, 97}, {6, 65}, {7, 80}, {8, 81}, {9, 92}
    };
    rank_by_score(data, [ G ]);
}
```

Subquestion 1

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the program.

Answer group for A

a) `*a = t`
b) `*b = *a`
c) `*b = t`
d) `t = *a`
e) `t = *b`

Answer group for B

a) `(child + 1) / 2`
b) `(child - 1) / 2`
c) `0`
d) `child + 1 / 2`
e) `child - 1 / 2`

Answer group for C

a) `parent == 0`
b) `parent == child`
c) `parent > 0`
d) `parent > child`

Answer group for D

a) `l_child < heap_size`
b) `l_child == 0`
c) `l_child > 0`
d) `l_child > heap_size`

Answer group for E

a) `&data[l_child], &data[r_child]`
b) `&data[parent], &data[child]`
c) `&data[parent], &data[l_child]`
d) `&data[parent], &data[r_child]`

Answer group for F

a) `heap_size, 0`
b) `heap_size, i`
c) `heap_size, n - 1`
d) `n, 0`
e) `n, i`
f) `n, n - 1`

Answer group for G

a) `sizeof(data)`
b) `sizeof(data) * sizeof(Data_t)`
c) `sizeof(data) + sizeof(Data_t)`
d) `sizeof(data) / sizeof(Data_t)`

Subquestion 2

From the answer group below, select the correct answer to be inserted in the blank [H] in the following description. It is assumed that the correct answer is inserted in blank [G].

The definition of the function `main` is replaced with the following 6 lines.

```
int main() {
    Data_t data[] = {
        {0, 84}, {1, 84}, {2, 96}, {3, 84}
    };
    rank_by_score(data, [ G ]);
}
```

When the replaced program is executed, it outputs the following result:

```
Students' scores (unsorted)
ID:     0,   1,   2,   3
Value:  84,  84,  96,  84
------------------
Students' scores (sorted)
ID:     [      H      ]  2
Value:  84,  84,  84,  96
------------------
```

Answer group for H

a) `0,  1,  3,`

b) `0,  3,  1,`

c) `1,  0,  3,`

d) `1,  3,  0,`

e) `3,  0,  1,`

f) `3,  1,  0,`

Concerning questions **Q7** and **Q8**, **select one** of the two questions.

Q8. Read the following description of Java programs, and then answer Subquestions 1 and 2.

The program manages students, courses, and faculty in a university.
During a term, the program manages the course results of students. At the end of a term, it calculates the average grade of cumulative course results up to the end of the term using total grades and total credit hours of attended courses.

[Program Description]

The class `Course` (Program 1) represents a course.
The credit hours of each course are 3 hours. The capacity of each course is 35 students. A member of the faculty teaches each course.

- The method `addStudent` registers the student to the course and returns `true`. If the student has already registered for the course, or reached the course capacity, it returns `false` without changing the internal state of the object.
- The method `submitGrade` registers the grade of the student and returns `true`. If the student has not registered for the course, it returns `false`.

The class `Faculty` (Program 2) represents a member of the faculty such as a professor or an instructor. Each member of the faculty teaches up to 3 courses.

- The method `addCourse` registers the member of the faculty to the course and returns `true`. If he/she has already registered to 3 courses, or he/she or another member of the faculty has already registered to the course, it returns `false`.
- The method `submitGrade` registers the grade of the student and returns `true`. If the member of the faculty or the student has not registered to the course, it returns `false`.

The class `Student` (Program 3) represents a student. A student can register up to 6 courses.

- The constructor has 4 arguments: the student name, student ID, cumulative credit hours up to the last term, and the average of cumulative grades up to the last term.
- The method `addCourse` registers the student to the course and returns `true`. If the student has already registered to 6 courses, the course capacity has reached, or the student has already registered for the course, it returns `false`.
- The method `calculateCgpa` calculates the average of cumulative grades up to this term and returns the value.

The class `CourseManagement` (Program 4) manages Course, Faculty, and Student.

- The method `assignCourse` assigns the course to the member of the faculty and returns `true`. If the assignment fails, it returns `false` or throws an exception.

o The method `takeCourse` registers the student to the course and returns `true`. If the registration fails, it returns `false` or throws an exception.
o The method `submitGrade` registers the grade of the student who takes the course taught by the member of the faculty, and returns `true`. If the registration fails, it returns `false` or throws an exception.
o The method `cgpa` calculates the average of cumulative grades up to this term and returns the value.
o The method `main` is used for testing.

Explanation of interface `Map<K, V>`
o The method `put(k key, V value)`
Associates the specified value with the specified key in this map. It returns the previous value associated with the key, or `null` if there was no mapping for the key.
o The method `putIfAbsent(k key, V value)`
If the specified key is not already associated with a value, associates it with the given value and returns `null`. Otherwise, returns the current value.
o The method `replace(k key, V value)`
Replaces the entry for the specified key only if it is currently mapped to some value. Returns the previous value associated with the specified key, or `null` if there was no mapping for the key.

[Program 1]

