



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 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.

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]

(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()	<ul style="list-style-type: none"> • Calls either LoanExtend() or LoanReturn()
LoanExtend()	<ul style="list-style-type: none"> • 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()	<ul style="list-style-type: none"> • Called with BookID • Returns TRUE if the book has been reserved, otherwise returns FALSE
LoanReturn()	<ul style="list-style-type: none"> • 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]

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

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

- `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.



- 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]

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
```


(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.

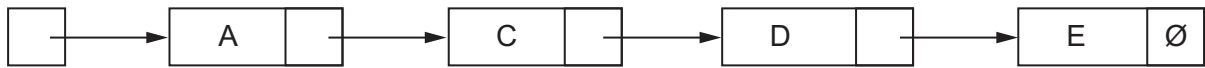
.....
..... [1]

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

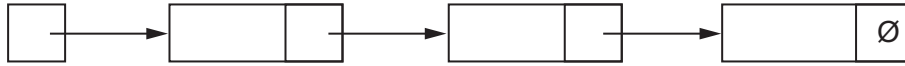
Write the number of the line to be moved and state its new position.

Line number
New position
..... [2]

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]

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
<code>GetStart()</code>	<ul style="list-style-type: none"> • Called with an <code>INTEGER</code> as a parameter, representing the number of a word in <code>FNString</code>. • Returns the character start position of that word in <code>FNString</code> or returns <code>-1</code> if that word does not exist • For example: if <code>FNString</code> contains the string "hot and cold", <code>GetStart(3)</code> returns 9
<code>GetWord()</code>	<ul style="list-style-type: none"> • Called with a parameter representing the position of the first character of a word in <code>FNString</code> • Returns the word from <code>FNString</code> • For example: if <code>FNString</code> contains the string "hot and cold", <code>GetWord(9)</code> returns "cold"

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.



Cambridge International AS & A Level

CANDIDATE
NAME

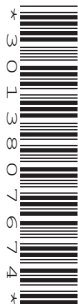
--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/22

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 **16** pages. Any blank pages are indicated.

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

- 1 (a) (i) Complete the following table by giving the appropriate data type in each case.

Variable	Example data value	Data type
Name	"Catherine"	
Index	100	
Modified	FALSE	
Holiday	25/12/2020	

[4]

- (ii) Evaluate each expression in the following table by using the initial data values shown in part (a)(i).

Expression	Evaluates to
Modified OR Index > 100	
LENGTH("Student: " & Name)	
INT(Index + 2.9)	
MID(Name, 1, 3)	

[4]

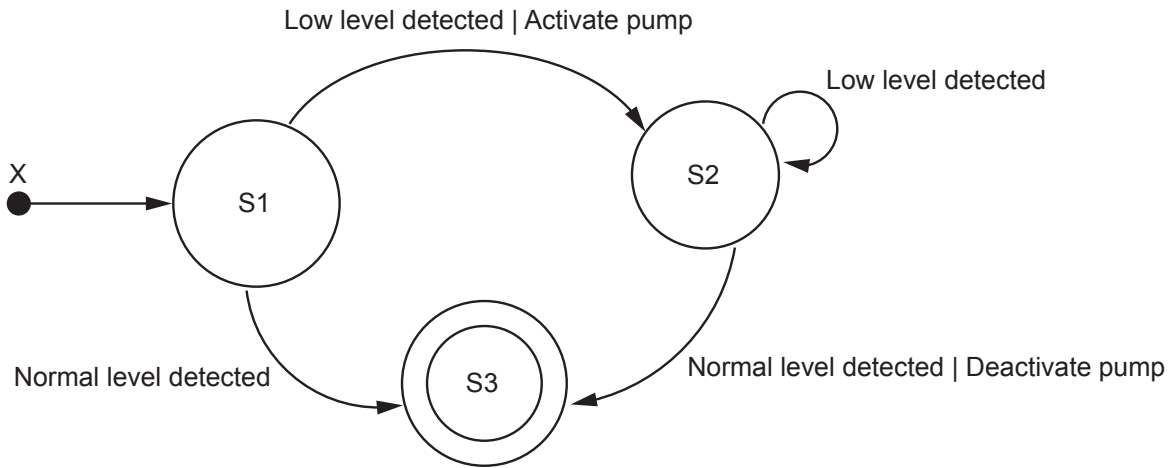
- (b) Each pseudocode statement in the following table contains an example of selection, assignment or iteration.

Put **one** tick (✓) in the appropriate column for each statement.

Statement	Selection	Assignment	Iteration
Index ← Index + 1			
IF Modified = TRUE THEN			
ENDWHILE			

[3]

2 (a) Examine the following state-transition diagram.



(i) Complete the table with reference to the diagram.

Answer

The number of transitions that result in a different state	
The number of transitions with associated outputs	
The label that should replace 'X'	
The final or halting state	

[4]

(ii) The current state is S1. The following inputs occur.

1. Low level detected
2. Low level detected
3. Low level detected
4. Low level detected

Give the number of outputs and the current state.

Number of outputs

Current state

[2]

(b) A system is being developed to help manage book loans in a library.

Registered users may borrow books from the library for a period of time.

(i) State **three** items of data that must be stored for each loan.

- 1
- 2
- 3 [2]

(ii) State **one** item of data that will be required in the library system but does not need to be stored for each loan.

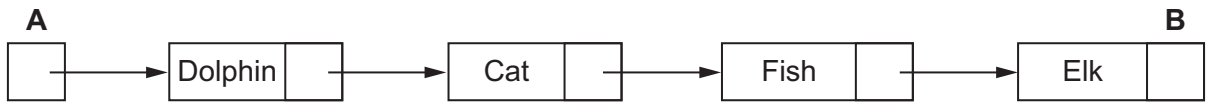
..... [1]

(iii) One operation that manipulates the data stored for each loan, would produce a list of all overdue books.

Identify **two other** operations.

- Operation 1
-
- Operation 2
- [2]

3 The following diagram represents an Abstract Data Type (ADT).



(a) Identify this type of ADT.

..... [1]

(b) Give the technical term for the item labelled **A** in the diagram.

..... [1]

(c) Give the technical term for the item labelled **B** in the diagram.

Explain the meaning of the value given to this item.

Term

Meaning

.....
.....

[2]

(d) Complete the diagram to show the ADT after the data has been sorted in alphabetical order.



[2]

(b) The following is a pseudocode function.

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
07      OutString ← ""
08
09      FOR Counter ← 1 TO LENGTH(InString)
10          NextChar ← MID(InString, Counter, 1)
11          NextChar ← LCASE(NextChar)
12          IF NOT((NextChar < 'a') OR (NextChar > 'z')) THEN
13              OutString ← OutString & NextChar
14          ENDIF
15      NEXT Counter
16
17      RETURN OutString
18
19  ENDFUNCTION

```

(i) Examine the pseudocode and complete the following table.

Answer

Give a line number containing an example of an initialisation statement.	
Give a line number containing the start of a repeating block of code.	
Give a line number containing a logic operation.	
Give the number of parameters to the function MID().	

[4]

(ii) Write a simplified version of the statement in line 12.

.....

..... [2]

BLANK PAGE

.....

