



October 2024

Fundamental IT Engineer Examination (Subject B)

Questions must be answered in accordance with the following:

Question Nos.	Q1 – Q20
Question Selection	All questions are compulsory.
Examination Time	12:30 – 14:10 (100 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) **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]

Sample	<input type="radio"/> a	<input type="radio"/> b	<input type="radio"/> c	<input checked="" type="radio"/>	<input type="radio"/> e	<input type="radio"/> f	<input type="radio"/> g	<input type="radio"/> h	<input type="radio"/> i	<input type="radio"/> j
--------	-------------------------	-------------------------	-------------------------	----------------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------

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

Pseudo programming language notations

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

[Pseudo programming language notations]

Notation	Description
○ <i>procedure</i> (<i>type</i> : <i>arg1</i> , ...)	Declares a <i>procedure</i> and its argument(s) <i>arg1</i> ,
○ <i>ret-type</i> : <i>function</i> (<i>type</i> : <i>arg1</i> , ...)	Declares a <i>function</i> , its argument(s) <i>arg1</i> , ... , and type of return value <i>ret-type</i> .
<i>type</i> : <i>var1</i> , ... <i>type</i> []: <i>array1</i> , ...	Declares variables <i>var1</i> , ... and arrays <i>array1</i> , ... by data <i>type</i> such as integer, real, and string.
<i>/* comment */</i>	Describes a comment between <i>/*</i> and <i>*/</i> .
<i>// comment</i>	Describes a comment after <i>//</i> till end of line.
<i>variable</i> ← <i>expression</i>	Assigns the value of the <i>expression</i> to the <i>variable</i> .
<i>procedure</i> (<i>arg1</i> , ...)	Calls a <i>procedure</i> by passing arguments <i>arg1</i> ,
<i>function</i> (<i>arg1</i> , ...)	Calls a <i>function</i> by passing arguments <i>arg1</i> , ... , and receiving the return value.
output <i>arg1</i> , ...	Outputs values of <i>arg1</i> , ... to a printing device.
return <i>ret-val</i>	Finishes a function by passing back a return value <i>ret-val</i> .
<pre> if (<i>condition-i</i>) } *1 <i>process-i</i> elseif (<i>condition-ei</i>) } *2 <i>process-ei</i> else } *3 <i>process-e</i> endif </pre>	<p>Indicates the selection process.</p> <p>*1 If <i>condition-i</i> is true, then execute <i>process-i</i>. Otherwise, proceed to the next elseif or else.</p> <p>*2 If <i>condition-ei</i> is true, then execute <i>process-ei</i>. Otherwise, proceed to the next elseif or else.</p> <p>*3 If all conditions are false, execute <i>process-e</i>. Note: *2 and *3 can be omitted. *2 may exist twice or more.</p>
<pre> for (<i>sequence</i>) <i>process</i> endfor </pre>	<p>Indicates the “for” iteration process.</p> <p>In the order specified in the <i>sequence</i>, execute the <i>process</i> repeatedly.</p>
<pre> while (<i>condition</i>) <i>process</i> endwhile </pre>	<p>Indicates the “while” iteration process.</p> <p>While the <i>condition</i> is true, execute the <i>process</i> repeatedly.</p>
<pre> do <i>process</i> while (<i>condition</i>) </pre>	<p>Indicates the “do - while” iteration process.</p> <p>Execute the <i>process</i> once, and then while the <i>condition</i> is true, execute the <i>process</i> repeatedly.</p>

Pseudo programming language notations (continued)

[Operators and their precedence]

Type of operator	Operators	Precedence	Note
Expression	(), . ⁽¹⁾	<div style="text-align: center;"> High ↑ ↓ Low </div>	⁽¹⁾ accessing member or method
Unary operator	+, -, not ⁽²⁾		⁽²⁾ logical negation
Binary operator	x, ÷, mod ⁽³⁾		⁽³⁾ remainder
	+, -		
	>, <, ≥, ≤, =, ≠		
	and ⁽⁴⁾		⁽⁴⁾ logical product
	or ⁽⁵⁾		⁽⁵⁾ logical sum

[Boolean-type constants]

true, false

[Array reference]

	1-dimensional array	2-dimensional array	Array of arrays																													
Array declaration	<i>type</i> []: <i>name</i> ...	<i>type</i> [,]: <i>name</i> ...	<i>type</i> [][]: <i>name</i> ...																													
Example	<div>integer []: a1</div> <div><div><div>1</div><div>2</div><div>3</div><div>4</div><div>5</div></div><table><tr><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td></tr></table></div>	1	3	5	7	9	<div>integer [,]: a2</div> <div><div><div>1</div><div>2</div><div>3</div></div><table><tr><td>1</td><td>11</td><td>12</td><td>13</td></tr><tr><td>2</td><td>14</td><td>15</td><td>16</td></tr><tr><td>3</td><td>17</td><td>18</td><td>19</td></tr></table></div>	1	11	12	13	2	14	15	16	3	17	18	19	<div>integer [] []: aa</div> <div><div><div>1</div><div>2</div><div>3</div></div><table><tr><td>1</td><td>21</td><td>22</td><td></td></tr><tr><td>2</td><td>23</td><td>24</td><td>25</td></tr><tr><td>3</td><td>26</td><td></td><td></td></tr></table></div>	1	21	22		2	23	24	25	3	26		
1	3	5	7	9																												
1	11	12	13																													
2	14	15	16																													
3	17	18	19																													
1	21	22																														
2	23	24	25																													
3	26																															
Data reference	Data 7 is referred to by a1[4]	Data 16 is referred to by a2[2, 3]	Data 25 is referred to by aa[2][3]																													
Notation of array contents	{1, 3, 5, 7, 9}	{{11, 12, 13}, {14, 15, 16}, {17, 18, 19}}	{{21, 22}, {23, 24, 25}, {26}}																													

Note: The indexes of example arrays start at 1.

