2021-DSE
ICT
PAPER 2D

Please stick the barcode label here.

HONG KONG EXAMINATIONS AND ASSESSMENT AUTHORITY

HONG KONG DIPLOMA OF SECONDARY EDUCATION EXAMINATION 2021

Candidate Number									

INFORMATION AND COMMUNICATION TECHNOLOGY

PAPER 2D

Software Development

Question-Answer Book

11:15 am – 12:45 pm (1 hour 30 minutes)
This paper must be answered in English

INSTRUCTIONS

(1) After the announcement of the start of the examination, you should first write your Candidate Number in the space provided on Page 1 and stick barcode labels in the spaces provided on Pages 1, 3, 5 and 7.

(2) Answer **THREE** out of four questions. Write your answers in the spaces provided in this Question-Answer book. Do not write in the margins. Answers written in the margins will not be marked.

(3) Supplementary answer sheets will be supplied on request. Write your candidate number, mark the question number box and stick a barcode label on each sheet, and fasten them with string **INSIDE** this book.

(4) No extra time will be given to candidates for sticking on the barcode labels or filling in the question number boxes after the 'Time is up' announcement.

* A 2 0 0 E 0 2 D *

Answer THREE questions only.

1. In a train station, the departure times of trains in one day are stored in an array `P` of size `n` ordered by time. Peter writes a program `F1(s,t)` to find the number of trains departed from time `s` to time `t` inclusive. He assumes that there are always trains departed at time `s` and time `t`. For example, according to the content of `P` below, `F1(12:20, 18:00)` returns 3.

i	1	2	3	4	5	6	7
P[i]	06:05	07:10	11:25	12:20	14:25	18:00	19:30

(a) (i) According to the content of `P` above, find the return values of the following:

(1) `F1(07:10, 14:25)` _____

(1 mark)

(2) `F1(11:25, 11:25)` _____

(1 mark)

(ii) Complete the following pseudocode for `F1`.

```
F1(s, t)
   count ← 0
   for i from 1 to n do
       if [          ] AND [          ] then
           count ← [          ]
   return count
```

(3 marks)

Peter writes a subprogram `BS` using binary search to find the index of the train departed at time `t`:

```
BS(t)
   s ← 1
   e ← n
   while s ≤ e do
      m ← (s+e)/2
      if  P[m] = t  then
            exit subprogram and return m
      else
            if  t > P[m]  then
                s ← m + 1
            else
                e ← m - 1
   return m
```

(b) Refer to the content of `P` above.

(i) What is the return value of `BS(14:25)`? _____

(1 mark)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

Please stick the barcode label here.

(ii) How many times will the 5th line in BS '`m ← (s+e)/2`' be executed when calling BS(19:30)?

_____

(1 mark)

(iii) Peter rewrites F1 as F2. F2 uses binary search to find the index of the train departed at time s. Then it sequentially searches the number of trains departed on or before time t. Complete the pseudocode for F2 below.

```
F2(s, t)
    count ← 0
    for i from BS( [      ] ) to n do
        if [                ] then
            count ← count +1
    return count
```

(3 marks)

(iv) Peter rewrites F1 as F3. F3 uses binary search twice to find the number of trains departed from time s to time t inclusive. Complete the pseudocode for F3 below.

```
F3(s, t)
    a ← BS(s)
    b ← BS(t)
    count ← [                ]
    return count
```

(2 marks)

(c) Peter writes F4 for cases that there may not be a train departed at time s or time t. For example, according to the content of P above, F4(12:00, 18:30) returns 3. Complete the pseudocode for F4 below.

```
F4(s, t)
    count ← F3(s, t)
    a ← BS(s)
    b ← BS(t)
    if  P[a] < s  then
        count ← count - 1
    if [                ] then
        count ← [                ]
    return count
```

(3 marks)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

2. Tom develops a system to arrange dropping off boxes from a ship to storage areas. He uses a string array `S` of size `k` to represent the storage areas. In the following example, areas `3` and `6` each stores a box of apples whereas the other areas are empty.