.....

.....

.....

..... [8]

7 A procedure, `FormatName()`:

- is called with a string containing words and spaces as its parameter. In this context, a word is any sequence of characters that does not contain a space character.
- creates a new formatted string from this string with the following requirements:
 1. Any leading spaces removed (spaces before the first word).
 2. Any trailing spaces removed (spaces after the last word).
 3. Any multiple spaces between words converted to a single space.
 4. All characters converted to lower case.

The `FormatName()` procedure has been written in a programming language and is to be tested using the black-box method.

(a) Give a test string that could be used to show that all **four** formatting requirements have been applied correctly.

Use the symbol '∇' to represent a space character.

..... [3]

(b) The `FormatName()` procedure should assign a value to the global variable `FString`.

There is a fault in the program, which means that the assignment does not always take place.

Explain **two** ways of exposing the fault.

.....

.....

.....

.....

..... [2]

8 A program is needed to take a string containing a full name and to 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 new 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 the following global variables:

- a ten element 1D array `IgnoreList` of type `STRING` to store the ignored words
- a string `FNString` to store the full name string.

Assume that:

- each alphabetic character in the full name string may be either upper or lower case
- the full name string contains at least one word.

The programmer has started to define program modules as follows:

Module	Description
<code>GetStart()</code>	<ul style="list-style-type: none"> • Called with an <code>INTEGER</code> as its parameter, representing the number of a word in <code>FNString</code> • Returns the character start position of that word in <code>FNString</code> or returns <code>-1</code> if that word does not exist • For example: <code>GetStart(3)</code> applied to "hot and cold" returns 9
<code>GetWord()</code>	<ul style="list-style-type: none"> • Called with the position of the first character of a word in <code>FNString</code> as its parameter • Returns the word from <code>FNString</code> • For example: if <code>FNString</code> contains the string "hot and cold", <code>GetWord(9)</code> returns "cold"
<code>IgnoreWord()</code>	<ul style="list-style-type: none"> • Called with a <code>STRING</code> parameter representing a word • Searches for the word in the <code>IgnoreList</code> array • Returns <code>TRUE</code> if the word is found, otherwise returns <code>FALSE</code>
<code>GetInitials()</code>	<ul style="list-style-type: none"> • Processes the sequence of words in the full name one word at a time • Calls <code>GetStart()</code>, <code>GetWord()</code> and <code>IgnoreWord()</code> to process each word to form the new string • Outputs the new string

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.

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]

(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()	<ul style="list-style-type: none"> • Calls either LoanExtend() or LoanReturn()
LoanExtend()	<ul style="list-style-type: none"> • 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()	<ul style="list-style-type: none"> • Called with BookID • Returns TRUE if the book has been reserved, otherwise returns FALSE
LoanReturn()	<ul style="list-style-type: none"> • 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]

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

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

- `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.



- 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]