[undefined state]

`undefined` is a state in which no value is set to a variable (or an element of an array).
By setting `undefined` to a variable, the variable is transformed into `undefined` state.

Q1. From the answer group below, select the correct combination of answers to be inserted into and in the program. Here, the array indexes start at 1.

Bicycle ridership in a city is studied. Examination of several years of data revealed that 30% of the people who regularly ride bicycles in a given year do not regularly ride bicycles in the subsequent year. Additionally, 2% of the people who do not regularly ride bicycles in that year begin to ride bicycles regularly in the subsequent year. If 5,000 people ride bicycles and 100,000 people do not ride bicycles in a given year, then the following program calculates the number of cyclists in the subsequent year (cycling) and that of people who do not ride bicycles in the subsequent year (noncycling):

[Program]

```

real: pc ← 0.3
real: pb ← 0.02
real: pa ← (1 - pc)
real: pd ← (1 - pb)
integer []: N ← {5000, 100000}
real [,]: P ← {{pa, pb}, {pc, pd}}
real: cycling, noncycling
cycling ← 
noncycling ← 
output cycling, noncycling

```

Answer group

	A	B
a)	$P[1,1] \times N[1] + P[1,2] \times N[2]$	$P[2,1] \times N[1] + P[2,2] \times N[2]$
b)	$P[1,1] \times N[1] + P[1,2] \times N[2]$	$P[2,2] \times N[1] + P[2,1] \times N[2]$
c)	$P[1,2] \times N[1] + P[1,1] \times N[2]$	$P[2,2] \times N[1] + P[2,1] \times N[2]$
d)	$P[1,2] \times N[1] + P[2,2] \times N[2]$	$P[2,1] \times N[1] + P[1,2] \times N[2]$
e)	$P[2,1] \times N[1] + P[1,2] \times N[2]$	$P[1,1] \times N[1] + P[2,2] \times N[2]$
f)	$P[2,1] \times N[1] + P[2,2] \times N[2]$	$P[1,1] \times N[1] + P[1,2] \times N[2]$

Q2. From the answer group below, select the correct combination of answers to be inserted into A and B in the program.

The function coupon receives the argument prod_id (a positive integer value) of product ID, and pur_prod (a positive integer value) of the number of purchased products by a customer. The function returns the number of coupons.

For every purchase of three products of the product ID of which the last digit is three, the customer receives one coupon. Otherwise, they receive no coupon.

[Program]

```

integer: coupon(integer: prod_id, integer: pur_prod)
integer: num_coupon ← 0
if ( A )
    num_coupon ← B
endif
return num_coupon

```

Answer group

	A	B
a)	integer part of $(\text{prod_id} \div 10) = 3$	integer part of $(\text{pur_prod} \div 3)$
b)	integer part of $(\text{prod_id} \div 10) = 3$	integer part of $(\text{pur_prod} \div 3) + 1$
c)	integer part of $(\text{prod_id} \div 10) = 3$	$\text{pur_prod} \bmod 3$
d)	$\text{prod_id} = 10 \bmod 3$	integer part of $(\text{pur_prod} \div 3)$
e)	$\text{prod_id} = 10 \bmod 3$	integer part of $(\text{pur_prod} \div 3) + 1$
f)	$\text{prod_id} = 10 \bmod 3$	$\text{pur_prod} \bmod 3$
g)	$\text{prod_id} \bmod 10 = 3$	integer part of $(\text{pur_prod} \div 3)$
h)	$\text{prod_id} \bmod 10 = 3$	integer part of $(\text{pur_prod} \div 3) + 1$
i)	$\text{prod_id} \bmod 10 = 3$	$\text{pur_prod} \bmod 3$

Q3. From the answer group below, select the correct combination of answers to be inserted into and in the program. Here, the array index starts at 1.

The program determines the second-largest element of an integer-type array and outputs its value. For instance, the second largest element of the array {2, 6, 9, 1, 7, 5} is 7. Here, we assume that the array has two or more elements and that no duplicate elements are present in the array.

[Program]

```
integer []: array ← {2, 6, 9, 1, 7, 5}
integer: i
integer: max1 ← -∞
integer: max2 ← -∞
for (increase i from 1
    to the number of elements of array by 1)
    if ()
        max2 ← max1
        max1 ← array[i]
    elseif (array[i] > max2)
        
    endif
endfor
output max2
```

Answer group

	A	B
a)	array[i] < max1	max1 ← array[i]
b)	array[i] < max1	max2 ← array[i]
c)	array[i] < max2	max1 ← array[i]
d)	array[i] < max2	max2 ← array[i]
e)	array[i] > max1	max1 ← array[i]
f)	array[i] > max1	max2 ← array[i]
g)	array[i] > max2	max1 ← array[i]
h)	array[i] > max2	max2 ← array[i]

Q4. From the answer group below, select the correct combination of answers to be inserted into and in the program.

