



Cambridge International AS & A Level

CANDIDATE
NAME

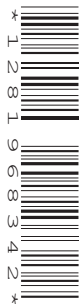
| |
|--|
| |
|--|

CENTRE
NUMBER

| | | | | |
|--|--|--|--|--|
| | | | | |
|--|--|--|--|--|

CANDIDATE
NUMBER

| | | | |
|--|--|--|--|
| | | | |
|--|--|--|--|



COMPUTER SCIENCE

9618/21

Paper 2 Fundamental Problem-solving and Programming Skills

May/June 2023

2 hours

You must answer on the question paper.

You will need: Insert (enclosed)

INSTRUCTIONS

- Answer **all** questions.
- Use a black or dark blue pen.
- Write your name, centre number and candidate number in the boxes at the top of the page.
- Write your answer to each question in the space provided.
- Do **not** use an erasable pen or correction fluid.
- Do **not** write on any bar codes.
- You may use an HB pencil for any diagrams, graphs or rough working.
- Calculators must **not** be used in this paper.

INFORMATION

- The total mark for this paper is 75.
- The number of marks for each question or part question is shown in brackets [].
- No marks will be awarded for using brand names of software packages or hardware.
- The insert contains all the resources referred to in the questions.

This document has **20** pages. Any blank pages are indicated.

Refer to the **insert** for the list of pseudocode functions and operators.

1 A programmer has written a program which includes the function `Calculate()`.
When the program is run, the function returns an unexpected value.

(a) Describe how a typical Integrated Development Environment (IDE) could be used to help debug the program to find the errors in the function `Calculate()`.

.....

 [4]

(b) The algorithm for function `Calculate()` contains the three pseudocode statements shown.
Describe the error in each statement or write 'no error' if the statement contains no error.
Assume any variables used are of the correct type for the given function.

Statement 1: `Index ← STR_TO_NUM(("27") + 2)`

Error ... Error 1: Brackets mismatch // 2/value should be added after
 brackets/function // Addition between a string and a number is not valid //
 STR_TO_NUM / the function needs to be passed a string / not an integer

Statement 2: `Index ← STR_TO_NUM(MID("CPE1704TKS", 4, 2))`

Error ... Error 2: No Error

Statement 3: `IF MONTH(ThisDate) > '6' THEN`

Error ... Error 3: MONTH () returns an integer and this is being compared with a
 character/string // Integer cannot be compared to a string // 6 should not be
 in quotes

[3]

(c) The program contains variables with values as follows:

| Variable | Value |
|----------|-------|
| Active | TRUE |
| Points | 75 |
| Exempt | FALSE |

(i) Complete the table by evaluating each expression.

| | Expression | Evaluation |
|---|---|------------|
| 1 | <code>(Points > 99) OR Active</code> | TRUE |
| 2 | <code>(Points MOD 2 = 0) OR Exempt</code> | FALSE |
| 3 | <code>(Points <= 75) AND (Active OR Exempt)</code> | TRUE |
| 4 | <code>(Active OR NOT Active) AND NOT Exempt</code> | TRUE |

[2]

(ii) Write expression 4 from the table in part (c)(i) in its simplest form.