```
import java.util.Map;
import java.util.HashMap;

public class Course {
   public static final int CREDIT_HOUR = 3;
   private static final int CAPACITY = 35;
   private final String title;
   private final String code;
   private String faculty;
   private final Map<    A    > gradeBook = new HashMap<>();

   Course(String title, String code) {
      this.title = title;
      this.code = code;
   }

   String getCode() { return code; }

   String getFaculty() { return faculty; }
```

```
    void setFaculty(String fid) { this.faculty = fid; }

    boolean addStudent(String sid) {
       if (gradeBook.size() < CAPACITY) {
          return gradeBook.[    B    ];
       }
       return false;
    }

    boolean submitGrade(String sid, double grade) {
       return gradeBook.[    C    ];
    }

    double getGrade(String sid) { return gradeBook.get(sid); }
 }
```

[Program 2]

```
 import java.util.Map;
 import java.util.HashMap;

 public class Faculty {
    private static final int MAX_COURSES = 3;
    private final String name;
    private final String id;
    private final Map<[    D    ]> courses = new HashMap<>();

    Faculty(String name, String id) {
       this.name = name;
       this.id = id;
    }

    String getName() { return name; }

    String getId() { return id; }

    boolean addCourse(Course course) {
       if (courses.size() < MAX_COURSES && [    E    ]) {
          courses.put(course.getCode(), course);
          course.setFaculty(id);
          return true;
       } else {
          return false;
       }
    }
```

```
    boolean submitGrade(String course, String student, double grade) {
        Course c = courses.get(course);
        return c == null ? false : c.submitGrade(student, grade);
    }
}
```

[Program 3]

```
import java.util.Map;
import java.util.HashMap;

public class Student {
    private static final int MAX_COURSES = 6;
    private final String name;
    private final String id;
    private final int creditHours;
    private final double cgpa;
    private final Map<    D    > courses = new HashMap<>();

    Student(String name, String id, int creditHours, double cgpa) {
        this.name = name;
        this.id = id;
        this.creditHours = creditHours;
        this.cgpa = cgpa;
    }

    boolean addCourse(Course course) {
        if (courses.size() < MAX_COURSES &&     F     ) {
            return courses.put(course.getCode(), course) == null;
        }
        return false;
    }

    double calculateCgpa() {
        double sum = creditHours * cgpa;
        double totalCreditHours;

        for (Course course : courses.values()) {
            sum += course.getGrade(id) * Course.CREDIT_HOUR;
        }
        totalCreditHours =
            creditHours + Course.CREDIT_HOUR * courses.size();
        return sum / totalCreditHours;
    }
}
```

[Program 4]