The procedure `primeFactors` outputs the prime factors for the input values as its argument. The first few prime numbers are 2, 3, 5, 7, 11, and 13. For instance, if the input integer is 12, the output is “2×2×3”. If the input integer is 78, the output is “2×3×13”. The input integer must be greater than 1.

[Program]

```

O primeFactors(integer: num)
  integer: i

  i ← 2
  do
    if ()
      num ← integer part of (num ÷ i)
      output i
      if ()
        output "x"
      endif
    else
      i ← i + 1
    endif
  while ()

```

Answer group

	A	B
a)	num < 1	num mod i ≠ 0
b)	num > 1	num mod i = 0
c)	num mod i = 0	num > 1
d)	num mod i ≠ 0	num < 1

Q5. From the answer group below, select the correct combination of answers to be inserted into

A

 and

B

 in the program. Here, the array index starts at 1.

Function `calcGeoMean` receives the array `dataArray` (the number of elements ≥ 1) as an argument and returns the geometric mean of values in `dataArray` as the return value. For the numbers a_1, a_2, \dots, a_n , the geometric mean is $\sqrt[n]{a_1 \times a_2 \times \dots \times a_n}$. Here, the `pow(x, y)` function returns a real value by raising x to the power of y . In the program, division is performed in data type `real`.

[Program]

```

O real: calcGeoMean(real []: dataArray)
  real: product, geomean
  integer: n ← the number of elements in dataArray
  integer: i

  product ← 1
  for (increase i from 1 to n by 1)
    product ← product × dataArray[i]
  endfor
  geomean ← pow(

|   |
|---|
| A |
|---|

, 

|   |
|---|
| B |
|---|

)
  return geomean

```

Answer group

	A	B
a)	$1 \div n$	product
b)	product	$1 \div n$
c)	product	n
d)	n	product

Q6. From the answer group below, select the correct combination of answers to be inserted into through in the program.

The standard form of a quadratic equation is as follows:

$$ax^2 + bx + c = 0, \text{ where } a, b \text{ and } c \text{ are real numbers and } a \neq 0.$$

Here, the term $b^2 - 4ac$ is known as the discriminant of a quadratic equation. It indicates the nature of the roots. The formula for solving a quadratic equation is shown in the figure. If the discriminant value is positive, there are two solutions; if it is zero, there is one solution. Here, we assume that the discriminant value is non-negative.

If the discriminant > 0	$root1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$ $root2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$
If the discriminant = 0	$root1 = root2 = \frac{-b}{2a}$

Figure The formula for solving a quadratic equation

The procedure findRoots receives three real number arguments a, b, and c as coefficients and outputs the value of the root(s) of the quadratic equation. Function sqrt(discriminant) returns the principal square root value of the parameter discriminant.

[Program]

```

○ findRoots(real: a, real: b, real: c)
  real: discriminant, root1, root2
  discriminant ← 
  if ()
    root1 ← (-b + sqrt(discriminant)) ÷ (2 × a)
    root2 ← (-b - sqrt(discriminant)) ÷ (2 × a)
    output "root1 = ", root1, " and root2 = ", root2
  elseif ()
    root1 ← -b ÷ (2 × a)
    output "root1 = root2 = ", root1
  endif

```

Answer group

	A	B	C
a)	$b \times 2 - 4 \times a \times c$	discriminant < 0	discriminant $= 0$
b)	$b \times 2 - 4 \times a \times c$	discriminant $= 0$	discriminant < 0
c)	$b \times 2 - 4 \times a \times c$	discriminant $= 0$	discriminant > 0
d)	$b \times 2 - 4 \times a \times c$	discriminant > 0	discriminant $= 0$
e)	$b \times b - 4 \times a \times c$	discriminant < 0	discriminant $= 0$
f)	$b \times b - 4 \times a \times c$	discriminant $= 0$	discriminant < 0
g)	$b \times b - 4 \times a \times c$	discriminant $= 0$	discriminant > 0
h)	$b \times b - 4 \times a \times c$	discriminant > 0	discriminant $= 0$

Q7. From the answer group below, select the correct combination of answers to be inserted into and in the program.

Given two strings, `str1` and `str2`, the task is to determine the length of the longest common subsequence (LCS), that is, the length of the longest subsequence present in both strings.

For instance, if `str1` is "ABXDZ" and `str2` is "ABCD", the LCS between `str1` and `str2` will be "ABD" and the length of the LCS will be 3.

The function `lcs(string: str1, string: str2, integer: m, integer: n)` takes two strings `str1` and `str2` and two integers `m` and `n` as arguments. `str1` and `str2` are the strings on which the length of the LCS is calculated. `m` and `n` are indexes addressing the target characters in `str1` and `str2`, respectively. On the first call, `m` and `n` represent the lengths of `str1` and `str2`, respectively. The function returns an integer containing the value of the length of the LCS of `str1` and `str2`. For instance, the function may be called as `lcs("ABXDZ", "ABCD", 5, 4)`.