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
```


(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.

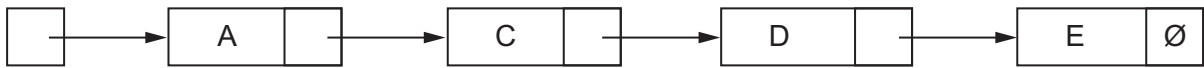
.....
..... [1]

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

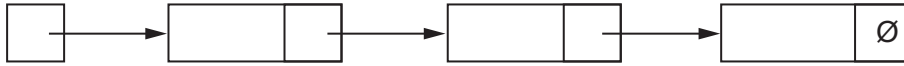
Write the number of the line to be moved and state its new position.

Line number
New position
..... [2]

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]

- 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
<code>GetStart()</code>	<ul style="list-style-type: none"> • Called with an <code>INTEGER</code> as a parameter, representing the number of a word in <code>FNString</code>. • Returns the character start position of that word in <code>FNString</code> or returns <code>-1</code> if that word does not exist • For example: if <code>FNString</code> contains the string "hot and cold", <code>GetStart(3)</code> returns 9
<code>GetWord()</code>	<ul style="list-style-type: none"> • Called with a parameter representing the position of the first character of a word in <code>FNString</code> • Returns the word from <code>FNString</code> • For example: if <code>FNString</code> contains the string "hot and cold", <code>GetWord(9)</code> returns "cold"

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.

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

- 1 (a) A programmer draws a program flowchart to show the sequence of steps required to solve a problem.

Give the technical term for a sequence of steps that describe how to solve a problem.

.....
 [1]

- (b) The table lists some of the variables used in a program.

- (i) Complete the table by writing the most appropriate data type for each variable.

Variable	Use of variable	Data type
Temp	Stores the average temperature	
PetName	Stores the name of my pet	
MyDOB	To calculate the number of days until my next birthday	
LightOn	Stores state of light; light is only on or off	

[4]

- (ii) One of the names used for a variable in the table in part 1(b)(i) is not an example of good practice.

Identify the variable and give a reason why it is **not** good practice to use that name.

Variable

Reason

.....

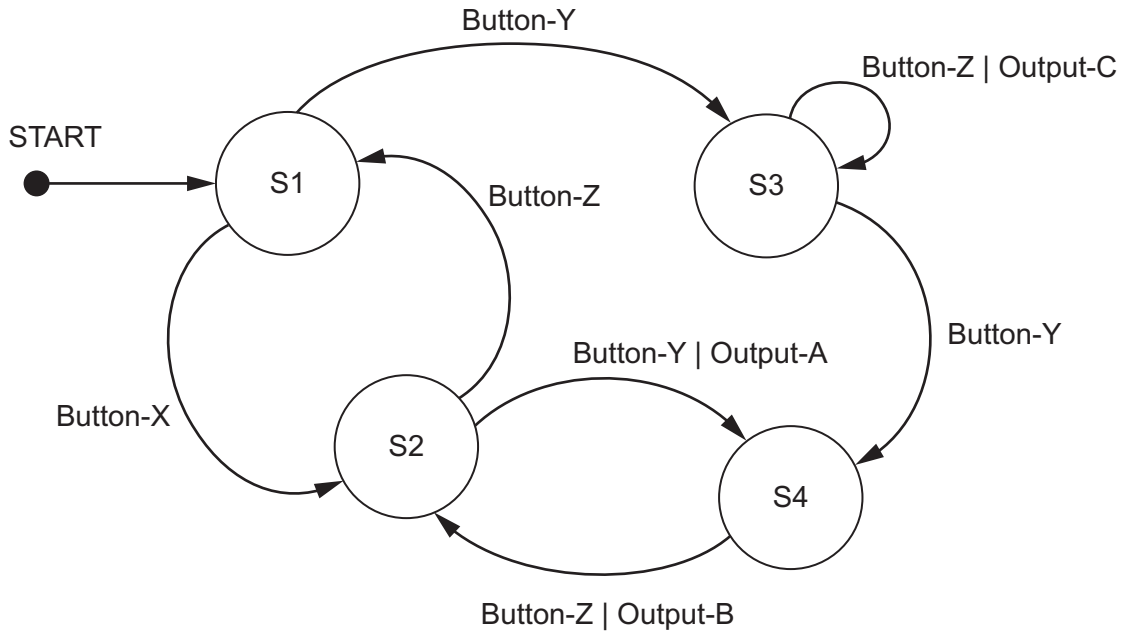
[2]

- (c) Complete the table by evaluating each expression.

Expression	Evaluation
<code>INT((31 / 3) + 1)</code>	
<code>MID(TO_UPPER("Version"), 4, 2)</code>	
<code>TRUE AND (NOT FALSE)</code>	
<code>NUM_TO_STR(27 MOD 3)</code>	

[4]

2 Examine the following state-transition diagram.



(a) Complete the table with reference to the diagram.

Answer

The number of different inputs	
The number of different outputs	
The single input value that could result in S4	

[3]

(b) The initial state is S1.

Complete the table to show the inputs, outputs and next states.

Input	Output	Next state
Button-Y		
	none	
Button-Z		S2
	none	

[4]

3 The manager of a cinema wants a program to allow users to book seats. The cinema has several screens. Each screen shows a different film.

(a) Decomposition will be used to break the problem down into sub-problems.

Describe **three** program modules that could be used in the design.

Module 1

.....

.....

Module 2

.....

.....

Module 3

.....

.....

[3]

(b) Two types of program modules may be used in the design of the program.

Identify the type of program module that should be used to return a value.

..... [1]

- 4 A stack is created using a high-level language. Memory locations 200 to 207 are to be used to store the stack.

The following diagram represents the current state of the stack.

TopOfStack points to the last value added to the stack.

Stack		Pointer
Memory location	Value	
200		
201		
202		
203	'F'	← TopOfStack
204	'C'	
205	'D'	
206	'E'	
207	'H'	

- (a) Complete the following table by writing the answers.

	Answer
The value that has been on the stack for the longest time.	
The memory location pointed to by TopOfStack if three POP operations are performed.	

[2]

(b) The following diagram shows the current state of the stack:

Stack		Pointer
Memory location	Value	
200		
201		
202	'W'	← TopOfStack
203	'Y'	
204	'X'	
205	'Z'	
206	'N'	
207	'P'	

The following operations are performed:

POP
 POP
 PUSH 'A'
 PUSH 'B'
 POP
 PUSH 'C'
 PUSH 'D'

Complete the diagram to show the state of the stack **after** the operations have been performed.