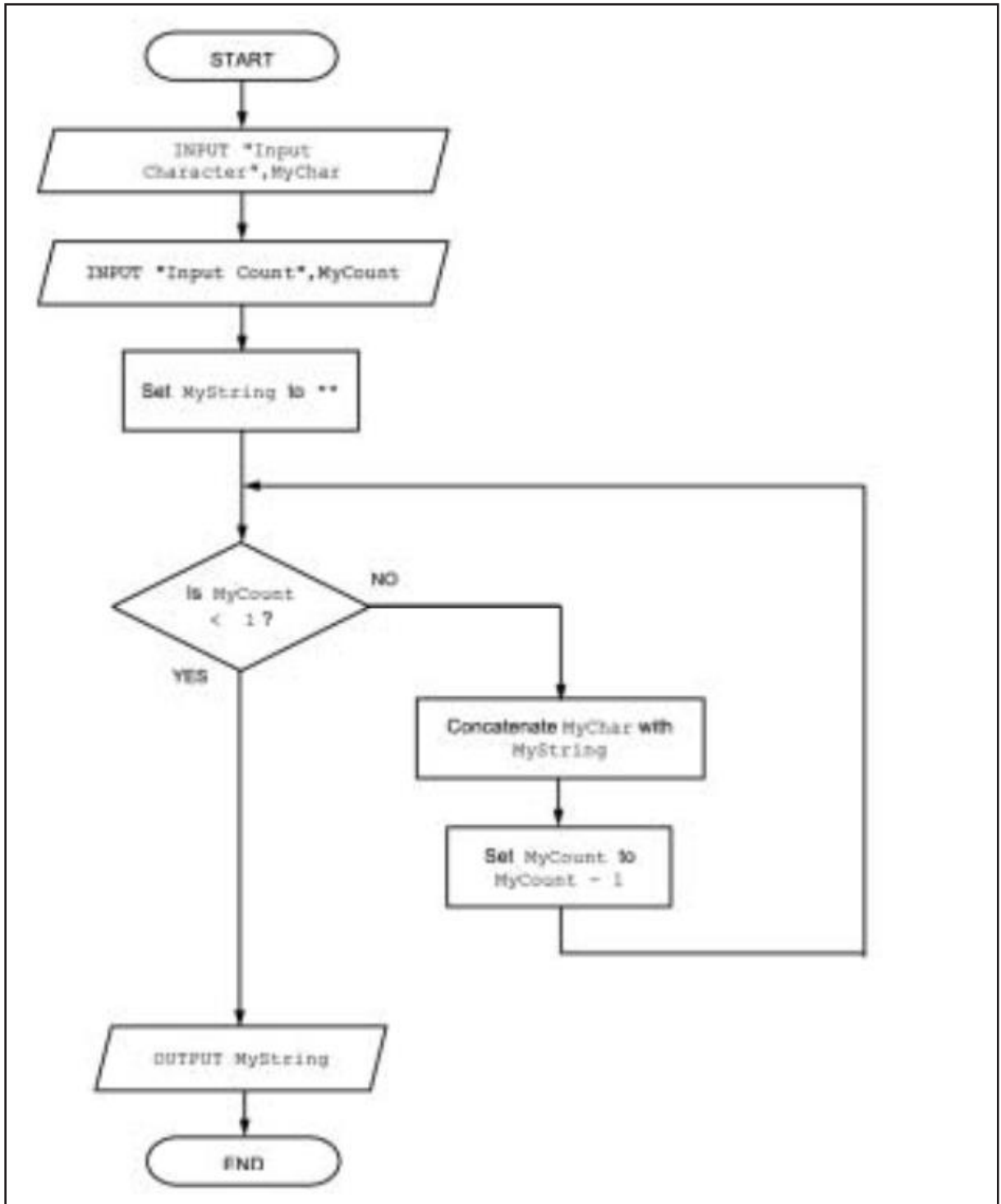
... NOT Exempt [1]

2 A program contains an algorithm to output a string of a specified length containing identical characters.

(a) The algorithm is described as follows:

1. prompt and input a character and store in `MyChar`
2. prompt and input an integer and store in `MyCount`
3. generate a string consisting of `MyChar` repeated `MyCount` times
4. output the string.

Draw a program flowchart to represent the algorithm.



[4]

(b) A different part of the program uses the variable `StartDate`.

Write pseudocode statements to declare `StartDate` **and** assign to it the date corresponding to 15/11/2005.

| | | |
|----------------|---|-------|
| Declaration .. | <code>DECLARE StartDate : DATE</code> | |
| Assignment .. | <code>StartDate ← SETDATE (15, 11, 2005)</code> | |

[3]

- 3 Customers collect points every time they make a purchase at a store.

A program is used to manage the points system and the table lists some of the information stored for one customer.

| Information | Data type required |
|----------------------------|--------------------|
| Name | String |
| Number of points collected | Integer |
| Date of birth | Date |

- (a) (i) Identify a suitable structure for storing the information for one customer. Explain the advantage of using this structure.

Structure **Record**

Advantage **• A set of data / all data related to one customer**

• of different types

• is held under a single identifier/entity

[4]

- (ii) Describe a data structure that could be used to store the information for **all** customers.

A (1D) array of records // An array of the given type could be used

[2]

- (b) Customers receive points depending on the amount they spend. The number of points depends on the band that the amount falls into:

| Band | Amount | Points |
|------|----------------------------------|--------------------------|
| 1 | Less than \$10 | 5 per whole dollar (\$) |
| 2 | Between \$10 and \$100 inclusive | 7 per whole dollar (\$) |
| 3 | Over \$100 | 10 per whole dollar (\$) |

For example, if the amount is \$99.77, this amount is in band 2 and therefore the number of points is 7×99 , which is 693 points.

The algorithm to calculate the points from a given amount is expressed as follows:

- work out the appropriate band
- calculate and output the number of points.

Apply the process of stepwise refinement to increase the detail of the algorithm. Structure your algorithm into a sequence of five steps that could be used to produce pseudocode.