i	1	2	3	4	5	6	7	8
S[i]			Apple			Apple		

Tom writes the pseudocode for a subprogram `CE(N)` that returns `TRUE` if there are `N` or more empty areas, `FALSE` otherwise.

```
CE(N)
    count ← 0
    for i from 1 to k do
          if  S[i] is empty  then
                count ← count + 1
    if count ≥ N then
          return TRUE
    else
          return FALSE
```

(a) Suppose that the initial content of `S` is

i	1	2	3	4	5	6	7	8
S[i]			Apple			Apple		

(i) What is the return value of `CE(3)`? _____

(1 mark)

(ii) Find the minimum value of `N` such that `CE(N)` returns `FALSE`. _____

(1 mark)

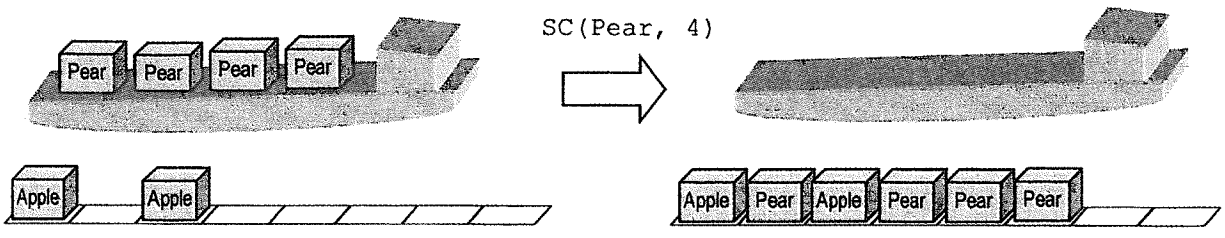
Tom writes a subprogram `SC(T, N)`. If there are less than `N` empty areas, `SC` only returns `-1`. Otherwise, `SC` puts `N` boxes of `T` in the empty areas with the smallest index first and returns `0`.

For example, the initial content of `S` is

i	1	2	3	4	5	6	7	8
S[i]	Apple		Apple					

After executing `SC(Pear, 4)`, `S` becomes

i	1	2	3	4	5	6	7	8
S[i]	Apple	Pear	Apple	Pear	Pear	Pear		



Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

Please stick the barcode label here.

(b) Suppose that the initial content of `S` is

i	1	2	3	4	5	6	7	8
S[i]			Apple			Apple		

(i) What is the content of `S` after executing `SC(Peach, 4)`, `SC(Melon, 3)` and `SC(Mango, 1)` sequentially?

i	1	2	3	4	5	6	7	8
S[i]			Apple			Apple		

(2 marks)

(ii) What is the return value of `SC(Plum, 9)`? _____

(1 mark)

(iii) Complete the pseudocode for `SC` below.

```
SC(T, N)
   if CE(N) then
      count ← N
      i ← 0
      repeat
            i ← i + 1
            if S[i] is empty then
                S[i] ← [          ]
                count ← [         ]
      until [                     ]
      return 0
   else
      return -1
```

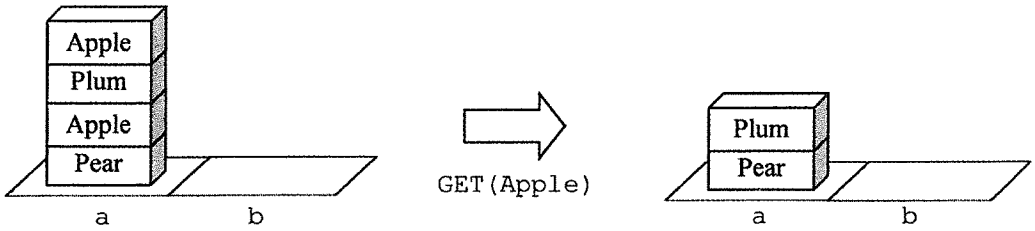
(3 marks)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

(c) Tom stacks up the boxes in an area `a`. He writes a subprogram `GET(T)`, which uses stack operations to remove all boxes of `T` and keep the remaining boxes in the same order, with the use of an empty area `b`. An example is shown below:



initial status

the status after executing `GET(Apple)`

Tom uses the following stack operations:

Stack operation	Description
`push(X, T)`	Store a box of `T` in area `X`.
`pop(X)`	Remove a box from area `X` and return the box.