Stack		Pointer
Memory location	Value	
200		
201		
202		
203		
204		
205		
206		
207		

[4]

7 A string is a palindrome if it reads the same forwards as backwards.

The following strings are examples of palindromes:

- "Racecar"
- "madam"
- "12344321"

Upper-case and lower-case characters need to be treated the same. For example, 'A' is equivalent to 'a'.

(a) A function `IsPalindrome()` will take a string parameter. The function will return `TRUE` if the string is a palindrome and will return `FALSE` if the string is not a palindrome.

Write pseudocode for `IsPalindrome()`.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

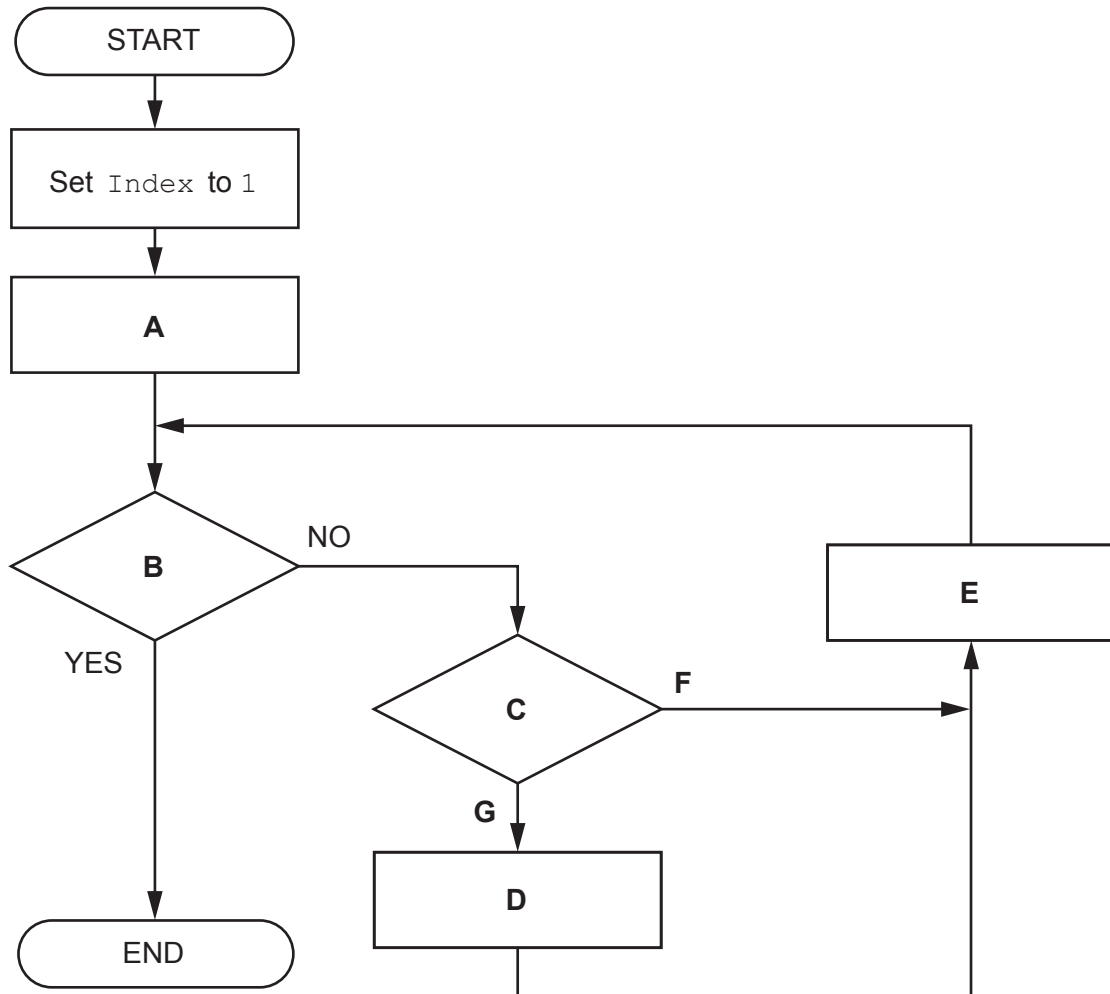
.....

..... [7]

(b) Strings may consist of several words separated by spaces.

For example, the string "never odd or even" becomes a palindrome if the spaces are removed.

The program flowchart represents an algorithm to produce a string `OutString` by removing all spaces from a string `InString`.



Complete the table by writing the text that should replace each of the labels **B**, **C**, **D**, **F** and **G**.

Note: the text may be written as a pseudocode statement.

Label	Text
A	Set <code>OutString</code> to ""
B	
C	
D	
E	Set <code>Index</code> to <code>Index + 1</code>
F	
G	

[4]

- 8 A program allows a user to save passwords used to login to websites. A stored password is inserted automatically when the user logs into the corresponding website.

A student is developing a program to generate a password. The password will be of a fixed format, consisting of **three groups of four** alphanumeric characters. The groups are separated by the hyphen character '-'.

An example of a password is: "FxAf-3haV-Tq49"

A global 2D array `Secret` of type `STRING` stores the passwords together with the website domain name where they are used. `Secret` contains 1000 elements organised as 500 rows by 2 columns.

Unused elements contain the empty string (""). These may occur anywhere in the array.

An example of a part of the array is:

Array element	Value
<code>Secret[27, 1]</code>	"www.thiswebsite.com"
<code>Secret[27, 2]</code>	"....."
<code>Secret[28, 1]</code>	"www.thatwebsite.com"
<code>Secret[28, 2]</code>	"....."

Note:

- For security, passwords are stored in an encrypted form, shown as "....." in the example.
- The passwords cannot be used without being decrypted.
- Assume that the encrypted form of a password will **not** be an empty string.

The programmer has started to define program modules as follows:

Module	Description
<code>RandomChar()</code>	<ul style="list-style-type: none"> • Generates a single random character from within one of the following ranges: <ul style="list-style-type: none"> ○ 'a' to 'z' ○ 'A' to 'Z' ○ '0' to '9' • Returns the character
<code>Encrypt()</code>	<ul style="list-style-type: none"> • Takes a password as a parameter of type string • Returns the encrypted form of the password as a string
<code>Decrypt()</code>	<ul style="list-style-type: none"> • Takes an encrypted password as a parameter of type string • Returns the decrypted form of the password as a string

For reference, relevant ASCII values are as follows:

Character range	ASCII range
'a' to 'z'	97 to 122
'A' to 'Z'	65 to 90
'0' to '9'	48 to 57

(a) Write pseudocode for module `RandomChar()`.

You may wish to refer to the **insert** for a description of the `CHR()` function. Other functions may also be required.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

..... [6]

(b) A new module is defined as follows:

Module	Description
<code>FindPassword()</code>	<ul style="list-style-type: none">• Takes a website domain name as a parameter of type string• Searches for the website domain name in the array <code>Secret</code>• If the website domain name is found, the decrypted password is returned• If the website domain name is not found, a warning message is output, and an empty string is returned

Write pseudocode for module `FindPassword()`.

Assume that modules `Encrypt()` and `Decrypt()` have already been written.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

[7]

(c) The modules `Encrypt()` and `Decrypt()` are called from several places in the main program.

