HONG KONG EXAMINATIONS AND ASSESSMENT AUTHORITY

HONG KONG DIPLOMA OF SECONDARY EDUCATION EXAMINATION 2012

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

Please stick the barcode label here.

Candidate Number								

Programming Language Used (Please tick one)	Pascal	☐
	C	☐
	Visual Basic	☐
	Java	☐

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)   Tick the appropriate box for the programming language used. **No marks will be awarded if you tick either more than one box or no boxes.**

(3)   **ANSWER ALL 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.

(4)   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.

(5)   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.

Answer all questions.

1. (a) The following algorithm, ALG1, processes a non-negative integer variable, N, and stores the results in an array, X, with indexes from 1 to 6.

ALG1
Step 1: Initialise each element in X with a value 0.
Step 2: i ← 0
Step 3: While N >= 1 do Steps 4 to 6
Step 4:   X[6 - i] ← remainder of (N / 2)
Step 5:   N ← integral part of (N / 2)
Step 6:   i ← i + 1

(i) Suppose N = 29. Dry run ALG1.

(1) Fill in the contents of X.

X[1]	X[2]	X[3]	X[4]	X[5]	X[6]

(2) How many iterations of the loop will be executed? _____

(ii) What is the range of the values of N that ALG1 can process without causing an error? Explain briefly.

_____

_____

_____

_____

(iii) Study the pattern of the values in X in (a)(i). What is the purpose of ALG1?

_____

_____

(6 marks)

(b) The following algorithm, ALG2, processes a non-negative integer variable, N, and stores the results in an array, Y, with indexes from 1 to 6. N is smaller than 64.

ALG2
Step 1: Initialise each element in Y with a value 0.
Step 2: j ← 1
Step 3: While N > 0 do Steps 4 to 7
Step 4:   If N >= Z[j] Then do Steps 5 to 6
Step 5:   Y[j] ← 1
Step 6:   N ← N - Z[j]
Step 7:   j ← j + 1

Z is an array with the following initial values.

Z[1]	Z[2]	Z[3]	Z[4]	Z[5]	Z[6]
32	16	8	4	2	1

(i) Suppose N = 48. Dry run ALG2. Fill in the content of Y.

Y[1]	Y[2]	Y[3]	Y[4]	Y[5]	Y[6]

(ii) In the worst case, how many iterations of the loop will be executed?

_____

(iii) Give a value of N (N > 0) and show that ALG2 will execute fewer iterations than ALG1 does.

_____

_____

_____

_____

_____

(6 marks)

Provided by dse.life

(c) ALG2 is implemented by a programming language called 'ICT'. The source program will be compiled to an 'ICT bytecode' which is machine independent. An 'ICT virtual machine' (ICT–VM) will be used to interpret and execute the 'ICT bytecode' whenever the program is executed. The diagram below shows this process.
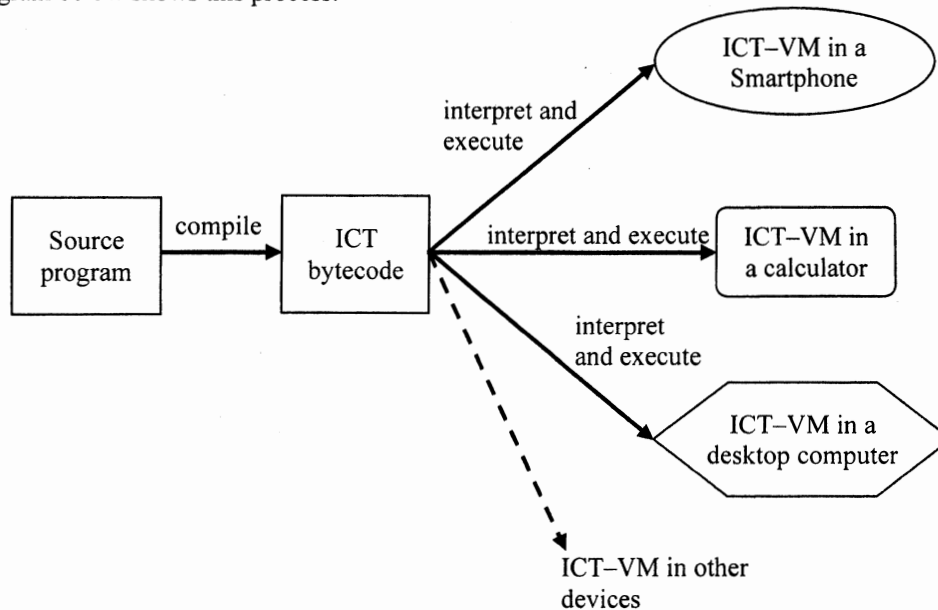
(i) Referring to the above diagram, suggest an advantage of using 'ICT bytecode'.

