OCR
Oxford Cambridge and RSA

Wednesday 3 June 2015 – Afternoon

GCE COMPUTING

F452/01 Programming Techniques and Logical Methods

Candidates answer on the Question Paper.

OCR supplied materials:
None

Other materials required:
None

Duration: 1 hour 30 minutes

F452 01

2714081747

Candidate forename		Candidate surname	

Centre number						Candidate number				

INSTRUCTIONS TO CANDIDATES

- Write your name, centre number and candidate number in the boxes above. Please write clearly and in capital letters.
- Use black ink. HB pencil may be used for graphs and diagrams only.
- Answer **all** the questions.
- Read each question carefully. Make sure you know what you have to do before starting your answer.
- Write your answer to each question in the space provided. Additional paper may be used if necessary but you must clearly show your candidate number, centre number and question number(s).
- Do **not** write in the bar codes.

INFORMATION FOR CANDIDATES

- The number of marks is given in brackets **[ ]** at the end of each question or part question.
- The total number of marks for this paper is **100**, the quality of written communication will be assessed where an answer requires a piece of extended writing.
- This document consists of **24** pages. Any blank pages are indicated.

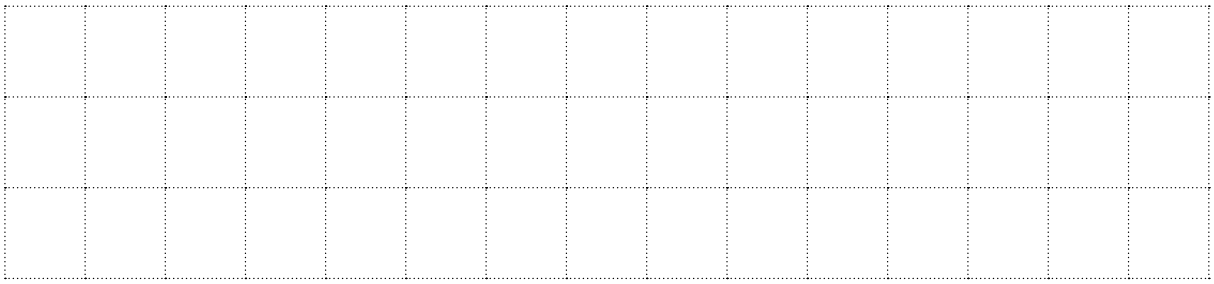
BLANK PAGE

PLEASE DO NOT WRITE ON THIS PAGE

1 The organisers of an international football competition are planning to use a large electronic score board to display information to spectators in the stadium. The board can display three lines of text of 15 characters each.

(a) During most of the match, the display should show the names of the two teams playing, the number of goals they have each scored and the time elapsed (in minutes and seconds) since the start of the match.

Use the outline below to design how this information can be displayed using example data.



[3]

State **three** assumptions you have made.

1 .....

.....

2 .....

.....

3 .....

.....

[3]

(b) A modular design is used to produce the program that controls the display.

Describe what is meant by a modular design and state **one** advantage of a modular design.

Description .....

.....

.....

.....

Advantage .....

.....

[3]

The program stores the text to be displayed in an array called Board, so that

- Board(1,1) contains the letter in the top left corner of the display board
- Board(3,15) contains the letter in the bottom right corner of the display board.

A module in the program updates the display every time the contents of this array are changed.

(c) State the identifier, number of dimensions and most appropriate data type of the array Board.

Identifier .....

Number of dimensions .....

Most appropriate data type .....

[3]

(d) The program contains a module which clears the display using a routine to insert a space in each element of the array using the following algorithm.

Complete this algorithm by filling in the blanks.

```
01    PROCEDURE ClearDisplay
02
03      FOR Row = 1 TO 3
04
05        FOR Column = 1 to ............
06
07          Board( Row, ............) = " "
08
09        NEXT Column
10
11      NEXT ...........
12
13    END PROCEDURE
```

[3]

The program contains a module which displays a message at a given position using the algorithm below. For example, DisplayString("HELLO",2,1) should display the message "HELLO" on the second row, starting from the first column.

```
01    PROCEDURE DisplayString(Message,Row,Column)
02      FOR i = 1 TO LENGTH(Message)
03        Board(Row, Column + i) = MID(Message,i,1)
04      NEXT i
05    END PROCEDURE
```

MID(Message,i,1) returns the character at position i in the string.

(e) There is an error in line 03.

(i) State the effect of this error.

.....

..... **[1]**

(ii) State how the error should be corrected.

.....

..... **[1]**

(iii) State the type of error that this is.

.....

..... **[1]**

(f) Explain why a different error would occur if the procedure is called with the arguments DisplayString("BRAZIL", 1, 10). State the type of error that this is.

