Cambridge International AS & A Level

Cambridge International Examinations
Cambridge International Advanced Subsidiary and Advanced Level

CANDIDATE NAME

CENTRE NUMBER | | | | | |

CANDIDATE NUMBER | | | | |

6264905153

COMPUTER SCIENCE **9608/21**

Paper 2 Fundamental Problem-solving and Programming Skills **May/June 2018**

2 hours

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names of software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 75.

This document consists of **15** printed pages and **1** blank page.

CAMBRIDGE International Examinations

Question 1 begins on the next page.

1 **(a)** A program stores data about hospital patients.

Give a suitable **identifier name** for each of the data items.

Description of data item	Suitable identifier name
The temperature of the patient	
The temperature of the room	
The patient identification number	
The name of the nurse taking the measurement	

[4]

(b) **(i)** Program variables have values as follows:

Variable	Value
`MyGreeting`	`"Happy Birthday"`
`MyInitial`	`'C'`
`AgeInYears`	`27`
`Weight`	`60.5`
`Married`	`TRUE`
`Children`	`TRUE`

Evaluate each expression in the following table.
If an expression is invalid, write ERROR.

For the built-in functions list, refer to the **Appendix** on page 15.

Expression	Evaluates to
`"Mon" & MID(MyGreeting, 10, 2)`	
`AgeInYears + ASC(MyInitial)`	
`INT(MyInitial)`	
`MOD(Weight * 2, 10)`	
`Married AND (NOT Children)`	

[5]

(ii) Programming languages support different data types.

Give an appropriate data type for each of these variables from **part (b)(i)**.

Variable	Data type
`MyGreeting`	
`MyInitial`	
`AgeInYears`	
`Weight`	
`Married`	

[5]

2 The following is a function design in pseudocode.

Line numbers are given for reference only.

```
01  FUNCTION StringClean(InString : STRING) RETURNS STRING
02
03     DECLARE NextChar : CHAR
04     DECLARE OutString : STRING
05     DECLARE Counter : INTEGER
06     DECLARE MyString : STRING
07
08     OutString ← ""
09
10     FOR Counter ← 1 TO LENGTH(InString)
11
12        NextChar ← MID(InString,Counter,1)
13        NextChar ← LCASE(NextChar)
14
15        IF (NextChar >= 'a') AND (NextChar <= 'z')
16
17           THEN
18
19              OutString ← OutString & NextChar
20
21        ENDIF
22
23     ENDFOR
24
25     RETURN OutString
26
27  ENDFUNCTION
```

(a) (i) This pseudocode includes features that make it easier to read and understand.

State **four** such features.

Feature 1 .....

Feature 2 .....

Feature 3 .....

Feature 4 .....

[4]

(ii) State **one** feature that could be added to make the pseudocode easier to understand.

.....[1]

(b) Study the function `StringClean()`. Identify the features of the function in the following table.

Feature	Answer
A line number containing an example of an assignment statement	
A line number containing the start of a repetition block	
A line number containing the end of a repetition block	
A line number containing the start of a selection statement	
The number of parameters of the MID function	
The Boolean operator used	
The number of local variables	
The number of function calls from within `StringClean()` resulting from the call: `NewString ← StringClean("Me")`	
The number of a line containing an unnecessary statement	

[9]

3 In a chemical factory, a procedure, `CheckSensor()` is required to allow an operator to monitor the temperature in different locations.

In the factory:

- the temperature is measured by 10 sensors, each at a different location
- each sensor has a unique ID (1 to 10).

The procedure `CheckSensor()` will compare the measured temperature against each of two constant values, `LowTemp` and `HighTemp`. It will perform the following actions depending on the result of the comparison.

Measured temperature	Action
below `LowTemp`	Output `"Cold"`
from `LowTemp` to `HighTemp`	Output `"Normal"`
above `HighTemp`	Call procedure `Alarm()`

A library function, `GetTemp()`, returns the temperature value from a given sensor.