_____

_____

(ii) Suppose machine code is used instead of 'ICT bytecode'.

(1) The process will change. Describe the new process briefly.

_____

_____

(2) What is the advantage of this change?

_____

_____

_____

(3 marks)

2. A company installs a smart card attendance system. When entering the company, each staff member taps his or her smart card on a card reader and the arrival time will be stored in a stack in the smart card.

(a) The stack can store a maximum of 31 data items on arrival time. It is implemented by an array, A, with an integer variable X. X stores the index of the next available array element for the stack. X stores 32 when A is full. A[1] is the first element.

The stack is initialised at the beginning of each month so that all data items in the stack are removed and X = 1.
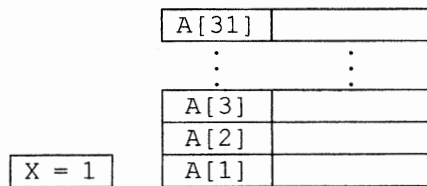
(i) RET(A) is a subprogram that retrieves a data item on arrival time and removes it from A. Suppose that a smart card has stored the arrival time for the first 19 days. In order to retrieve the arrival time of the 4th day, how many times should RET(A) be called? What is the value of X after the calls?

RET(A) will be called _____ times; X = _____

(ii) An extra stack is needed for the operation in (a)(i).

(1) What would happen if only A were used?

_____

_____

_____

_____

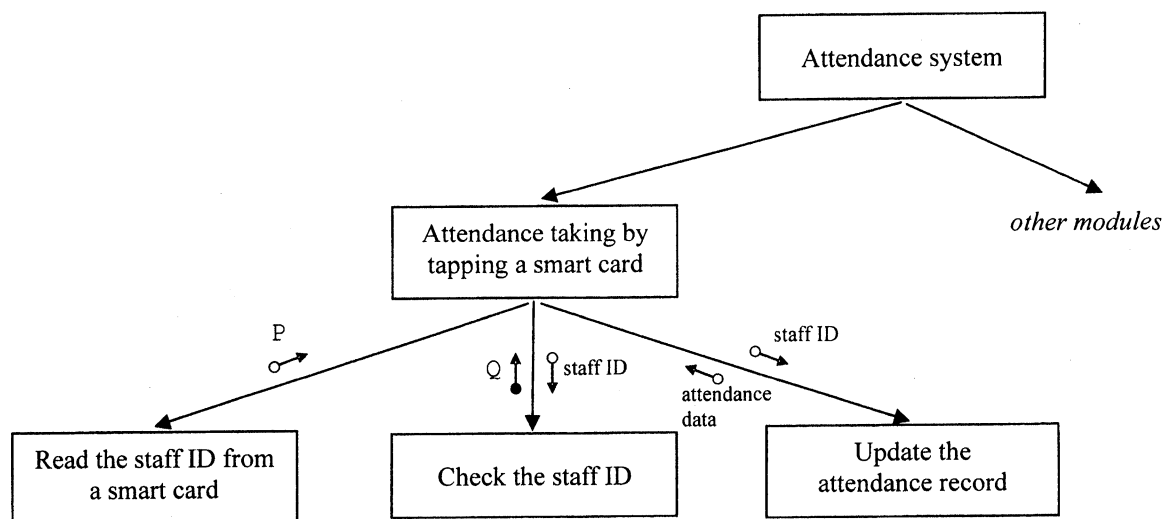(2) How is the extra stack used in the operation?

