Cambridge Assessment International Education

Cambridge International AS & A Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

0910143936

COMPUTER SCIENCE **9618/21**

Paper 2 Fundamental Problem-solving and Programming Skills **May/June 2021**

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 **24** pages. Any blank pages are indicated.

© UCLES 2021

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

1 (a) A program is being developed to help manage the membership of a football club.

Complete the following identifier table.

Example value	Explanation	Variable name	Data type
`"Wong"`	The preferred name of the member joining the football club		
`FALSE`	A value to indicate whether an existing member of the club lives at the same address		
`19/02/1983`	When the member joined the football club		
`1345`	The number of points a member has earned. Members of the club earn points for different activities.		

[4]

(b) Each pseudocode statement in the following table may contain an error due to the incorrect use of the function or operator.

Describe the error in each case, or write 'NO ERROR' if the statement contains no error.

You can assume that none of the variables referenced are of an incorrect type.

Statement	Error
`Result ← 2 & 4`	
`SubString ← MID("pseudocode", 4, 1)`	
`IF x = 3 OR 4 THEN`	
`Result ← Status AND INT(x/2)`	
`Message ← "Done" + LENGTH(MyString)`	

[5]

© UCLES 2021

(c) The following data items need to be stored for each student in a group:

- student name (a string)
- test score (an integer).

State a suitable data structure and justify your answer.

Structure .....

Justification .....

.....

.....
[3]

2 (a) Four program modules form part of a program for a library.

A description of the relationship between the modules is summarised as follows:

Module name	Description
`UpdateLoan()`	• Calls either `LoanExtend()` or `LoanReturn()`
`LoanExtend()`	• Called with parameters `LoanID` and `BookID` • Calls `CheckReserve()` to see whether the book has been reserved for another library user • Returns `TRUE` if the loan has been extended, otherwise returns `FALSE`
`CheckReserve()`	• Called with `BookID` • Returns `TRUE` if the book has been reserved, otherwise returns `FALSE`
`LoanReturn()`	• Called with parameters `LoanID` and `BookID` • Returns a `REAL` (which is the value of the fine to be paid in the case of an overdue loan)

Draw a structure chart to show the relationship between the four modules and the parameters passed between them.

[5]

© UCLES 2021

(b) The definition for module `LoanReturn()` is amended as follows:

Module name	Description
`LoanReturn()`	Called with parameters `LoanID`, `BookID` and `Fine` The module code checks whether the book has been returned on time and then assigns a new value to `Fine`

- `LoanID` and `BookID` are of type `STRING`
- `Fine` is of type `REAL`

Write the pseudocode header for the **amended** module `LoanReturn()`.

.....

..... [2]

(c) A program will:

- input 50 unique integer values
- output the largest value
- output the average of the values **excluding** the largest value.

Draw a program flowchart to represent the algorithm.

Variable declarations are **not** required.

It is not necessary to check that each input value is unique.

[6]

BLANK PAGE

3 **(a)** A concert venue uses a program to calculate admission prices and store information about ticket sales.

A number of arrays are used to store data. The computer is switched off overnight and data has to be input again at the start of each day before any tickets can be sold. This process is very time consuming.

(i) Explain how the program could use text files to speed up the process.

.....

.....

.....

.....

.....

..... [2]

(ii) State the characteristic of text files that allow them to be used as explained in **part (a)(i)**.

.....

..... [1]

(iii) Information about ticket sales will be stored as a booking. The booking requires the following data:

- name of person booking
- number of people in the group (for example a family ticket or a school party)
- event type.

Suggest how data relating to each booking may be stored in a text file.

.....

.....

.....

..... [2]

© UCLES 2021

(b) A procedure `Preview()` will:

- take the name of a text file as a parameter
- output a warning message if the file is empty
- otherwise output the first five lines from the file (or as many lines as there are in the file if this number is less than five).

Write pseudocode for the procedure `Preview()`.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

4 Study the following pseudocode. Line numbers are for reference only.

```
10 FUNCTION Convert(Name : STRING) RETURNS STRING
11
12   DECLARE Flag: BOOLEAN
13   DECLARE Index : INTEGER
14   DECLARE ThisChar : CHAR
15   DECLARE NewName : STRING
16
17   CONSTANT SPACECHAR = ' '
18
19   Flag ← TRUE
20   Index ← 1
21   NewName ← ""      // formatted name string
22
23   WHILE Index <= LENGTH(Name)
24      ThisChar ← MID(Name, Index, 1)
25      IF Flag = TRUE THEN
26         NewName ← NewName & UCASE(ThisChar)
27         IF ThisChar <> SPACECHAR THEN
28            Flag ← FALSE
29         ENDIF
30      ELSE
31         NewName ← NewName & ThisChar
32      ENDIF
33      IF ThisChar = SPACECHAR THEN
34         Flag ← TRUE
35      ENDIF
36      Index ← Index + 1
37   ENDWHILE
38
39   RETURN NewName
40
41 ENDFUNCTION
```

© UCLES 2021

(a) Complete the trace table below by dry running the function when it is called as follows:

```
Result ← Convert("∇in∇a∇∇Cup")
```

Note: The symbol `'∇'` has been used to represent a space character.
Use this symbol for any space characters in the trace table.