Identify a method that could have been used to test the main program before these modules were completed. Describe how this would work.

Method

Description

.....

.....

..... [3]

(d) A validation function is written to check that the passwords generated are valid.

To be valid, each password must:

- be 14 characters long
- be organised as three groups of four case-sensitive alphanumeric characters. The groups are separated by hyphen characters
- not include any duplicated characters, except for the hyphen characters.

Note: lower-case and upper-case characters are not the same. For example, 'a' is not the same as 'A'.

Give **two** password strings that could be used to test different areas of the validation rules.

Password 1

Password 2

[2]

(e) The `RandomChar()` module is to be modified so that alphabetic characters are generated twice as often as numeric characters.

Describe how this might be achieved.

.....

.....

.....

.....

.....

..... [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.



Cambridge International AS & A Level

CANDIDATE
NAME

--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/22

Paper 2 Fundamental Problem-solving and Programming Skills

May/June 2022

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) A programmer is testing a program using an Integrated Development Environment (IDE). The programmer wants the program to stop when it reaches a specific instruction or program statement in order to check the value assigned to a variable.

Give the technical term for the position at which the program stops.

..... [1]

- (b) The following table lists some activities from the program development life cycle.

Complete the table by writing the life cycle stage for each activity.

Activity	Life cycle stage
An identifier table is produced.	
Syntax errors can occur.	
The developer discusses the program requirements with the customer.	
A trace table is produced.	

[4]

- (c) An identifier table includes the names of identifiers used.

State **two other** pieces of information that the identifier table should contain.

1

2

[2]

- (d) The pseudocode statements in the following table may contain errors.

State 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.

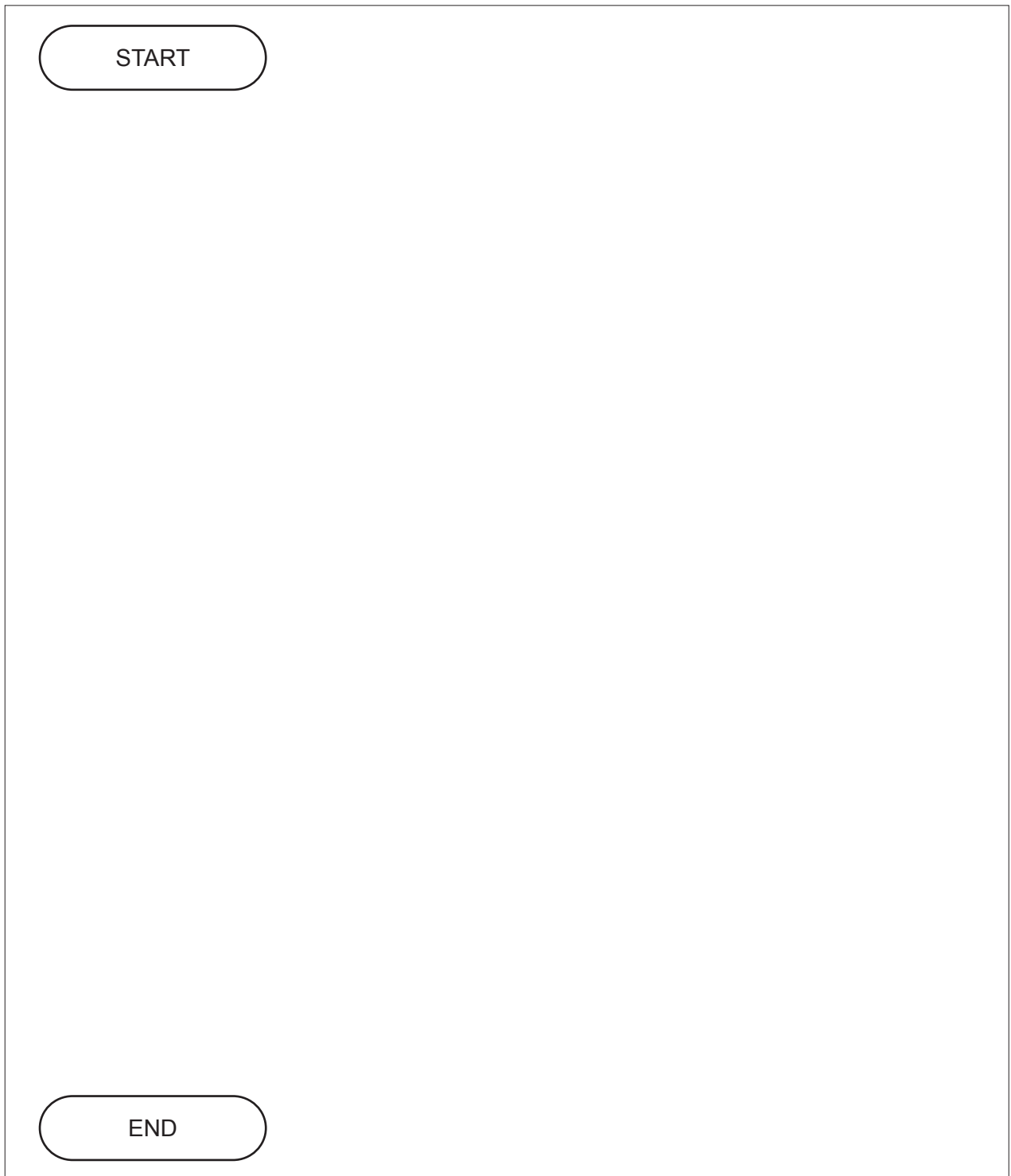
Statement	Error
Status ← TRUE AND FALSE	
IF LENGTH("Password") < "10" THEN	
Code ← LCASE("Electrical")	
Result ← IS_NUM(-27.3)	

[4]

2 An algorithm is described as follows:

1. Input an integer value.
2. Jump to step 6 if the value is less than zero.
3. Call the function `IsPrime()` using the integer value as a parameter.
4. Keep a count of the number of times function `IsPrime()` returns `TRUE`.
5. Repeat from step 1.
6. Output the value of the count with a suitable message.

Draw a program flowchart to represent the algorithm.



[4]

- 3 (a) The module headers for five modules in a program are defined in pseudocode as follows:

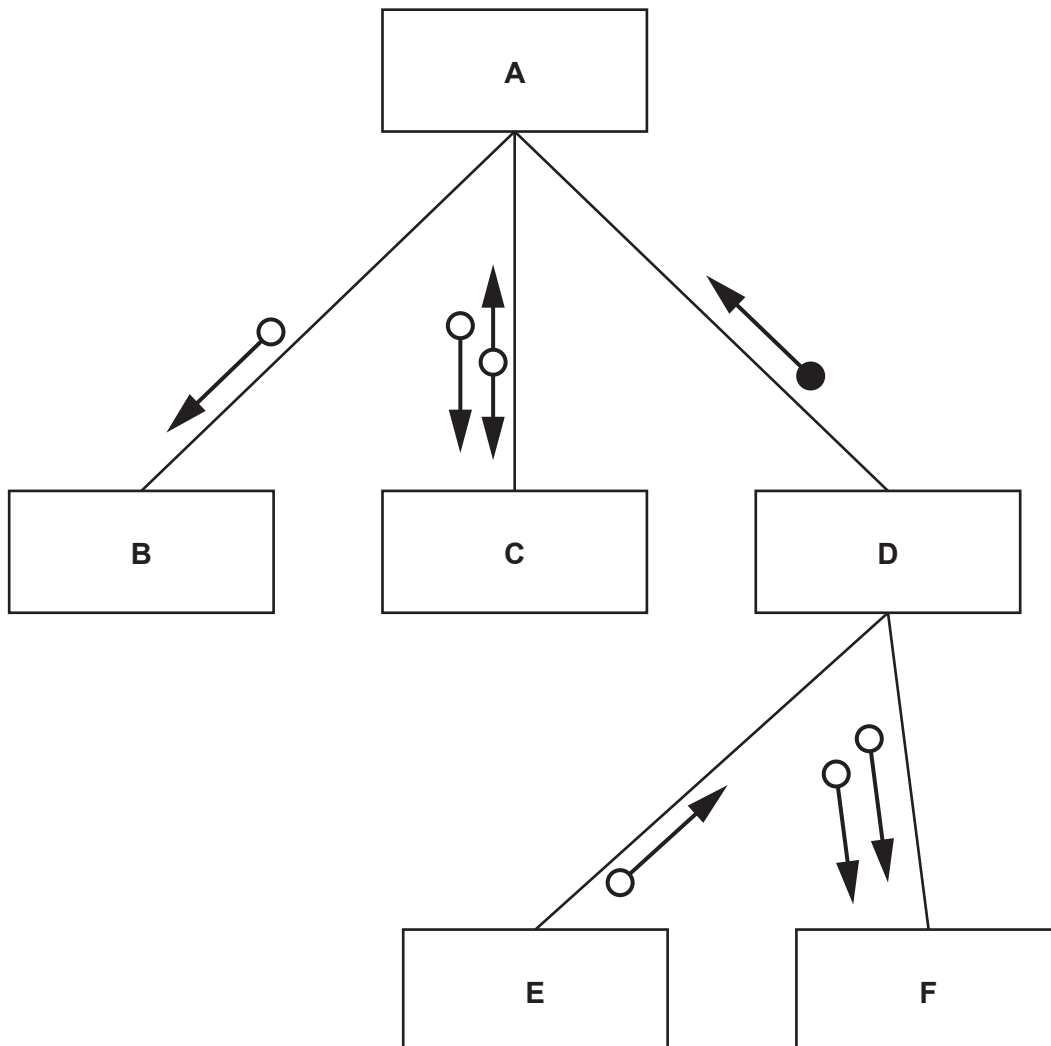
Pseudocode module header
FUNCTION Mod_V(S2 : INTEGER) RETURNS BOOLEAN
PROCEDURE Mod_W(P4 : INTEGER)
PROCEDURE Mod_X(T4 : INTEGER, BYREF P3 : REAL)
PROCEDURE Mod_Y(W3 : REAL, Z8 : INTEGER)
FUNCTION Mod_Z(F3 : REAL) RETURNS INTEGER

An additional module `Head()` repeatedly calls three of the modules in sequence.

A structure chart has been partially completed.

- (i) Complete the structure chart to include the information given about the six modules.

Do **not** label the parameters and do **not** write the module names.



[3]

- (ii) Complete the table using the information in **part 3(a)** by writing each module name to replace the labels **A** to **F**.

Label	Module name
A	
B	
C	
D	
E	
F	

[3]

- (b) The structure chart represents part of a complex problem. The process of decomposition is used to break down the complex problem into sub-problems.

Describe **three** benefits of this approach.

1

.....

2

.....

3

.....

[3]

(b) The algorithm in **part (a)** is to be amended. The calling program will pass the number of lines to be output as well as the name of the text file.

The number of lines could be any value from 1 to 30.

It can be assumed that the file contains **at least** the number of lines passed.

Outline **three** changes that would be needed.

- 1
-
-
- 2
-
-
- 3
-
-

[3]

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