Another function `max(integer: a, integer: b)` is also used. The function takes two integers `a` and `b` as arguments. It compares these two integers and returns the integer value of whichever is the maximum between `a` and `b`.

[Program]

```
○ integer: max(integer: a, integer: b)
    if (a > b)
        return a
    else
        return b
    endif

○ integer: lcs(string: str1, string: str2, integer: m, integer: n)
    if (m = 0 or n = 0)
        return 0
    endif
    if (the m-th character of string str1 = the n-th character of string str2)
        return 1 + lcs(str1, str2, m - 1, n - 1)
    else
        return max(lcs(str1, str2, m, ),
                    lcs(str1, str2, m - 1, ))
    endif
```

Answer group

	A	B
a)	n	$n - 1$
b)	n	$n + 1$
c)	$n - 1$	n
d)	$n - 1$	$n + 1$
e)	$n + 1$	n
f)	$n + 1$	$n - 1$

Q8. From the answer group below, select the correct answer to be inserted into in the description.

This program performs operations on a priority queue.

A priority queue is a queue where the handled elements have a priority assigned to them, and the elements are extracted with the order of the highest priority first. The class `PrioQueue` represents a priority queue. The Figure shows an explanation of the class `PrioQueue`. Here, the priority is the integer value 1, 2, or 3, and the smaller the value the higher the priority.

When the procedure `prioSched` is called, the order of the output is .

Constructor	Description
<code>PrioQueue()</code>	Creates an empty priority queue.

Method	Type of return value	Description
<code>enqueue(string: s, integer: prio)</code>	None	Adds string <code>s</code> as an element to a priority queue with the priority <code>prio</code> .
<code>dequeue()</code>	string	Extracts the element with the highest priority in a priority queue and returns it. If multiple elements with the highest priority exist, it extracts the element that was added first and returns it.
<code>size()</code>	integer	Returns the number of elements that are stored in a priority queue.

Figure Explanation of the class `PrioQueue`

[Program]

```
○ prioSched()  
  PrioQueue: prioQueue ← PrioQueue()  
  prioQueue.enqueue("E", 3)  
  prioQueue.enqueue("F", 2)  
  prioQueue.enqueue("G", 1)  
  prioQueue.enqueue("H", 1)  
  prioQueue.dequeue() /* The return value is ignored */  
  prioQueue.dequeue() /* The return value is ignored */  
  prioQueue.enqueue("I", 1)  
  prioQueue.enqueue("J", 1)  
  prioQueue.dequeue() /* The return value is ignored */  
  prioQueue.enqueue("K", 2)  
  prioQueue.enqueue("L", 3)  
  prioQueue.enqueue("M", 1)
```

```
while (prioQueue.size() is not equal to 0)
  output prioQueue.dequeue()
endwhile
```

Answer group

- a) "M", "I", "K", "F", "L", "E"
- b) "M", "L", "K", "I", "F", "E"
- c) "J", "M", "F", "K", "E", "L"
- d) "E", "L", "F", "K", "I", "M"

Q9. From the answer group below, select the correct answer to be inserted into in the description. Here, the array indexes start at 1.

The procedure `traverse` traces through a vertex of the graph shown in the Figure, and outputs all vertex numbers in the graph. The vertex number of the graph is specified with the argument `k`. The global variable `n` indicates the number of vertices in the graph. The global array `graph` represents the graph in the figure. Each element `graph[i][j]` is equal to 1 if an edge exists between vertices `i` and `j`, and it is equal to 0 otherwise. The global array `visited` stores boolean values, where `visited[i]` indicates whether vertex `i` of the graph has been visited during the procedure.

When the procedure is called as `traverse(1)`, the output is in the order .

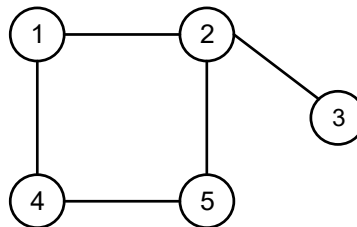


Figure Structure of the graph

[Program]

```

global: integer: n ← 5
global: integer [] []: graph ← {{0, 1, 0, 1, 0}, {1, 0, 1, 0, 1},
                                {0, 1, 0, 0, 0}, {1, 0, 0, 0, 1},
                                {0, 1, 0, 1, 0}}
global: boolean []: visited ← {false, false, false, false, false}

```

O `traverse(integer: k)`

```

integer: i
visited[k] ← true
output k

```

```

for (increase i from 1 to n by 1)
  if (graph[k][i] = 1 and visited[i] = false)
    traverse(i)
  endif
endfor

```

Answer group

- | | |
|------------------|------------------|
| a) 1, 2, 3, 4, 5 | b) 1, 2, 3, 5, 4 |
| c) 1, 2, 4, 3, 5 | d) 1, 2, 4, 5, 3 |

Q10. From the answer group below, select the correct answer to be inserted into in the program.

The procedure `addNode` adds a node to a singly-linked list at the position specified by the argument `pos`. The argument `pos` is a positive integer that is equal to or less than the (number of nodes + 1) in the list. The position at the top of the list is 1.