Complete the pseudocode for `GET` below.

```
GET(T)
   while a is not empty do
         temp ← pop(a)
         if  temp <> T  then
               push(  [          ]  )
   while  [                        ]  do
         push(  [          ]  )
```

(3 marks)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

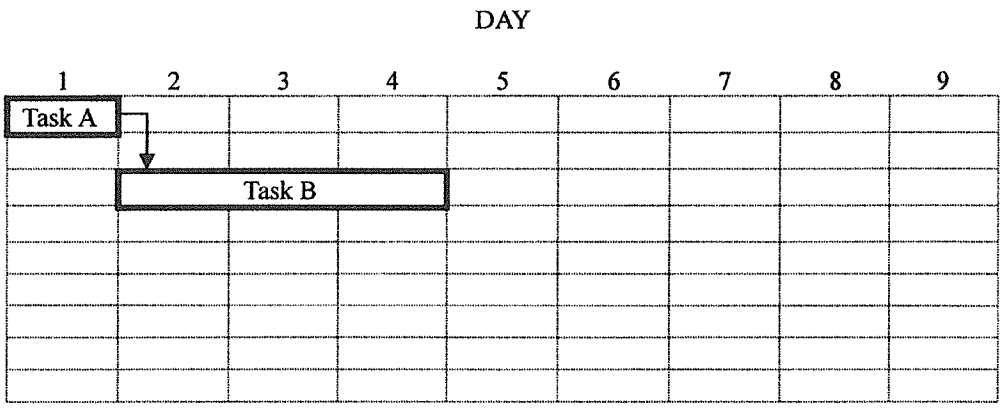
Answers written in the margins will not be marked.

Please stick the barcode label here.

(d) Tom has the following tasks in the system development.

Task	Description	Duration (day)	Depending on
A	Develop S	1	-
B	Develop CE	3	A
C	Develop GET	2	A
D	Develop SC	2	B
E	System testing	1	B, C, D

(i) Complete the Gantt Chart below.



(3 marks)

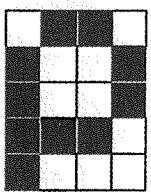
(ii) What is the minimum number of days for completing all tasks? _____

(1 mark)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

3. `A` is an `m×n` two-dimensional array. Mary uses `A` to represent an image. A white pixel and a black pixel are represented by '`0`' and '`1`' respectively. The following example shows a 5×4 image represented by `A`.



Image

		n			
		1	2	3	4
m	1	0	1	1	0
	2	1	0	0	1
	3	1	0	0	1
	4	1	1	1	0
	5	1	0	0	0

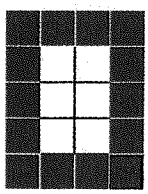
A

`B` is a one-dimensional array with `m×n` elements. Mary uses `B` for transmitting images through the Internet.

Mary writes a subprogram `AtoB` for converting `A` to `B` by copying data in `A` to `B` row by row from the top to the bottom. For example, `A` in the above is converted to `B` below.

idx	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
B[idx]	0	1	1	0	1	0	0	1	1	0	0	1	1	1	1	0	1	0	0	0

(a) Mary uses `A` to represent the following image. Write the content of `B` after executing `AtoB`.



idx	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
B[idx]																				

(2 marks)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

(b) (i) Complete the following pseudocode for `AtoB` that converts `A` to `B`.

```
AtoB
        for i from 1 to m do
            for j from 1 to n do
                B[ [          ] + [     ] ] ← A[i,j]
```

(3 marks)

(ii) Mary writes a subprogram `BtoA` that converts `B` to the original data structure `A`. Complete the pseudocode for `BtoA` below.

```
BtoA
        for i from 1 to m do
            for j from 1 to n do
                [                                    ]
```

(1 mark)

Answers written in the margins will not be marked

Answers written in the margins will not be marked.

A, B and P are global arrays.