```
import java.util.Map;
import java.util.HashMap;

public class CourseManagement {
   public static void main(String[] args) {
      CourseManagement cm = new CourseManagement();
      String c1 = cm.newCourse("Programming Language I", "C105");
      String c2 = cm.newCourse("Programming Language II", "C205");
      String c3 = cm.newCourse("Data Structures", "C215");
      String s1 = cm.newStudent("Kamal Rahim", "17127", 15, 2.0);
      String s2 = cm.newStudent("Tamara Iqbal", "16200", 24, 3.0);
      String f1 = cm.newFaculty("Dr. Tasnim Khan", "F7980");
      String f2 = cm.newFaculty("Dr. Zia Kabir", "F4589");
      cm.assignCourse(f1, c1);
      cm.assignCourse(f2, c2);
      cm.assignCourse(f2, c3);
      cm.takeCourse(s1, c1);
      cm.takeCourse(s1, c2);
      cm.takeCourse(s2, c1);
      cm.takeCourse(s2, c2);
      cm.submitGrade(f1, c1, s1, 2.0);
      cm.submitGrade(f1, c1, s2, 3.0);
      cm.submitGrade(f2, c2, s2, 2.0);
      System.out.printf("Student: %s, CGPA: %3.2f%n",
                        s2, cm.cgpa(s2));
   }

   public Map<String, Student> students = new HashMap<>();
   public Map<String, Faculty> faculties = new HashMap<>();
   public Map<String, Course> courses = new HashMap<>();

   public String newStudent(String name, String id,
                            int creditHours, double cgpa) {
      students.put(id, new Student(name, id, creditHours, cgpa));
      return id;
   }

   public String newFaculty(String name, String id) {
      faculties.put(id, new Faculty(name, id));
      return id;
   }
```

```
    public String newCourse(String title, String code) {
       courses.put(code, new Course(title, code));
       return code;
    }

    public boolean assignCourse(String fid, String code) {
       return    G    ;
    }

    public boolean takeCourse(String sid, String code) {
       return students.get(sid).addCourse(courses.get(code));
    }

    public boolean submitGrade(String fid, String code, String sid,
                               double grade) {
       return faculties.get(fid).submitGrade(code, sid, grade);
    }

    public double cgpa(String sid) {
       return students.get(sid).calculateCgpa();
    }
}
```

Subquestion 1

From the answer groups below, select the correct answer to be inserted in each blank [ ] in the programs.

Answer group for A

a) `Double`
b) `double`
c) `String`
d) `String, Double`
e) `String, double`

Answer group for B

a) `put(sid, 0.0) != null`
b) `put(sid, 0.0) == null`
c) `putIfAbsent(sid, 0.0) != null`
d) `putIfAbsent(sid, 0.0) == null`
e) `replace(sid, 0.0) != null`
f) `replace(sid, 0.0) == null`

Answer group for C

a) `put(sid, grade) != null`
b) `put(sid, grade) == null`
c) `putIfAbsent(sid, grade) != null`
d) `putIfAbsent(sid, grade) == null`
e) `replace(sid, grade) != null`
f) `replace(sid, grade) == null`

Answer group for D

a) `Course`
b) `String`
c) `String, Course`
d) `String, String`

Answer group for E and F

a) `!course.addStudent(id)`
b) `course.addStudent(id)`
c) `course.getCode() != null`
d) `course.getCode() == null`
e) `course.getFaculty() != null`
f) `course.getFaculty() == null`
g) `courses.get(course.getCode()) != null`
h) `courses.get(course.getCode()) == null`

Answer group for G

a) `courses.get(code).setFaculty(faculties.get(fid))`
b) `courses.get(code).setFaculty(fid)`
c) `faculties.get(fid).addCourse(code)`
d) `faculties.get(fid).addCourse(courses.get(code))`

Subquestion 2

The class `CourseManagement` is executed.

From the answer group below, select the correct output of the class `CourseManagement`.

Answer group

a) `Student: 16200, CGPA: 2.50`
b) `Student: 16200, CGPA: 2.58`
c) `Student: 16200, CGPA: 2.81`
d) `Student: 16200, CGPA: 2.90`