The class `ListNode` represents a node in a singly-linked list. The table summarizes an explanation of the member variables of the class `ListNode`. `ListNode`-type variables store references to instances of the class `ListNode`. A reference to the first node in the list is pre-stored in the global variable `listHead`.

Table Explanation of the member variables of the class `ListNode`

Member variable	Type	Description
<code>val</code>	character	The value of a node.
<code>next</code>	<code>ListNode</code>	A reference for the next node. If no next node exists, the status is undefined.

[Program]

```
global: ListNode: listHead // stores the first node in the list

O addNode(integer: pos, character: val)
  ListNode: prev, newNode
  integer: i
  newNode ← ListNode()
  newNode.val ← val
  if (pos is equal to 1)
    newNode.next ← listHead
    listHead ← newNode
  else
    prev ← listHead
    /* if pos is equal to 2, the following iteration process is not executed */
    for (increase i from 2 to pos - 1 by 1)
      prev ← prev.next
    endfor
    newNode.next ← prev.next
     ← newNode
  endif
```

Answer group

- | | | |
|--------------------------|-------------------------------|------------------------------------|
| a) <code>listHead</code> | b) <code>listHead.next</code> | c) <code>listHead.next.next</code> |
| d) <code>prev</code> | e) <code>prev.next</code> | f) <code>prev.next.next</code> |

Q11. From the answer group below, select the correct combination of answers to be inserted into

A

 and

B

 in the program. Here, the array indexes start at 1.

The procedure sort sorts an integer array containing certain number (≥ 2) of elements in ascending order.

[Program]

```

O sort(integer []: arg)
  Integer []: A ← arg
  integer: i, k
  for (increase i from 2 to the number of elements in A by 1)
    k ← i
    while (k > 1)
      if (A[k - 1] ≤ A[k])
        exit the while block
      endif
      

|   |
|---|
| A |
|---|



|   |
|---|
| B |
|---|


    endwhile
  endfor
  output A

```

Answer group

	A	B
a)	$A[i] \leftarrow A[i - 1]$	$k \leftarrow k + 1$
b)	$A[k] \leftarrow A[k - 1]$	swap k and i
c)	$A[k - 1] \leftarrow A[k]$	$k \leftarrow k + 1$
d)	swap A[i] and A[i - 1]	$k \leftarrow k - 1$
e)	swap A[i] and A[k]	swap k and i
f)	swap A[k] and A[k - 1]	$k \leftarrow k - 1$

Q12. From the answer group below, select the correct combination of answers to be inserted into and in the program. Here, the array index starts at 1.

The function `isPalindrome` determines whether the character array `s` given as argument is a palindrome. Palindromes are words when read backward will still be the same such as “noon” and “madam.”

If character array `s` is a palindrome, it returns the value `true` and if not, the function returns `false`.

The table lists examples of `s` given to the function `isPalindrome` and the return values. In the program, areas outside of the arrays must not be referenced.

Table Examples of `s` given to the function `isPalindrome` and the return values

s	Return value
{"n", "o", "o", "n"}	true
{"n", "i", "g", "h", "t"}	false
{"m", "a", "d", "a", "m"}	true
{"s", "i", "r"}	false

[Program]

```

O boolean: isPalindrome(character []: s)
  integer: left ← 1
  integer: right ← the number of elements in s
  boolean: ok ← true
  while (left < right)
    if ()
      left ← left + 1
      
    else
      ok ← false
      break
    endif
  endwhile
  return ok

```

Answer group

	A	B
a)	$s[\text{left}] = s[\text{right}]$	$\text{right} \leftarrow \text{right} + 1$
b)	$s[\text{left}] = s[\text{right}]$	$\text{right} \leftarrow \text{right} - 1$
c)	$s[\text{left}] = s[\text{right}]$	$\text{right} \leftarrow \text{right} + \text{left}$
d)	$s[\text{left}] = s[\text{right}]$	$\text{right} \leftarrow \text{right} - \text{left}$
e)	$s[\text{left}] \neq s[\text{right}]$	$\text{right} \leftarrow \text{right} + 1$
f)	$s[\text{left}] \neq s[\text{right}]$	$\text{right} \leftarrow \text{right} - 1$
g)	$s[\text{left}] \neq s[\text{right}]$	$\text{right} \leftarrow \text{right} + \text{left}$
h)	$s[\text{left}] \neq s[\text{right}]$	$\text{right} \leftarrow \text{right} - \text{left}$

Q13. From the answer group below, select the correct combination of answers to be inserted into and in the program. Here, the array indexes start at 1.

Hashing in data structures is a fundamental concept used for efficient data retrieval and storage mechanisms. A program storing a key-data pairs in an array by transforming keys into array indexes using a hash function exists. A collision occurs when two keys hash to the same index in the array representing the hash table. A common method handling collisions, the probing mechanism (checking for an available element), is used in the function. The procedure `insertData` inserts a pair of key and data, if the element in the array is `{undefined}`. Assumptions are made that no data with the same key is stored, and that at least one element in the array `hashTable` is `{undefined}` when the procedure `insertData` is called. The function `hashFunction` takes a key as input and returns a hash value.