_____

_____

(iii) If the stack continues to be used next month without initialisation, what will happen?

_____

(6 marks)

(b) Each smart card is identified by a staff ID. Some modules in this attendance system are shown in the structure chart below.

The symbol ○ represents data that is passed between modules. The symbol ● represents information resulted from a validation test.

(i) What is P?

_____

(ii) What is Q? Briefly describe how Q is used.

_____

_____

_____

(3 marks)

(c) The company hires a software company to develop the attendance system.

(i) The software company carries out a user acceptance test, system test and unit test before the delivery of the system.

(1) Which test should be carried out first? _____

(2) Which test should be carried out last? _____

(3) State the main purpose of each kind of test.

User acceptance test: _____

_____

System test: _____

_____

Unit test: _____

_____

(ii) The company adopts a direct cutover approach to converting the system.

(1) What is the major risk of this approach?

_____

_____

(2) Why does the company choose to adopt this approach despite the risk?

_____

_____

(7 marks)

3. A company uses a computer program to control the use of lifts in a commercial building. The building has four lifts with lift numbers 1, 2, 3 and 4. On the ground floor, passengers can enter a floor number on a control panel. The program will find and display a lift number, as shown in the example below. The floor number of the ground floor is 0.
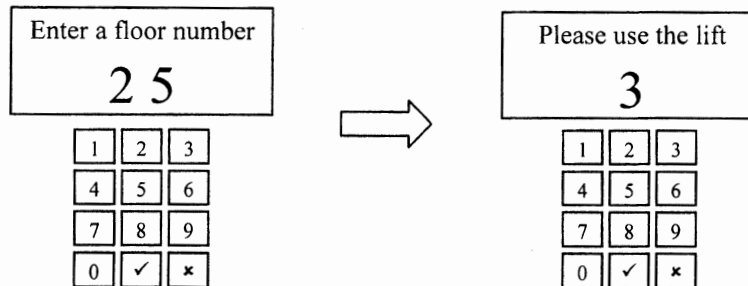
(a) After entering a floor number, a lift will be randomly selected to move to the ground floor. The subprogram, myrand, with an input parameter K will return a random integer between 0 and K-1 inclusive. Write a subprogram, call_random, with an input parameter N where N stores the total number of lifts, that will return a lift number to emulate the random selection.

(2 marks)

In order to ensure a more efficient use of lifts, a subprogram, closest, replaces call_random.

The following statements are declared before closest where MAXFLOOR and LIFTTOTAL respectively store the highest floor number and the total number of lifts in the building.

[Pascal version]
```
const MAXFLOOR = 60;
      LIFTTOTAL = 4;
```

[Visual Basic version]
```
const MAXFLOOR As Integer = 60
const LIFTTOTAL As Integer = 4
```

[C version]
```
#define MAXFLOOR 60
#define LIFTTOTAL 4
```

[Java version]
```
static final int MAXFLOOR = 60;
static final int LIFTTOTAL = 4;
```

(b) Why is the use of MAXFLOOR and LIFTTOTAL regarded as good programming style? Give **two** reasons.

_____

_____

_____

(2 marks)

(c) `LiftPos` is a global integer array which stores the positions of all lifts. Assume that `LiftPos[i]` has already stored the floor number of lift `i`.

By making use of MAXFLOOR and LIFTTOTAL, write the subprogram, `closest`, to return the lift number of a lift that is closest to the ground floor, with the following requirements:

- Declare integer variables `cPos` and `cLift` to store the position and lift number of a lift respectively;
- Declare an integer variable `i` to store an index;
- Use a `for` loop to find a lift that is closest to the ground floor.

<div style="border:1px solid black; min-height:500px;"></div>

(7 marks)

(d) During system development, a system analyst collects user requirements from passengers.

(i) Suggest **two** ways in which the system analyst can collect user requirements.

_____

_____

_____

_____

(ii) Give a user requirement that may be proposed by the passengers.