Write the **five** steps.

1

| | |
|---|---|
| 1 | Reference to the use of constants or variables for the two threshold values of 10 and 100 // Input amount spent (by customer and store in a numeric variable) |
| 2 | Work out one band that amount maps to |
| 3 | Work out all bands that amount maps to |
| 4 | Calculate rounded value of amount / whole number part of amount |
| 5 | Calculate the points by multiplying the (rounded) amount by the appropriate value for appropriate band /all bands |
| 6 | Output the number of points |

.....

3

.....

.....

4

.....

.....

5

.....

.....

4 Function `Replace()` will:

1. take three parameters:
 - a string (the original string)
 - a char (the original character)
 - a char (the new character)
2. form a new string from the original string where all instances of the original character are replaced by the new character
3. return the new string.

Write pseudocode for function `Replace()`.

```

Function Replace(OldString : STRING, Char1, Char2 :
                CHAR) RETURNS : STRING
  DECLARE NewString : STRING
  DECLARE ThisChar : CHAR
  DECLARE Index : INTEGER

  NewString ← ""
  FOR Index ← 1 TO LENGTH(OldString)
    ThisChar ← MID(OldString, Index, 1)
    IF ThisChar = Char1 THEN
      ThisChar ← Char2
    ENDIF
    NewString ← NewString & ThisChar
  NEXT Index

  RETURN NewString
ENDFUNCTION

```

[6]

5 Several companies are developing websites to market a new type of games console. The company that is first to create a website that can demonstrate the interactive features of the games console will have an advantage over the others. The requirements for the website are likely to change as more information about the features of the console are made available.

One company has decided to develop their website using a program development life cycle based on the waterfall model.

(a) (i) Give **two** reasons why this may **not** be the most appropriate model to use in this case.

| | | |
|----------|---|--|
| Reason 1 | 1 | No working software until late in the life cycle so slower to market than competitors // Does not allow the creation of early versions/prototypes (which can be updated later) |
| Reason 2 | 2 | More difficult/slower to cope with changes to the requirements // website slower to be updated to reflect new requirements |
| | 3 | Needs high involvement/feedback of the stake holders /customer / client |

[2]

(ii) Identify a **more appropriate** program development life cycle model for this scenario.

..... **Iterative / Rapid Application Development / RAD** [1]

(b) The website has been running in test mode for several weeks.

Identify **and** describe a final stage of testing that should take place before the website is made available to all customers.

Stage **Beta testing**

| | | |
|-------------|---|--|
| Description | 1 | Testing carried out by a small group of (potential) users |
| | 2 | Users will check that the website/software works as required / works in the real world //User will identify errors in the website/software |
| | 3 | Users will feedback (problems) / suggestions for improvement |
| | 4 | Problems / suggestions identified are addressed (before the program is sold) |

[3]

- 6 A video-conferencing program supports up to six users. Speech from each user is sampled and digitised (converted from analogue to digital). Digitised values are stored in array `Sample`.

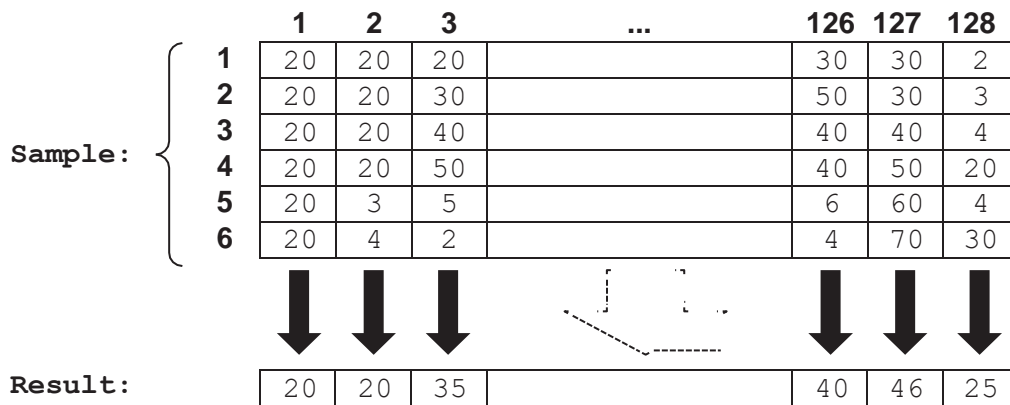
The array `Sample` consists of 6 rows by 128 columns and is of type integer. Each row contains 128 digitised sound samples from one user.

The digitised sound samples from each user are to be processed to produce a single value which will be stored in a 1D array `Result` of type integer. This process will be implemented by procedure `Mix()`.