(c) Mary finds that during data transmission, some array elements may be transmitted wrongly. She writes a subprogram Add1 to assign values to P. Then, P[i] can be used for error checking on the i-th column of A.

```
Add1
   for i from 1 to n do
        count ← 0
        for j from 1 to m do
             if A[j,i] = 1 then
                  count ← count + 1
        if remainder of (count/2) = 0 then
             P[i] ← 1
        else
             P[i] ← 0
```

(i) Suppose that the content of A is

m \ n	1	2	3	4
1	0	1	1	1
2	1	0	0	0
3	1	1	1	1
4	1	0	0	0
5	1	1	1	0

What is the content of P after executing ADD1?

i	1	2	3	4
P[i]				

(2 marks)

(ii) Refer to the following P and A with one error element.

m \ n	1	2	3	4
1	1	0	1	0
2	0	1	0	1
3	1	0	1	0
4	1	1	0	0
5	1	0	0	0

A

i	1	2	3	4
P[i]	1	0	1	0

Which column in A contains the error element?

_____

(1 mark)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

(d) (i) Mary considers using procedural languages and object-oriented languages to write the subprograms. Give one major difference between these two kinds of programming language.

(2 marks)

(ii) Mary considers using interpreters and compilers. Give an advantage of each type of translator.

Interpreters:

Compilers:

(2 marks)

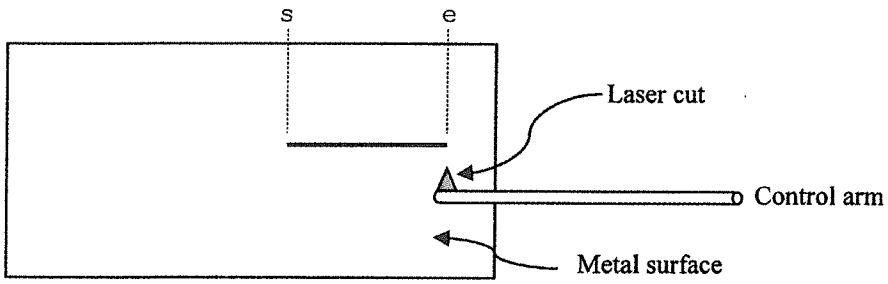
(iii) Mary chooses to use program libraries instead of writing her own subprograms. Give **two** reasons to support her choice.

(2 marks)

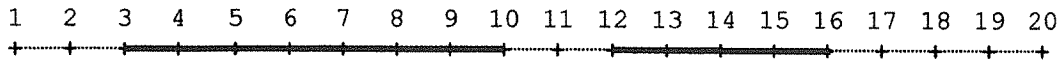
Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

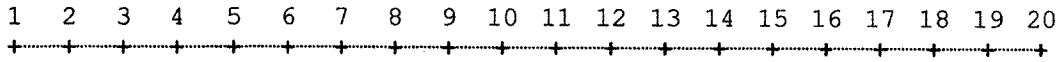
4. Susan develops a program that controls a laser cutter to cut a metal surface.



Susan uses `(s,e)` to represent the horizontal line from position `s` to position `e` where `s < e`. She writes a subprogram `Hcut(s,e)` for cutting `(s,e)`. For example, executing `Hcut(3,10)` and `Hcut(12,16)` will cut the two lines, as shown below:

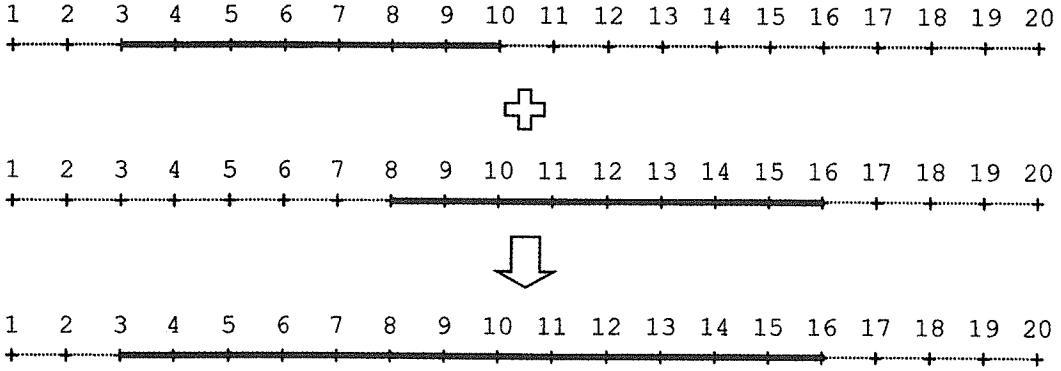


(a) (i) Draw the lines to be cut by executing `Hcut(2,4)`, `Hcut(11,13)` and `Hcut(13,19)`.