[Program]

```
global: integer [][: hashTable ← {Elements comprising 1000 {undefined}}
global: integer: size ← 1000
○ integer: hashFunction(integer: key)
    return (key mod size) + 1

○ insertData(integer: key, integer: data)
    integer: index
    index ← hashFunction(key)
    while ()
        if (index = size)
            
        else
            index ← index + 1
        endif
    endwhile
    hashTable[index] ← {key, data}
```

Answer group

	A	B
a)	hashTable[index][1] \neq key	index \leftarrow 1
b)	hashTable[index][1] \neq key	index \leftarrow index - 1
c)	hashTable[index][1] = key	index \leftarrow 1
d)	hashTable[index][1] = key	index \leftarrow index - 1
e)	hashTable[index][1] \neq undefined	index \leftarrow 1
f)	hashTable[index][1] \neq undefined	index \leftarrow index - 1
g)	hashTable[index][1] = undefined	index \leftarrow 1
h)	hashTable[index][1] = undefined	index \leftarrow index - 1

Q14. From the answer group below, select the correct answer to be inserted into in the description. Here, the array indexes start at 1.

The function `summarize` receives the array `sortedData` sorted in ascending order and returns five values that characterize the array. The array `sortedData` must have at least one element. The `summarize` calls the `findRank` with two arguments, `sortedData` and `q`. When the function `summarize` is called as `summarize({0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1})`, the return value is .

[Program]

```
○ real: findRank(real []: sortedData, real: q)
  integer: j
  j ← floor(q × (the number of elements in sortedData - 1))
  // floor returns the closest integer less than or equal to a given
  // number, e.g. floor(7.75) returns 7.
  return sortedData[j + 1]

○ real []: summarize(real []: sortedData)
  real []: rankData ← {} /* array with 0 elements */
  real []: q ← {0, 0.25, 0.5, 0.75, 1}
  integer: i
  for (increase i from 1 to the number of elements in q by 1)
    add the return value of findRank(sortedData, q[i]) to the end of rankData
  endfor
  return rankData
```

Answer group

- a) {0.1, 0.2, 0.3, 0.4, 0.5}
- b) {0.1, 0.2, 0.3, 0.4, 0.7}
- c) {0.1, 0.2, 0.4, 0.6, 0.9}
- d) {0.1, 0.2, 0.4, 0.7, 0.9}
- e) {0.1, 0.3, 0.5, 0.7, 1}
- f) {0.1, 0.3, 0.5, 0.8, 1}
- g) {0.1, 0.3, 0.6, 0.9, 1}
- h) {0.1, 0.3, 0.7, 0.9, 1}

Q15. From the answer group below, select the correct combination of answers to be inserted into and in the program. Here, the array indexes start at 1.

Suppose the weather in a city, from day to day, follows a Markov process. This implies the next day's weather depends only on today's weather and not on those of the previous days. If it rains today in the city, a probability 0.3 exists that it will rain again tomorrow. Similarly, the probability that tomorrow's weather will be sunny is 0.4 and that it will be cloudy is 0.3. If today's weather is sunny, the next day will be rainy, sunny, or cloudy with probabilities 0.2, 0.7, and 0.1, respectively. Additionally, if today's weather is cloudy, the next day will be rainy, sunny, or cloudy with probabilities 0.25, 0.5, and 0.25, respectively. Let us say that the weather is in state 1 if it is rainy, in state 2 if it is sunny, and in state 3 if it is cloudy. Then, the above is a three-state Markov chain whose transition probabilities are given by the following:

$$\begin{array}{c} \text{to} \\ \begin{array}{ccc} & 1 & 2 & 3 \\ \text{from} \begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \left(\begin{array}{ccc} 0.3 & 0.4 & 0.3 \\ 0.2 & 0.7 & 0.1 \\ 0.25 & 0.5 & 0.25 \end{array} \right) \end{array}$$

Using this transition matrix, the probability distribution of the weather after several days can be estimated.

The program calculates the probability distribution of the weather of the city after three days from the present day and outputs a 3×3 square matrix that shows this probability distribution of the weather. For instance, if the weather on the present day is sunny (state 2), the probability that the weather three days after will be rainy (state 1) is the value in the second row and the first column of the outputted matrix.

[Program]

```
real [,]: pmat ← {{0.3, 0.4, 0.3}, {0.2, 0.7, 0.1}, {0.25, 0.5, 0.25}}
real [,]: dmat ← pmat
real [,]: smat
integer: i, j, k, m
for (increase m from 1 to  by 1)
  smat ← {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
  for (increase i from 1 to 3 by 1)
    for (increase j from 1 to 3 by 1)
      for (increase k from 1 to 3 by 1)
        smat[i, j] ← smat[i, j] + 
```

```

        endfor
    endfor
endfor
dmat ← smat
endfor
output dmat

```

Answer group

	A	B
a)	2	$\text{dmat}[i, k] \times \text{pmat}[k, j]$
b)	2	$\text{dmat}[k, i] \times \text{pmat}[j, k]$
c)	3	$\text{dmat}[i, k] \times \text{pmat}[k, j]$
d)	3	$\text{dmat}[k, i] \times \text{pmat}[j, k]$
e)	4	$\text{dmat}[i, k] \times \text{pmat}[k, j]$
f)	4	$\text{dmat}[k, i] \times \text{pmat}[j, k]$

Q16. From the answer group below, select the correct combination of answers to be inserted into through in the program.