_____

_____

(3 marks)

Provided by dse.life

4.　(a)　A text file `club.txt` stores a maximum of 100 strings and the maximum length of each string is 6 characters. An example is shown below.

```
sports
music
chess
art
ict
```

A subprogram, `ReadData`, has a formal parameter `A` which is an array. It is called by reference to read all data in `club.txt` and store them in `A`.

[Pascal version]

```
type arraytype = array[0..99] of string[6];
procedure ReadData(    Part (a)(i)    );
var i : integer;
    infile : text;
begin
   assign(infile, 'club.txt');
   reset(infile);
   i := 0;
         Part (a)(ii)

   close(infile);
end;
```

[C version]

```
void ReadData(    Part (a)(i)    ) {
   int i;
   FILE *infile;
   infile = fopen("club.txt", "r");
   i = 0;
         Part (a)(ii)

   fclose(infile);
}
```

[Visual Basic version]

```
Sub ReadData(    Part (a)(i)    )
   Dim i As Integer
   Dim infile As IO.StreamReader
   Dim item As String
   infile= IO.File.OpenText("club.txt")
   i = 0
   item = infile.ReadLine()
         Part (a)(ii)

   infile.Close()
End Sub
```

[Java version]

```
public static void ReadData( [Part (a)(i)] ) throws IOException {
    int i;
    String item;
    BufferedReader infile =
                new BufferedReader(new FileReader("club.txt"));
    i = 0;
    item = infile.readLine();
    [Part (a)(ii)]
    infile.close();
}
```

(i) Write down the formal parameter list in ReadData.

_____

(ii) Write down a while loop to complete ReadData.

(5 marks)

(b) The following algorithm uses *Insertion Sort* to sort the data in A in ascending order. The size of A is N and the first index of A is 0.

<pre>
Step 1:   For j = 1 to N - 1 do Steps 2 to 7
Step 2:   Temp ← A[j]
Step 3:   i ← j - 1
Step 4:   While i >= 0 and A[i] > Temp do Steps 5 to 6
Step 5:   A[i + 1] ← A[i]
Step 6:   i ← i - 1
Step 7:   A[i + 1] ← Temp
</pre>

(i) Dry run the algorithm. List the contents of A immediately after the second and the third passes of Step 7.

	A[0]	A[1]	A[2]	A[3]	A[4]
Initial	sports	music	chess	art	ict
First pass	music	sports	chess	art	ict
Second Pass					
Third pass					
Fourth Pass	art	chess	ict	music	sports

(ii) Suppose Step 1 is changed to

For j = 0 to N - 1 do Steps 2 to 7

How does this change the program execution? Explain briefly.

_____

_____

_____

(iii) Describe the content of A that requires the longest computation time to sort.

_____

_____

(5 marks)

(c) A program is written in a logic language with the following facts and rules.

Fact	Clause in the program
Sports Club belongs to Student Union	`belongsto(sports, union).`
Music Club belongs to Student Union	`belongsto(music, union).`
Chess Club belongs to Student Union	`belongsto(chess, union).`
Art Club belongs to Student Union	`belongsto(art, union).`
Flute group belongs to Music Club	`belongsto(flute, music).`
Oboe group belongs to Music Club	`belongsto(oboe, music).`
Basketball group belongs to Sports Club	`belongsto(basketball, sports).`

Rule	Clause in the program
X is a club if it belongs to Student Union.	`club(X) :- belongsto(X, union).`
X is a group if it belongs to a club.	`group(X) :- belongsto(X, Y) & club(Y).`

The following are some examples of the results of some queries:

Query	Result
`?- belongsto(chess, union).`	`true.`
`?- belongsto(science, union).`	`false.`
`?- club(A).`	`A = sports, music, chess, art.`

(i) What are the results of the following queries?

    (1) `?- group(volleyball).`

                         _____

    (2) `?- belongsto(B, music).`

                         _____

    (3) `?- group(C).`

                         _____

(ii) Write a query to find the organisation that Art Club belongs to.

_____

(iii) What is the benefit of using logic languages over procedural languages?

_____

_____

_____

                                                          (5 marks)

END OF PAPER