.....

.....

.....

.....

.....

..... **[3]**

(g) The program contains a module which changes the display when a goal is scored, using the corrected procedure DisplayString.

The algorithm for this module has the following requirements.

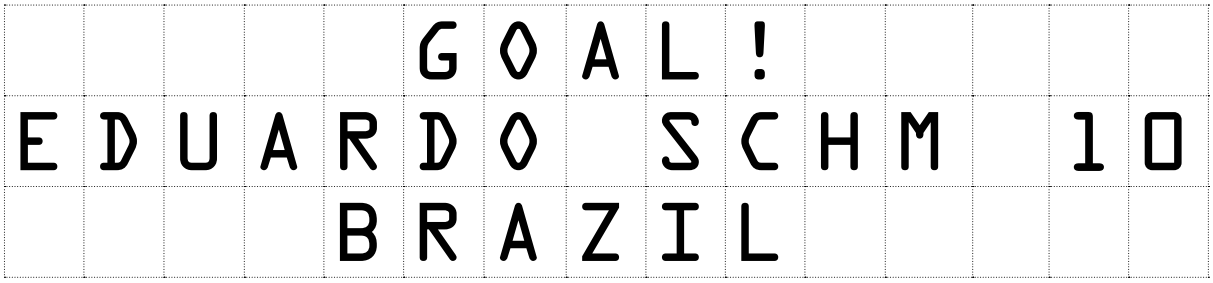
- The algorithm should assume that the display has been cleared and is blank
- The algorithm should require the name, number and team of the player who has scored
- The algorithm should use the procedure

DisplayString(Message, Row, Column)

to display the following text.

  - The first line of the display should say "GOAL!" in the centre
  - The second line should show the name of the player who scored the goal on the left, and the player's number on the right. (If the name is too long, the algorithm should use as many letters as possible. The player's number may be 1 or 2 digits.)
  - The third line should show the name of the team that scored in the centre. (If the name of the team is too long, the algorithm should use as many letters as possible.)

An example of a display which meets these requirements is shown below:



Write an algorithm to update the display according to these requirements.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

.... [8]

2 A Huffman code is a type of binary code where characters are represented by binary numbers of different lengths. A possible Huffman code for a character set of four characters is:

A = 0 B = 11 C = 100 D = 101

For example the word BAD would be represented by 110101.

(a) State how the word CAB would be represented in this code.

..... **[1]**

The following algorithm takes a message as binary digits, one at a time, from a source and outputs the message that is being transmitted.

```
01    d = ""
02    REPEAT
03      x =  next binary digit from source
04      d = d + x
05      SELECT CASE d
06        CASE "0" :
07          OUTPUT "A"
08          d = ""
09        CASE "11" :
10          OUTPUT "B"
11          d = ""
12        CASE "100" :
13          OUTPUT "C"
14          d = ""
15        CASE "101" :
16          OUTPUT "D"
17          d = ""
18      END SELECT
19    UNTIL end of transmission
```

(b) The algorithm uses identifiers.

(i) State what is meant by an identifier and name an identifier in the algorithm.

.....

.....

.....

..... **[2]**

(ii) State why the identifiers make the algorithm difficult to understand.

.....

..... **[1]**

(c) Explain the purpose of line 01.

.....

.....

.....

..... **[2]**

(d) State what the operation + does on line 04. State the name of this operation.

.....

.....

.....

..... **[2]**

(e) The source of the message needs a routine to encode messages into the Huffman code. The routine should allow the user to enter a message and output the encoded message.

Write this routine in a high level language you have studied, stating the name of the language you have used. You should use good program writing techniques to ensure that your routine is easy to understand.

You can assume that the message consists only of the characters A, B, C and D.

Name of language .....

Routine

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[7]**

(f) Programming language environments provide several facilities for editing and debugging programs.

Name **two** of these facilities. Describe how each can be used when writing the routine in part **(e)**.

1 .....

.....

.....

.....

.....

2 .....

.....

.....

.....

.....

[6]

BLANK PAGE

PLEASE DO NOT WRITE ON THIS PAGE

3 A primary school uses a top-up smart card system to take payments for school meals.

Every pupil is given a card when they join the school. Each card has a six digit identification number.

(a) Data about the cards is stored in a sequential file called CardFile.

Describe what is meant by a sequential file.

.....

.....

.....

..... **[2]**

(b) Each record in CardFile contains data as in the table below.

For each item of data, state the most appropriate data type and the size in bytes.

Item	Data type	Size in bytes
The card's six digit identification number		
The amount of credit on the card		
Whether the owner of the card is entitled to free school meals		

[6]

(c) The school has 100 pupils.