```

10 PROCEDURE Encode()
11   DECLARE CountA, CountB, ThisNum : INTEGER
12   DECLARE ThisChar : CHAR
13   DECLARE Flag : BOOLEAN
14   CountA ← 0
15   CountB ← 10
16   Flag ← TRUE
17   INPUT ThisNum
18   WHILE ThisNum <> 0
19     ThisChar ← LEFT(NUM_TO_STR(ThisNum), 1)
20     IF Flag = TRUE THEN
21       CASE OF ThisChar
22         '1' : CountA ← CountA + 1
23         '2' : IF CountB < 10 THEN
24             CountA ← CountA + 1
25             ENDIF
26         '3' : CountB ← CountB - 1
27         '4' : CountB ← CountB - 1
28             Flag ← FALSE
29         OTHERWISE : OUTPUT "Ignored"
30       ENDCASE
31     ELSE
32       IF CountA > 2 THEN
33         Flag ← NOT Flag
34         OUTPUT "Flip"
35       ELSE
36         CountA ← 4
37       ENDIF
38     ENDIF
39     INPUT ThisNum
40   ENDWHILE
41   OUTPUT CountA
42 ENDPROCEDURE

```

(a) Procedure `Encode()` contains a loop structure.

Identify the type of loop **and** state the condition that ends the loop.

Do **not** include pseudocode statements in your answer.

Type

Condition

.....

[2]

(b) Complete the trace table below by dry running the procedure `Encode()` when the following values are input:

12, 24, 57, 43, 56, 22, 31, 32, 47, 99, 0

The first row is already complete.

ThisNum	ThisChar	CountA	CountB	Flag	OUTPUT
		0	10	TRUE	

[6]

(c) Procedure `Encode()` is part of a modular program. Integration testing is to be carried out on the program.

Describe **integration testing**.

.....

.....

.....

..... [2]

6 A string represents a series of whole numbers, separated by commas.

For example:

"12,13,451,22"

Assume that:

- the comma character ',' is used as a separator
- the string contains only the characters '0' to '9' and the comma character ','.

A procedure `Parse` will:

- take the string as a parameter
- extract each number in turn
- calculate the total value and average value of all the numbers
- output the total and average values with a suitable message.

Write pseudocode for the procedure.

```
PROCEDURE Parse(InString : STRING)
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

ENDPROCEDURE

[7]

- 8 A program allows a user to save passwords used to log in to websites. A stored password is then inserted automatically when the user logs in to the corresponding website.

A global 2D array `Secret` of type `STRING` stores the passwords together with the website domain name where they are used. `Secret` contains 1000 elements organised as 500 rows by 2 columns.

Unused elements contain the empty string (`""`). These may occur anywhere in the array.

An example of a part of the array is:

Array element	Value
<code>Secret[27, 1]</code>	<code>"thiswebsite.com"</code>
<code>Secret[27, 2]</code>	<code>"....."</code>
<code>Secret[28, 1]</code>	<code>"thatwebsite.com"</code>
<code>Secret[28, 2]</code>	<code>"....."</code>

Note:

- For security, the passwords are stored in an encrypted form, shown as `"....."` in the example.
- The passwords cannot be used without being decrypted.
- You may assume that the encrypted form of a password will **NOT** be an empty string.

The programmer has started to define program modules as follows:

Module	Description
<code>Exists()</code>	<ul style="list-style-type: none"> • Takes two parameters: <ul style="list-style-type: none"> ◦ a string ◦ a character • Performs a case-sensitive search for the character in the string • Returns <code>TRUE</code> if the character occurs in the string, otherwise returns <code>FALSE</code>
<code>Encrypt()</code>	<ul style="list-style-type: none"> • Takes a password as a parameter of type string • Returns the encrypted form of the password as a string
<code>Decrypt()</code>	<ul style="list-style-type: none"> • Takes an encrypted password as a parameter of type string • Returns the decrypted form of the password as a string

Note: in a case-sensitive comparison, 'a' is not the same as 'A'.

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

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

..... [5]

(b) A new module `SearchDuplicates()` will:

- search for the **first** password that occurs more than once in the array and output a message each time a duplicate is found.

For example, if the same password was used for the three websites `ThisWebsite.com`, `website27.net` and `websiteZ99.org`, then the following messages will be output:

```
"Password for ThisWebsite.com also used for website27.net"  
"Password for ThisWebsite.com also used for websiteZ99.org"
```

- end once all messages have been output.

The module will output a message if no duplicates are found.
For example:

```
"No duplicate passwords found"
```

Write efficient pseudocode for the module `SearchDuplicates()`. `Encrypt()` and `Decrypt()` functions have been written.

Note: It is necessary to decrypt each password before checking its value.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

- (c) A password has a fixed format, consisting of **three groups of four** alphanumeric characters, separated by the hyphen character '-'.

An example of a password is:

"FxAf-3haV-Tq49"

Each password must:

- be 14 characters long
- be organised as three groups of four alphanumeric characters. The groups are separated by hyphen characters
- not include any duplicated characters, except for the hyphen characters.

An algorithm is needed for a new function `GeneratePassword()`, which will generate and return a password in this format.

Assume that the following modules have already been written:

Module	Description
<code>Exists()</code>	<ul style="list-style-type: none"> • Takes two parameters: <ul style="list-style-type: none"> ○ a string ○ a character • Performs a case-sensitive search for the character in the string • Returns <code>TRUE</code> if the character occurs in the string, otherwise returns <code>FALSE</code>
<code>RandomChar()</code>	<ul style="list-style-type: none"> • Generates a single random character from within one of the following ranges: <ul style="list-style-type: none"> ○ 'a' to 'z' ○ 'A' to 'Z' ○ '0' to '9' • Returns the character

Note: in a case-sensitive comparison, 'a' is not the same as 'A'.

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.



Cambridge International AS & A Level

CANDIDATE
NAME

CENTRE
NUMBER

--	--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--	--



COMPUTER SCIENCE

9618/23

Paper 2 Fundamental Problem-solving and Programming Skills

May/June 2022

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) The following table contains pseudocode examples.

Each example may include all or part of:

- selection
- iteration (repetition)
- assignment.

Complete the table by placing **one or more** ticks (✓) in each row.

Pseudocode example	Selection	Iteration	Assignment
FOR Index ← 1 TO 3 Safe[Index] ← GetResult() NEXT Index			
OTHERWISE : OUTPUT "ERROR 1202"			
REPEAT UNTIL Index = 27			
INPUT MyName			
IF Mark > 74 THEN Grade ← 'A' ENDIF			

[5]

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

Variable	Value
AAA	TRUE
BBB	FALSE
Count	99

Complete the table by evaluating each expression.

Expression	Evaluation
AAA AND (Count > 99)	
AAA AND (NOT BBB)	
(Count <= 99) AND (AAA OR BBB)	
(BBB AND Count > 50) OR NOT AAA	

[2]

(ii) Give an example of when a variable of type Boolean would be used.

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

2 A program has been written to implement a website browser and maintenance is now required.

One type of maintenance is called perfective.

Name **two other** types of maintenance that the program may require **and** give a reason for each.

Type 1

Reason

.....

.....

.....

Type 2

Reason

.....

.....

.....

[4]

3 Four program modules are defined as follows:

Pseudocode module header
PROCEDURE Sub1_A(XT : INTEGER, PB : STRING)
FUNCTION Sub1_B(RA : INTEGER) RETURNS BOOLEAN
PROCEDURE Sub1_C(SB : INTEGER, BYREF SA : STRING)
PROCEDURE Section_1()

(a) A structure chart will be produced as part of the development process.

Describe the purpose of a structure chart.

.....

.....

.....

..... [2]

(b) Module `Section_1()` calls one of the other three modules. The module called will be selected when the program runs.

Draw the structure chart.

[5]

- 4 Items in a factory are weighed automatically. The weight is stored as an integer value representing the item weight to the nearest gram (g).

A function is written to validate the weight of each item. The function will return "PASS" if the weight of the item is within the acceptable range, otherwise the function will return "FAIL".