The structured English representing the algorithm for the procedure `CheckSensor()` is as follows:

1. Prompt for the input of a sensor ID.
2. Input a sensor ID.
3. If the sensor ID is invalid, repeat from step 1.
4. Call the `GetTemp()` function with the sensor ID as the parameter, to obtain the relevant temperature.
5. Compare the temperature against the two constant values and take the appropriate action.

Draw a program flowchart on the next page to represent the algorithm for procedure `CheckSensor()`.

Variable declarations are not required in program flowcharts.

[8]

4 **(a)** A structure chart is used in modular program design.

Iteration and selection are two features of an algorithm that may be shown on a structure chart.

Give **three** other features.

Feature 1 .....

.....

Feature 2 .....

.....

Feature 3 .....

.....

[3]

(b) Pseudocode for a function is shown.

```
FUNCTION ItemProcess (AddItem, InString : STRING) RETURNS BOOLEAN
  DECLARE RetFlag : BOOLEAN
  RetFlag ← FALSE

  IF AddItem = "Yes"
     THEN
        RetFlag ← AddToList(InString)
     ELSE
        CALL RemoveFromList(InString)
  ENDIF

  RETURN RetFlag

ENDFUNCTION
```

Draw a structure chart on the next page to represent this pseudocode.

[6]

5 A golf club holds information about its members. When a member completes a round of golf, their score is stored along with their membership number and the date of the round.

(a) Explain why the club stores these data in a file rather than an array.

.....

.....[1]

(b) Editing functions such as cut, copy and paste are features provided by an Integrated Development Environment (IDE).

Give **two** additional features of an IDE that are helpful when **coding** a program.

Feature 1 .....

.....

Feature 2 .....

.....
[2]

(c) The information is stored in a text file, `ScoreDetails.txt`. The format of each line of the text file is as follows:

`<MembershipNumber><Date><Score>`

- `MembershipNumber` is a four-digit numeric string.
- `Date` is a six-digit numeric string in the format DDMMYY
- `Score` is a two-digit numeric string in the range "`50`" to "`99`".

A procedure, `AddNewScores()`, is being developed. This will allow the user to enter scores for several members on a particular date.

The procedure, `AddNewScores()`, will perform the following actions:

1. Prompt for the date of the scores.
2. Input the date of the scores.
3. Prompt for the membership number.
4. Input the membership number.
5. If the membership number is an empty string then end the procedure.
6. Prompt for the score.
7. Input the score.
8. Validate the score.
9. If the validation fails then repeat from step 6.
10. Form a text string from the data and write this to the `ScoreDetails.txt` file.
11. Repeat from step 3.

Write **program code** for the `AddNewScores()` procedure.

Visual Basic and Pascal: You should include the declaration statements for variables.
Python: You should show a comment statement for each variable used with its data type.

Programming language .....

Program code

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[9]

6 **(a)** The following pseudocode includes references to a 1D array.

```
DECLARE StudentGrade : ARRAY[1:5] OF CHAR
DECLARE n : INTEGER
DECLARE x : CHAR

n ← 3
x ← StudentGrade[n]
```

(i) Use the correct technical terms to explain the meaning of `[1:5]` in this pseudocode.

.....

.....

.....

.....[2]

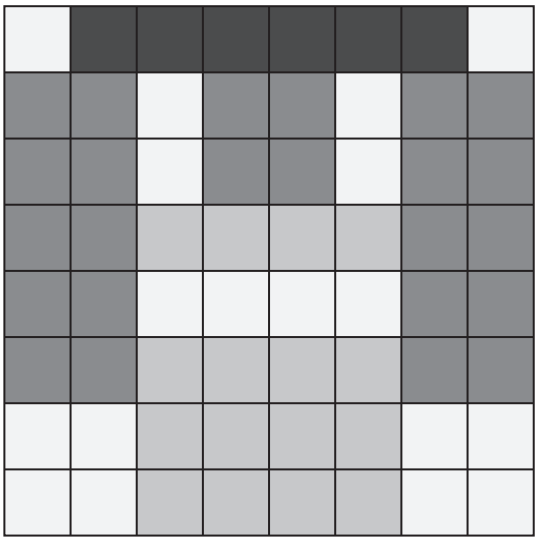
(ii) Use the correct technical term to complete the following statement.