The first row has been completed for you.

Name	Flag	Index	NewName	ThisChar
"∇in∇a∇∇Cup"				

[5]

(b) The pseudocode for `Convert()` contains a conditional loop.

State a more appropriate loop structure.

Justify your answer.

Loop structure .....

.....

Justification .....

.....

.....

[2]

(c) Two changes need to be made to the algorithm.

Change 1: Convert to lower case any character that is not the first character after a space.
Change 2: Replace multiple spaces with a single space.

(i) Change 1 may be implemented by modifying one line of the pseudocode.

Write the modified line.

.....

.....[1]

(ii) Change 2 may be implemented by moving one line of the pseudocode.

Write the number of the line to be moved and state its new position.

Line number .....

New position .....

.....

[2]

BLANK PAGE

5 A global 2D array `Result` of type `INTEGER` is used to store a list of exam candidate numbers together with their marks. The array contains 2000 elements, organised as 1000 rows and 2 columns.

Column 1 contains the candidate number and column 2 contains the mark for the corresponding candidate. All elements contain valid exam result data.

A procedure `Sort()` is needed to sort `Result` into ascending order of mark using an efficient bubble sort algorithm.

Write pseudocode for the procedure `Sort()`.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

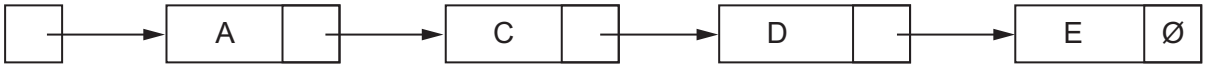
.....

.....

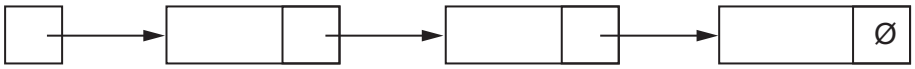
.....

..... [8]

6 The following diagram represents an Abstract Data Type (ADT) for a linked list.



The free list is as follows:



(a) Explain how a node containing data value B is added to the list in alphabetic sequence.

.....

.....

.....

.....

.....

.....

..... [4]

(b) Describe how the linked list in **part (a)** may be implemented using variables and arrays.

.....

.....

.....

..... [2]

© UCLES 2021

BLANK PAGE

7 A program is needed to take a string containing a full name and produce a new string of initials.

Some words in the full name will be ignored. For example, "the", "and", "of", "for" and "to" may all be ignored.

Each letter of the abbreviated string must be upper case.

For example:

Full name	Initials
Integrated Development Environment	IDE
The American Standard Code for Information Interchange	ASCII

The programmer has decided to use a global variable `FNString` of type `STRING` to store the full name.

It is assumed that:

- words in the full name string are separated by a single space character
- space characters will not occur at the beginning or the end of the full name string
- the full name string contains at least one word.

The programmer has started to define program modules as follows:

Module	Description
`GetStart()`	• Called with an `INTEGER` as a parameter, representing the number of a word in `FNString`. • Returns the character start position of that word in `FNString` or returns `-1` if that word does not exist • For example: if `FNString` contains the string `"hot and cold"`, `GetStart(3)` returns `9`
`GetWord()`	• Called with a parameter representing the position of the first character of a word in `FNString` • Returns the word from `FNString` • For example: if `FNString` contains the string `"hot and cold"`, `GetWord(9)` returns `"cold"`

(a) Write pseudocode for the module `GetStart()`.

.....

.....

.....

.....

.....

.....

..... [7]

(b) The programmer has decided to use a global ten-element 1D array `IgnoreList` of type `STRING` to store the ignored words. Unused elements contain the empty string (`""`) and may occur anywhere in the array.

A new module `AddWord()` is needed as follows:

Module	Description
`AddWord()`	• Called with a parameter representing a word • Stores the word in an unused element of the `IgnoreList` array and returns `TRUE` • Returns `FALSE` if the array was already full or if the word was already in the array

Write a detailed description of the algorithm for `AddWord()`. Do **not** include pseudocode statements in your answer.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

(c) As a reminder, the module description of `GetWord()` is repeated:

Module	Description
`GetWord()`	• Called with a parameter representing the position of the first character of a word in `FNString` • Returns the word from `FNString` • For example: if `FNString` contains the string `"hot and cold"`, `GetWord(9)` returns `"cold"`

Write pseudocode for the module `GetWord()`.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

BLANK PAGE

BLANK PAGE

© UCLES 2021

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 the Cambridge Assessment Group. Cambridge Assessment is the brand name of the University of Cambridge Local Examinations Syndicate (UCLES), which itself is a department of the University of Cambridge.

© UCLES 2021