The acceptable weight range for an item is 150g to 155g inclusive.

The validation function is to be properly tested. Black-box testing will be used and a test plan needs to be produced.

Complete the table by writing additional tests to test this function.

Type of test data	Example test value	Expected return value	Explanation
Normal	153	"PASS"	Value within the acceptable range

[4]

- 5 A program will store attendance data about each employee of a company.

The data will be held in a record structure of type `Employee`. The fields that will be needed are as shown:

Field	Typical value	Comment
<code>EmployeeNumber</code>	123	A numeric value starting from 1
<code>Name</code>	"Smith, Eric"	Format: <last name>', <first name>
<code>Department</code>	"1B"	May contain letters and numbers
<code>Born</code>	13/02/2006	Must not be before 04/02/1957
<code>Attendance</code>	97.40	Represents a percentage

- (a) (i) Write pseudocode to declare the record structure for type `Employee`.

.....

 [4]

- (ii) A 1D array `Staff` containing 500 elements will be used to store the employee records.

Write pseudocode to declare the `Staff` array.

.....
 [2]

- (b) There may be more records in the array than there are employees in the company. In this case, some records of the array will be unused.

- (i) State why it is good practice to have a standard way to indicate unused array elements.

.....
 [1]

- (ii) Give **one** way of indicating an unused record in the `Staff` array.

.....
 [1]

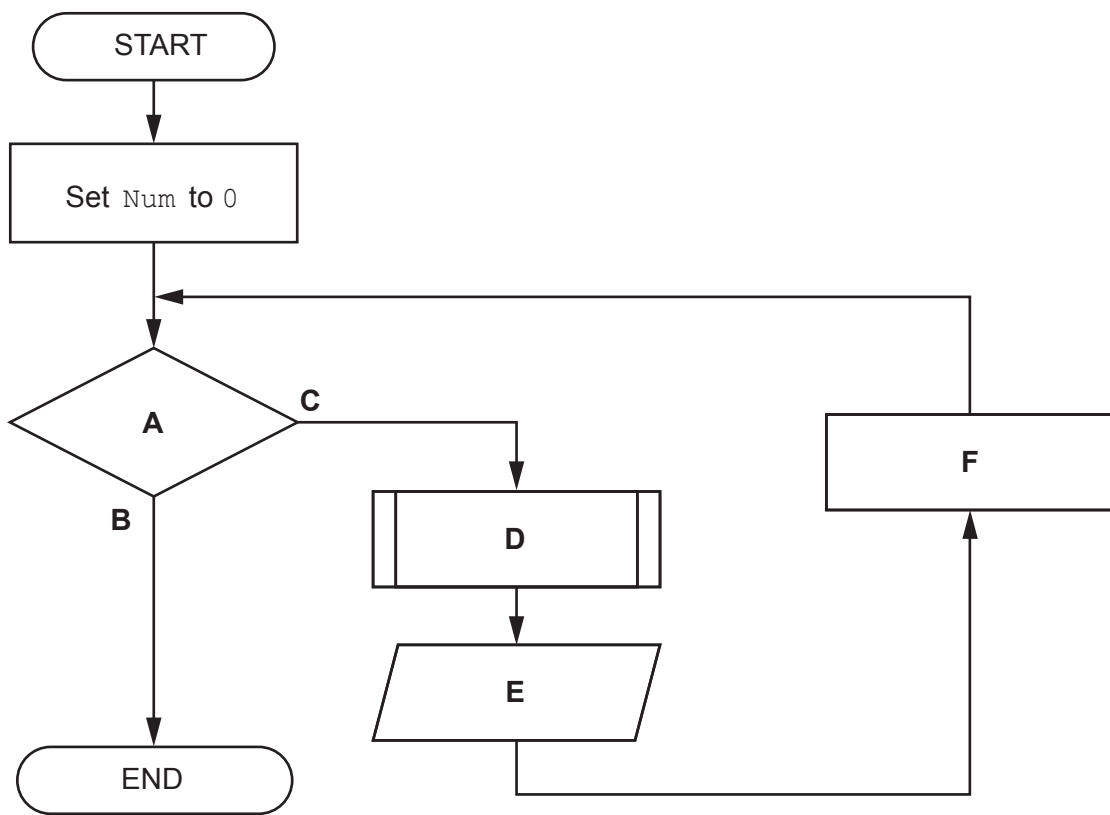
(b) A procedure `FirstTen()` will output the factorial of the numbers from 0 to 9. The procedure will use the function from **part (a)**.

The required output is:

```

Factorial of 0 is 1
Factorial of 1 is 1
Factorial of 2 is 2
    )
Factorial of 9 is 362880
    
```

The program flowchart represents an algorithm for `FirstTen()`.



Complete the table by writing the text that should replace each label **A** to **F**.

Label	Text
A	
B	
C	
D	
E	
F	

7 The following pseudocode represents an algorithm intended to output the last three lines as they appear in a text file. Line numbers are provided for reference only.

```

10 PROCEDURE LastLines(ThisFile : STRING)
11     DECLARE ThisLine : STRING
12     DECLARE Buffer : ARRAY[1:3] OF STRING
13     DECLARE LineNum : INTEGER
14     LineNum ← 1
15     OPENFILE ThisFile FOR READ
16
17     WHILE NOT EOF(ThisFile)
18         READFILE Thisfile, ThisLine // read a line
19         Buffer[LineNum] ← ThisLine
20         LineNum ← LineNum + 1
21         IF LineNum = 4 THEN
22             LineNum ← 1
23         ENDIF
24     ENDWHILE
25
26     CLOSEFILE ThisFile
27     FOR LineNum ← 1 TO 3
28         OUTPUT Buffer[LineNum]
29     NEXT LineNum
30 ENDPROCEDURE
    
```

(a) There is an error in the algorithm. In certain cases, a text file will have at least three lines but the output will be incorrect.

(i) State how the output may be incorrect.

.....
 [1]

(ii) Describe the error in the algorithm **and** explain how it may be corrected.

Description

.....

.....

.....

Explanation

.....

.....

.....

[4]

- (b) The original algorithm is implemented and sometimes the last three lines of the text file are output correctly.

State the condition that results in the correct output.