Integer `n` is used as the ..... to `StudentGrade`. [1]

(b) A 2D array, `Picture`, contains data representing a bitmap image. Each element of the array represents one pixel of the image. The image is grey-scale encoded where the value of each pixel ranges from 0 (representing black) to 255 (representing white) with intermediate values representing different levels of grey.

The following is an example of an image and the corresponding data values for the `Picture` array.

Bitmap image



Values

240	10	10	10	10	10	10	240
80	80	240	80	80	240	80	80
80	80	240	80	80	240	80	80
80	80	150	150	150	150	80	80
80	80	240	240	240	240	80	80
80	80	150	150	150	150	80	80
240	240	150	150	150	150	240	240
240	240	150	150	150	150	240	240

In pseudocode, the array is declared as follows:

```
DECLARE Picture : ARRAY[1:8, 1:8] OF INTEGER
```

A function, `Lighten()`, is required to lighten the image. Lightening an image may cause it to 'burn out'. An image is said to be 'burnt out' if any pixel is set to the maximum value of 255.

The function `Lighten()` will:

1. increase the value of each pixel by 10%
2. return TRUE if the resultant image is 'burnt out'.

Write **pseudocode** to implement the `Lighten()` function.

Assume that the array `Picture` is a global variable.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....[8]

7 A function, `ProcessMarks()`, is required to analyse test marks for a class of students.

- There are 20 students in the class.
- A mark is between 0 and 100.
- The marks for the class are stored in an array, `Mark`, which has 20 elements.
- The array is passed to the function as a parameter.
- The function will output a message stating the average and highest marks. For example:
  `"The average mark is 34 and the highest mark is 76"`
- The function returns the subscript of the highest mark.

Write **program code** to implement the `ProcessMarks()` function.

Visual Basic and Pascal: You should include the declaration statements for variables.
Python: You should show a comment statement for each variable used with its data type.

Programming language .....

Program code

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[7]

Appendix

Built-in functions (pseudocode)

Each function returns an error if the function call is not properly formed.

`MID(ThisString : STRING, x : INTEGER, y : INTEGER) RETURNS STRING`
returns a string of length `y` starting at position `x` from `ThisString`

Example: `MID("ABCDEFGH", 2, 3)` returns string `"BCD"`

`LENGTH(ThisString : STRING) RETURNS INTEGER`
returns the integer value representing the length of string `ThisString`

Example: `LENGTH("Happy Days")` returns `10`

`LEFT(ThisString : STRING, x : INTEGER) RETURNS STRING`
returns leftmost `x` characters from `ThisString`

Example: `LEFT("ABCDEFGH", 3)` returns string `"ABC"`

`LCASE(ThisChar : CHAR) RETURNS CHAR`
returns the character value representing the lower case equivalent of `ThisChar`

If `ThisChar` is not an upper-case alphabetic character then it is returned unchanged.
Example: `LCASE('W')` returns `'w'`

`INT(x : REAL) RETURNS INTEGER`
returns the integer part of `x`

Example: `INT(27.5415)` returns `27`

`ASC(ThisChar : CHAR) RETURNS INTEGER`
returns the ASCII value of character `ThisChar`

Example: `ASC('A')` returns `65`

`MOD(ThisNum : INTEGER, ThisDiv : INTEGER) RETURNS INTEGER`
returns the integer value representing the remainder when `ThisNum` is divided by `ThisDiv`

Example: `MOD(10,3)` returns `1`

Operators (pseudocode)

Operator	Description
`&`	Concatenates (joins) two strings Example: `"Summer" & " " & "Pudding"` produces `"Summer Pudding"`
`AND`	Performs a logical `AND` on two Boolean values Example: `TRUE AND FALSE` produces `FALSE`
`OR`	Performs a logical `OR` on two Boolean values Example: `TRUE OR FALSE` produces `TRUE`

BLANK PAGE