A procedure `Mix()` will:

- calculate the average of each of the 6 sound samples in a column
- ignore sound sample values of 10 or less
- store the average value in the corresponding position in `Result`
- repeat for each column in array `Sample`

The diagram uses example values to illustrate the process:



Write pseudocode for procedure `Mix()`.

Assume `Sample` and `Result` are global.

```

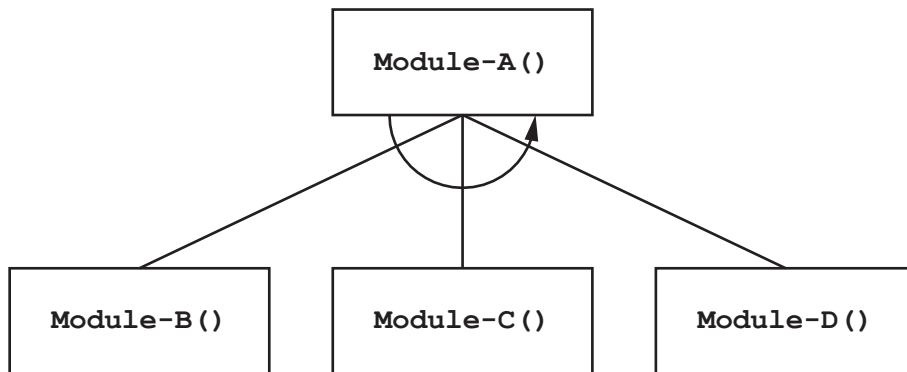
PROCEDURE Mix()
  DECLARE Count, Total ThisNum : INTEGER
  DECLARE ThisUser, ThisSample : INTEGER

  FOR ThisSample ← 1 TO 128
    Count ← 0
    Total ← 0
    FOR ThisUser ← 1 TO 6
      IF Sample[ThisUser, ThisSample] > 10 THEN
        Count ← Count + 1
        Total ← Total + Sample[ThisUser, ThisSample]
      ENDIF
    NEXT ThisUser
    Result[ThisSample] ← INT(Total / Count)
  NEXT ThisSample
ENDPROCEDURE

```

[6]

(b) The program designer produces a structure chart for the new module. Part of the structure chart is shown:



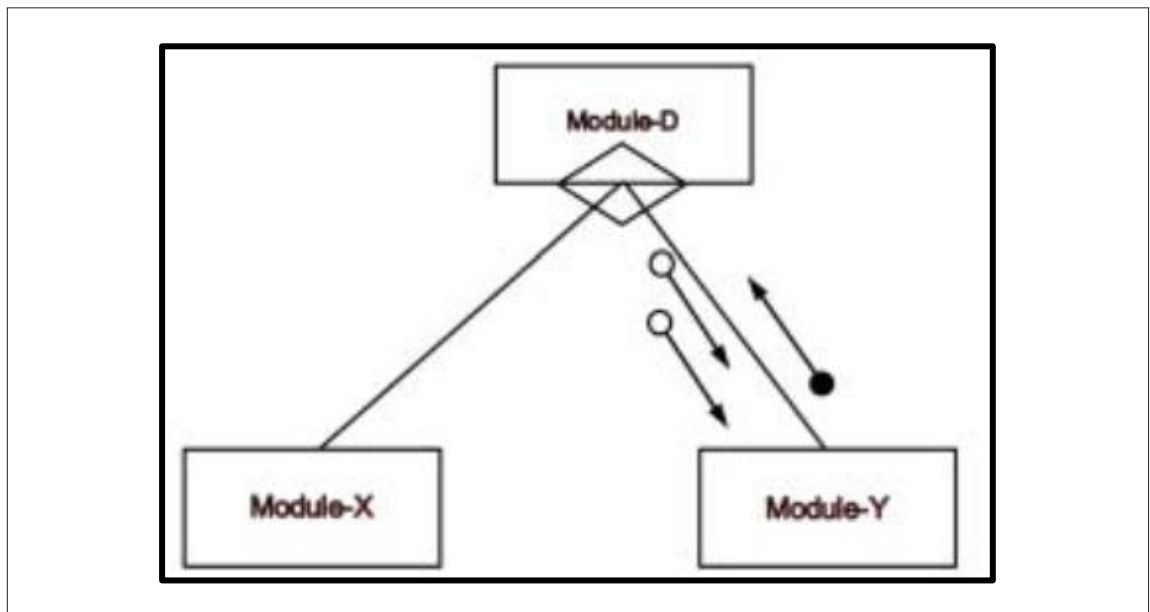
(i) Explain the relationship between the four modules shown.