Function `hammingDistance` receives two strings as argument and returns Hamming distance as the return value. Hamming distance is a metric used to measure the difference between two strings of equal length. It counts the number of positions at which the corresponding characters are different. For instance, the return value is 2 when the function `hammingDistance("101010", "111000")` is called, because positions 2 and 5 have different characters.

It can also be extended to strings of different lengths by considering the unmatched characters as differences. For instance, the return value is 5 when the function `hammingDistance("101010", "111000111")` is called, because of two differences in first six characters and addition of three extra unmatched characters.

[Program]

```
○ integer: hammingDistance(string: s1, string: s2)
  integer: distance, length1, length2, minLength, remainingLength
  length1 ← length of s1
  length2 ← length of s2
  distance ← 0
  minLength ← length1
  if ()
    minLength ← length2
  endif
  for (increase i from 1 to minLength by 1)
    if (the i-th character of string s1  the i-th character of string s2)
      distance ← distance + 1
    endif
  endfor
  if (length1 > length2)
    remainingLength = length1 - length2
  else
    remainingLength = length2 - length1
  endif
  distance ← 
  return distance
```

Answer group

	A	B	C
a)	$\text{length2} < \text{length1}$	is equal to	remainingLength
b)	$\text{length2} < \text{length1}$	is equal to	distance + remainingLength
c)	$\text{length2} < \text{length1}$	is not equal to	distance + minLength
d)	$\text{length2} < \text{length1}$	is not equal to	distance + remainingLength
e)	$\text{length2} > \text{length1}$	is equal to	remainingLength
f)	$\text{length2} > \text{length1}$	is equal to	distance + remainingLength
g)	$\text{length2} > \text{length1}$	is not equal to	distance + minLength
h)	$\text{length2} > \text{length1}$	is not equal to	distance + remainingLength

Q17. From the answer group below, select the most appropriate combination of answers to be inserted into and in the description.

Company A is an e-commerce company specializing in sporting goods. One morning, employee X of the company's accounting department starts up his PC to begin his work. When he logged into his PC, he saw an unfamiliar screen with the following message:

*** Attention ***

Your files have been encrypted.

All of your important files have been encrypted. To restore these files, please follow these steps:

1. transfer \$500 worth of cryptocurrency to the currency address provided:

Address: 1ABCD.... EFGH

2. As soon as the transfer is confirmed, we will send you the key to decrypt the files.

3. If the transfer is not confirmed within 48 h, the decryption key will be deleted and the files will be lost forever.

Contact: help@ransomware.example.com

The accounting department immediately requested the security team in company A to help resolve the situation. The security team analyzed all the PCs in the accounting department, including that of employee X, and noted the following:

1. A total of tow compromised PCs were found. One PC was used by Employee X and the other PC was shared by multiple employees.
2. IDS/IPS was not installed in the accounting department LAN, delaying detection of ransomware.
3. The shared PC was backed up daily, and important data could be restored from the backup at the time when the PC was not compromised by the ransomware.
4. Employee X's PC was not backed up, and important data could not be restored.
5. The ransomware was saved on Employee X's PC as an attachment file of an email sent by a company pretending to be a business partner. The email envelope and content were sophisticatedly disguised and difficult to detect through enhanced spam filters.

6. Employee X inadvertently executed the attached file and allowed the ransomware to invade his PC.
7. After compromising Employee X's PC, the ransomware attempted to compromise other PCs in the LAN by taking advantage of vulnerabilities in the target OS, but succeeded only in compromising the shared PC, which had neglected to apply OS security patches.
8. On both PCs, the anti-virus software versions and virus definition files were checked for updates on a daily basis and maintained up-to-date.
9. Both PCs had OS storage encryption feature enabled, but it was not effective against the ransomware attack in this incident.

The security team and the accounting department decided to take the following approaches and specific measures as a plan to strengthen the measures against future ransomware attacks.

Table 1. Approaches and Specific Measures (excerpts)

Approaches	Specific Measures
Prevention	User education and training <div style="border: 1px solid black; padding: 2px; display: inline-block;">A</div>
Detection	Network monitoring with IDS/IPS
Data Protection	<div style="border: 1px solid black; padding: 2px; display: inline-block;">B</div>

Answer group

	A	B
a)	SPAM filter enhancement	Data backup
b)	SPAM filter enhancement	Data encryption
c)	OS updates	Data backup
d)	OS updates	Data encryption
e)	Virus definition updates	Data backup
f)	Virus definition updates	Data encryption

Q18. From the answer group below, select the correct answer to be inserted into in the description.

Company B runs several web services. Mr. Y is a software engineer in the company. He was directed to develop a feature that prevents web service users from setting weak passwords. Mr. Y investigated and found that “Password Entropy” is the important factor for the password strength.

Password entropy measures the strength of a password based on the difficulty of cracking the password through guessing or a brute-force attack. The entropy of a password is typically based on the type of characters used—lowercase letters, uppercase letters, numerical digits, or special characters—and the length of the password or total number of characters.