Calculate an estimate of the size of the file in bytes.

You **must** show your working.

.....

.....

.....

.....

.....

..... **[3]**

When a pupil tops up a card, the following algorithm is used to update the amount of credit on the card. The algorithm is written in pseudocode.

```
01    INPUT CardToTopUp, AmountToAdd
02    OPEN CardFile in READ MODE
03    OPEN NewFile in WRITE MODE
04    REPEAT
05      READ CardId, Amount, FreeMeals FROM Cardfile
06      IF CardID = CardToTopUp THEN
07        NewAmount = Amount + AmountToAdd
08        WRITE CardID, NewAmount, Freemeals TO NewFile
09      ELSE
10        WRITE CardID, Amount, FreeMeals TO NewFile
11      END IF
12    UNTIL CardFile at end of file
13    CLOSE CardFile
14    CLOSE NewFile
15    Replace CardFile with NewFile
```

(d) Explain the difference in the use of = in lines 06 and 07, identifying the type of operator being used in each case.

.....

.....

.....

.....

.....

.....

.....

..... **[4]**

(e) At the start of each day, a routine is executed which tops up the cards of all pupils who are entitled to free school meals with £3.50.

```
01    OPEN CardFile in READ MODE

02    OPEN NewFile in WRITE MODE

03    REPEAT

04      READ CardId, Amount, FreeMeals FROM Cardfile

05      IF ......................  THEN

06        NewAmount = ..........................

07        WRITE CardID, NewAmount, Freemeals TO NewFile

08      ELSE

09        WRITE CardID, Amount, ............. TO NewFile

10      END IF

11    UNTIL CardFile at end of file

12    CLOSE CardFile

13    CLOSE NewFile

14    Replace CardFile with NewFile
```

Complete the algorithm for this routine by filling in the spaces.

[3]

(f) When a new pupil is given a card, the record for the card needs to be inserted into the file.

Write an algorithm in pseudocode which:

- Allows the user to input the six-digit identification number, the initial amount of credit and whether the pupil has free school meals
- Produces a new sequential file with the record for the new card inserted.

The quality of written communication will be assessed in your answer to this question.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... **[8]**

4 Some whole numbers are known by mathematicians as evil numbers.
One way to find out if a number is evil, is to use the integer division operators DIV and MOD.

(a) Complete the following calculations.

9 DIV ..... = 2

7 MOD 2 = ...

1 MOD 3 = ...

[3]

The following function determines whether a number is evil.

```
01    FUNCTION IsEvil(n : INTEGER)
02      Temp = TRUE
03      WHILE (n > 0)
04        IF (n MOD 2) = 1 THEN
05          Temp = NOT(Temp)
06          n = n - 1
07        END IF
08        n = n DIV 2
09      END WHILE
10      RETURN Temp
11    END FUNCTION
```

(b) Describe how iteration has been used in this function.

.....

.....

.....

..... **[2]**

(c) 0 is an evil number.

Describe each step of the execution of the call IsEvil(0), showing that it returns the value TRUE.

.....

.....

.....

.....

.....

..... **[3]**

(d) Using the trace table below, show what happens in the execution of the call IsEvil(2), showing that 2 is not an evil number.

You should use a new row in the table for every line that is executed, and show any values that are changed during the execution of that line. You may not need every row in the table. The first two rows have been filled in for you.

Line Number	n	Temp	Comment
01	2		Call IsEvil(2)
02		TRUE	Temp = TRUE

[6]

All numbers that are not evil are known as odious numbers.

The following function determines whether a number is odious.

```
01    FUNCTION IsOdious(n : INTEGER)
02      IF n = 0 THEN
03        RETURN FALSE
04      ELSE
05        IF n MOD 2 = 0 THEN
06          RETURN IsOdious(n DIV 2)
07        ELSE
08          RETURN NOT(IsOdious(n DIV 2))
09        END IF
10      END IF
11    END FUNCTION
```

(e) Describe how recursion has been used in this function.

.....

.....

.....

..... **[2]**

(f) Many functions can be defined using either recursion or iteration.

(i) State **one** advantage of using recursion instead of iteration.

.....

.....

(ii) State **one** disadvantage of using recursion instead of iteration.

.....

..... **[2]**

(g) 2 is an odious number.

Show each step of the execution of the call IsOdious(2), including all recursive calls and the values returned. You may use a diagram.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

[6]

END OF PAPER

BLANK PAGE

PLEASE DO NOT WRITE ON THIS PAGE

BLANK PAGE

PLEASE DO NOT WRITE ON THIS PAGE

PLEASE DO NOT WRITE ON THIS PAGE