(2 marks)

(ii) Susan finds that the execution of `Hcut(3,10)` and `Hcut(8,16)` is the same as the execution of `Hcut(3,16)`.



Simplify the execution of `Hcut(2,7)` and `Hcut(6,9)` so that `Hcut` can be executed once only.

`Hcut(_____,_____)`

(1 mark)

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

(iii) Susan develops a subprogram `MH(s1, e1, s2, e2)` with two overlapping horizontal lines `(s1,e1)` and `(s2,e2)` as input parameters that returns a simplified horizontal line. For example, `MH(3, 10, 8, 16)` will return `(3,16)`. Complete the pseudocode for `MH` below.

```
MH(s1, e1, s2, e2)
    a ← minimum of s1 and s2
    b ← [                    ]
    return [                 ]
```

(2 marks)

(b) Susan writes a subprogram `CO` that checks whether two horizontal lines `(s1,e1)` and `(s2,e2)` overlap each other. Complete the pseudocode for `CO` below.

```
CO(s1, e1, s2, e2)
   if (s1 ≤ e2) AND ( [          ] ) then
      return TRUE
   else
      return FALSE
```

(2 marks)

(c) `L` is a list that contains horizontal lines. `L[i]` represents the `i`-th node that contains a horizontal line `(L[i].s, L[i].e)`. Susan writes the following pseudocode for merging all overlapping horizontal lines in `L`. Complete the pseudocode below by using `MH`.

```
for any L[i] and L[j] in L, do
  if i <> j then
     if CO(L[i].s, L[i].e, L[j].s, L[j].e) then
        remove L[ [    ] ] and L[ [    ] ] from L
        insert [                    ] into L
```

(3 marks)

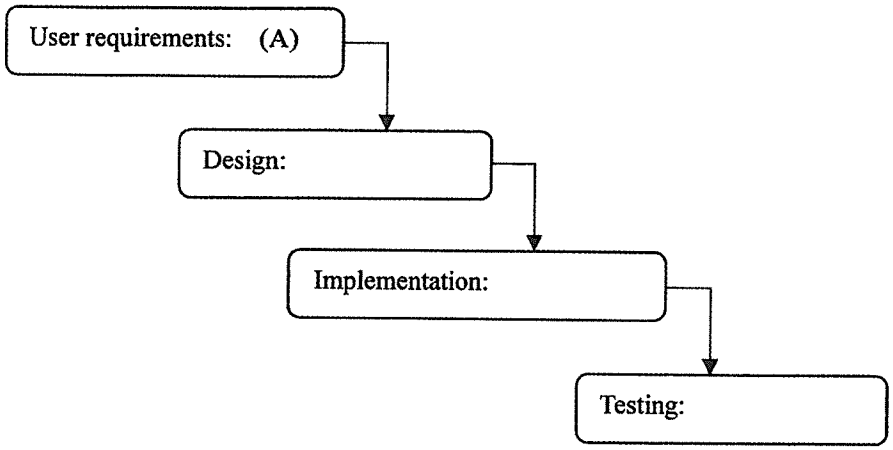
Answers written in the margins will not be marked.

Answers written in the margins will not be marked.

(d) Susan uses a waterfall model to develop a system that controls the laser cut with the requirements below:

- Do the following tasks:
  - (A) Collect user requirements.
  - (B) Perform a user acceptance test.
  - (C) Draw a flowchart for the system.
  - (D) Design the user interface.
  - (E) Write the program source code.
- A 'Maintenance' stage is needed for doing the following task:
  - (F) Improve the system performance.
- Program bugs that may be found in the 'Testing' stage need to be debugged.

Complete the waterfall model below.



(5 marks)

END OF PAPER

Answers written in the margins will not be marked.

Answers written in the margins will not be marked.