Password entropy formula:

You can measure the entropy of a password in bits using the following formula:

$$E = [L \times \text{Log}_2 (R)]$$

In this formula:

- E denotes the password entropy
- R denotes the possible range of character types in the password
- L denotes the number of characters in the password (its length)
- Log_2 answers the question, "to what power 2 must be raised to equal this number"
- $[X]$ denotes the greatest integer N such that $N \leq X$

The following totals show the breakdown for each character set as they appear on a typical American QWERTY keyboard:

Lowercase letters (a–z) = 26.

Uppercase letters (A–Z) = 26.

Numerical digits (0–9) = 10.

Special characters (!, @, #, \$, %, ^, etc.) = 32.

The password **vfssxfrb** has a pool of 26 characters (lowercase letters);

Changing the password to **JAAdMAde** would increase the pool to 52 characters (lowercase letters and uppercase letters);

Changing it further to **u7aHqsbt** would increase the pool to 62 characters (lowercase letters, uppercase letters along with numbers) and

Finally, **Re*ct7\$%** has a pool of $26 + 26 + 10 + 32 = 94$ characters (lowercase letters,

uppercase letters, numbers, and special characters).

In our example:

- For **vfssxfrb**, we have $R = 26$ and $L = 8$; thus,
 $E = \lfloor 8 \times \log_2(26) \rfloor \approx \lfloor 8 \times 4.700 \rfloor \approx \lfloor 37.60 \rfloor = 37$ bits
- For **JAAdMADE**, we have $R = 52$ and $L = 8$; thus,
 $E = \lfloor 8 \times \log_2(52) \rfloor \approx \lfloor 8 \times 5.700 \rfloor \approx \lfloor 45.60 \rfloor = 45$ bits
- For **u7aHqsbt**, we have $R = 62$ and $L = 8$; thus,
 $E = \lfloor 8 \times \log_2(62) \rfloor \approx \lfloor 8 \times 5.954 \rfloor \approx \lfloor 47.63 \rfloor = 47$ bits
- For **Re*ct7\$%**, we have $R = 94$ and $L = 8$; thus,
 $E = \lfloor 8 \times \log_2(94) \rfloor \approx \lfloor 8 \times 6.555 \rfloor \approx \lfloor 52.44 \rfloor = 52$ bits

Here, when the password is **A6GmyVyOC** (Using numerical digits, lowercase letters, and uppercase letters. $L = 9$), the password entropy is bits.

Answer group

- a) 42 b) 51 c) 52 d) 53 e) 58

Q19. From the answer group below, select the correct combination of answers to be inserted into and in the description.

Company X is an investment company. Recently, one of their database servers was attacked by an attacker.

Mr. L, the security team leader of company X, decided to ask Ms. A, a security analyst working for the same company, to analyze the logs. Mr. L will extract log entries related to database access from the log file of the attacked database server and send the extracted log entries to Ms. A.

When sending access log data from Mr. L to Ms. A, the following conditions were agreed upon in advance:

- Mr. L must encrypt the file such that only Ms. A can view it.
- Ms. A must confirm that the sender of the file is Mr. L.

They each have an RSA key pair and have already exchanged their public keys with each other. They sign, verify, encrypt, and decrypt using those keys and a previously shared cryptographic processing software.

This time, the log entries must be and then .

Answer group

	A	B
a)	signed using Mr. L's private key	encrypted using Mr. L's private key
b)	signed using Mr. L's private key	encrypted using Mr. L's public key
c)	signed using Mr. L's private key	encrypted using Ms. A's private key
d)	signed using Mr. L's private key	encrypted using Ms. A's public key
e)	signed using Mr. L's public key	encrypted using Mr. L's private key
f)	signed using Mr. L's public key	encrypted using Mr. L's public key
g)	signed using Mr. L's public key	encrypted using Ms. A's private key
h)	signed using Mr. L's public key	encrypted using Ms. A's public key

Q20. From the answer group below, select the most appropriate combination of answers to be inserted into and in the description.

Company Y, a trading company, is developing an e-commerce web application for selling products to customers directly. The web server hosting the web application is installed on-premise and the web application is to be developed in-house. The communications to the web server are allowed only with HTTPS and the server certificate is already signed by the third party and installed on the web server. Company Y has an IT department comprising three teams—Developer, Security, and Tech Support—and the requirements for the web application are specified as follows (excerpt):

- The web application is user-friendly and easy to navigate.
- The web application is cost-effective to develop and maintain.
- The web application can prevent malicious inputs into the system.
- The web application is scalable and can handle a large number of customers.
- The purchase orders and invoices are digitally signed to confirm integrity and authenticity.

During the initial project meeting, the members from each department discuss their respective tasks. The developer team requests the security team to derive specific tasks for the above requirements related to security. Ms. B, a member of the security team, suggests the following tasks:

- Set up a to provide the infrastructure.
- Install a in front of the web server.

Answer group

	A	B
a)	PKI environment	honeypot
b)	PKI environment	VPN gateway
c)	PKI environment	web application firewall
d)	SPF record	honeypot
e)	SPF record	VPN gateway
f)	SPF record	web application firewall

Company names and product names appearing in the test questions are trademarks or registered trademarks of their respective companies. Note that the ® and ™ symbols are not used within the text.