- Module-A calls the other three modules
 - The process is repeated
- [2]

(ii) Two new modules are added: Module-X() and Module-Y().

- Module-X() has no parameters.
- Module-Y() will take a string and a real number as parameters and return a Boolean value.
- Module-D() will call either Module-X() or Module-Y().

Draw **only** the part of the structure chart that represents the relationship between Module-X(), Module-Y() and Module-D().



[3]

- 8 A computer shop assembles computers using items bought from several suppliers. A text file `Stock.txt` contains information about each item.

Information for each item is stored as a single line in the `Stock.txt` file in the format:

```
<ItemNum><SupplierCode><Description>
```

Item information is as follows:

| | Format | Comment |
|--------------|-------------------------|--|
| ItemNum | 4 numeric characters | unique for each item in the range "0001" to "5999" inclusive |
| SupplierCode | 5 alphabetic characters | to identify the supplier of the item |
| Description | a string | a minimum of 12 characters |

The file is organised in ascending order of `ItemNum` and does **not** contain all possible values in the range.

A programmer has started to define program modules as follows:

| Module | Description |
|--|--|
| <code>SuppExists()</code> (already written) | <ul style="list-style-type: none"> called with a parameter of type string representing a supplier code returns <code>TRUE</code> if the supplier code is already in use, otherwise returns <code>FALSE</code> |
| <code>IsNewSupp()</code> | <ul style="list-style-type: none"> called with a parameter of type string representing a new supplier code returns <code>TRUE</code> if the string only contains alphabetic characters (either upper or lower case) and the supplier code is not already in use, otherwise returns <code>FALSE</code> |

(b) A new module has been defined:

| Module | Description |
|----------------|---|
| CheckNewItem() | <ul style="list-style-type: none"> called with a parameter of type string representing a line of item information checks to see whether an item with the same <code>ItemNum</code> already exists in the file returns <code>TRUE</code> if the <code>ItemNum</code> is not already in the file, otherwise returns <code>FALSE</code> |

Write **efficient** pseudocode for module `CheckNewItem()`.

```

FUNCTION CheckNewItem(NewLine : STRING) RETURNS BOOLEAN
  DECLARE NotFound : BOOLEAN
  DECLARE NewItemNum, ThisItemNum, ThisLine : STRING

  NotFound ← TRUE

  OPENFILE "Stock.txt" FOR READ
  NewItemNum ← LEFT(NewLine, 4)
  ThisItemNum ← "0000" //rogue initial value

  WHILE NOT EOF("Stock.txt") AND NotFound = TRUE AND ___
    READFILE("Stock.txt", ThisLine) //brackets optional
    ThisItemNum ← LEFT(ThisLine, 4)
    IF ThisItemNum = NewItemNum THEN
      NotFound ← FALSE
    ENDIF
  ENDWHILE

  CLOSEFILE "Stock.txt"

  RETURN NotFound
ENDFUNCTION

```


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

(c) The program modules `SuppExists()`, `IsNewSupp()` and `CheckNewItem()` are part of a group of modules that are combined to create a complete stock control program.

Each module in the program is tested individually during development and is debugged as necessary. It is then added to the program and further testing performed.

(i) Identify this method of testing.

..... **Integration testing** [1]

(ii) One of the modules does not work properly when it is added to the program.

Describe a testing method that can be used to address this problem so that testing can continue and other modules can be added.

- A dummy/simple module is written to replace the module that does not work properly
- The dummy/simple module will return an expected value // will output a message to show it has been called

..... [2]

(d) A new module `AddItem()` will be used to add information to the `Stock.txt` file.

State the file mode that should be used for the algorithm within this module.

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

(e) A new module `FindItem()` searches for a given item in the `Stock.txt` file, which is already organised in ascending order of `ItemNum`.

Describe how this organisation may improve the efficiency of the algorithm.

- The algorithm / search / iteration can stop /only iterates
- if the current value read from the file // current line in file
- is greater than the value being searched for

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

BLANK PAGE

Permission to reproduce items where third-party owned material protected by copyright is included has been sought and cleared where possible. Every reasonable effort has been made by the publisher (UCLES) to trace copyright holders, but if any items requiring clearance have unwittingly been included, the publisher will be pleased to make amends at the earliest possible opportunity.

To avoid the issue of disclosure of answer-related information to candidates, all copyright acknowledgements are reproduced online in the Cambridge Assessment International Education Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cambridgeinternational.org after the live examination series.

Cambridge Assessment International Education is part of Cambridge Assessment. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which is a department of the University of Cambridge.