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

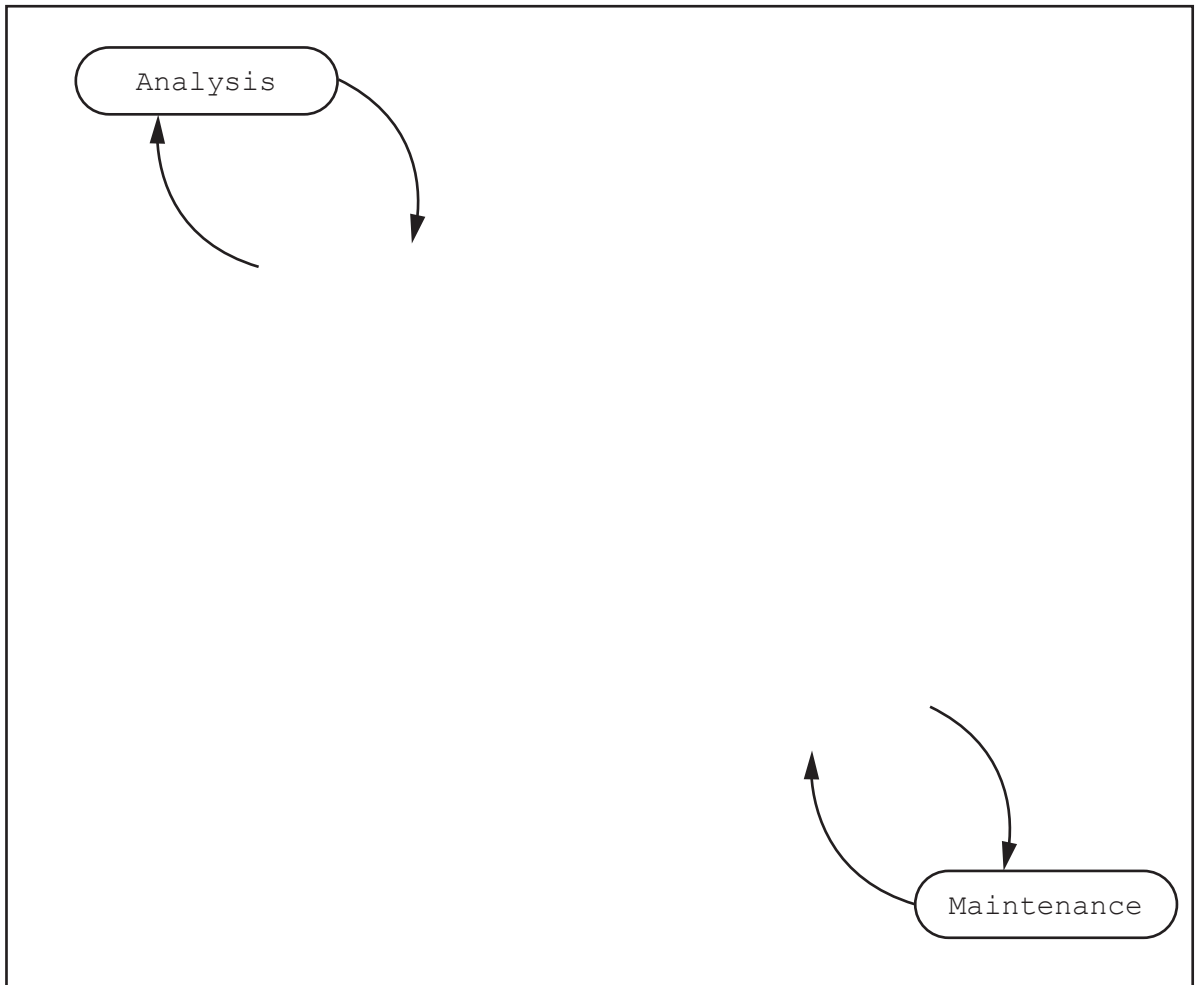
- (c) Lines 20 to 23 inclusive could be replaced with a single pseudocode statement.

Write the pseudocode statement.

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

8 The following diagram shows the incomplete waterfall model of the program development life cycle.

(a) Complete the diagram.



[3]

(b) Explain the meaning of the downward and upward arrows.

Downward arrows

.....

.....

Upward arrows

.....

.....

[2]

(c) Identify another type of model for the program development life cycle.

..... [1]

BLANK PAGE

- 9 A program allows a user to save passwords used to log in to websites. A stored password is then inserted automatically when the user logs in to the corresponding website.

A student is developing a program to generate a strong password. The password will be of a fixed format, consisting of **three groups of four** alphanumeric characters, separated by the hyphen character '-'.
 An example of a password is: "FxAf-3hzV-Aq49"

An example of a password is: "FxAf-3hzV-Aq49"

A valid password:

- must be 14 characters long
- must be organised as three groups of four alphanumeric characters. The groups are separated by hyphen characters
- may include duplicated characters, **provided** these appear in different groups.

The programmer has started to define program modules as follows:

Module	Description
RandomChar()	<ul style="list-style-type: none"> • Generates a single random character from within one of the following ranges: <ul style="list-style-type: none"> ○ 'a' to 'z' ○ 'A' to 'Z' ○ '0' to '9' • Returns the character
Exists()	<ul style="list-style-type: none"> • Takes two parameters: <ul style="list-style-type: none"> ○ a string ○ a character • Performs a case-sensitive search for the character in the string • Returns TRUE if the character occurs in the string, otherwise returns FALSE
Generate()	<ul style="list-style-type: none"> • Generates a valid password • Uses RandomChar() and Exists() • Returns the password

Note: in a case-sensitive comparison, 'a' is not the same as 'A'.

- (b) A global 2D array `Secret` of type `STRING` stores the passwords together with the website domain name where they are used. `Secret` contains 1000 elements organised as 500 rows by 2 columns.

Unused elements contain the empty string (`""`). These may occur anywhere in the array.

An example of part of the array is:

Array element	Value
<code>Secret[27, 1]</code>	<code>"www.thiswebsite.com"</code>
<code>Secret[27, 2]</code>	<code>"●●●●●●●●●●"</code>
<code>Secret[28, 1]</code>	<code>"www.thatwebsite.com"</code>
<code>Secret[28, 2]</code>	<code>"●●●●●●●●●●"</code>

Note:

- For security, the passwords are stored in an encrypted form, shown as `"●●●●●●●●●●"` in the example.
- The passwords cannot be used without being decrypted.
- You may assume that the encrypted form of a password will **not** be an empty string.

Additional modules are defined as follows:

Module	Description
<code>Encrypt()</code>	<ul style="list-style-type: none"> • Takes a password as a string • Returns the encrypted form of the password as a string
<code>Decrypt()</code>	<ul style="list-style-type: none"> • Takes an encrypted password as a string • Returns the decrypted form of the password as a string
<code>FindPassword()</code>	<ul style="list-style-type: none"> • Takes a website domain name as a string • Searches for the website domain name in the array <code>Secret</code> • If the website domain name is found, the decrypted password is returned • If the website domain name is not found, an empty string is returned
<code>AddPassword()</code>	<ul style="list-style-type: none"> • Takes two parameters as strings: a website domain name and a password • Searches for the website domain name in the array <code>Secret</code> and if not found, adds the website domain name and the encrypted password to the array • Returns <code>TRUE</code> if the website domain name and encrypted password are added to the array, otherwise returns <code>FALSE</code>

The first three modules have been written.

(c) The content of the array `Secret` is to be stored in a text file for backup.

It **must** be possible to read the data back from the file and extract the website domain name and the encrypted password.

Both the website domain name and encrypted password are stored in the array as strings of characters.

The encrypted password may contain any character from the character set used and the length of both the encrypted password and the website domain name is variable.

Explain how a single line of the text file can be used to store the website domain name and the encrypted password.

.....

.....

.....

.....

.....

.....

..... [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.



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
.....

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

Error
.....

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

Error
.....

[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>	
2	<code>(Points MOD 2 = 0) OR Exempt</code>	
3	<code>(Points <= 75) AND (Active OR Exempt)</code>	
4	<code>(Active OR NOT Active) AND NOT Exempt</code>	

[2]

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

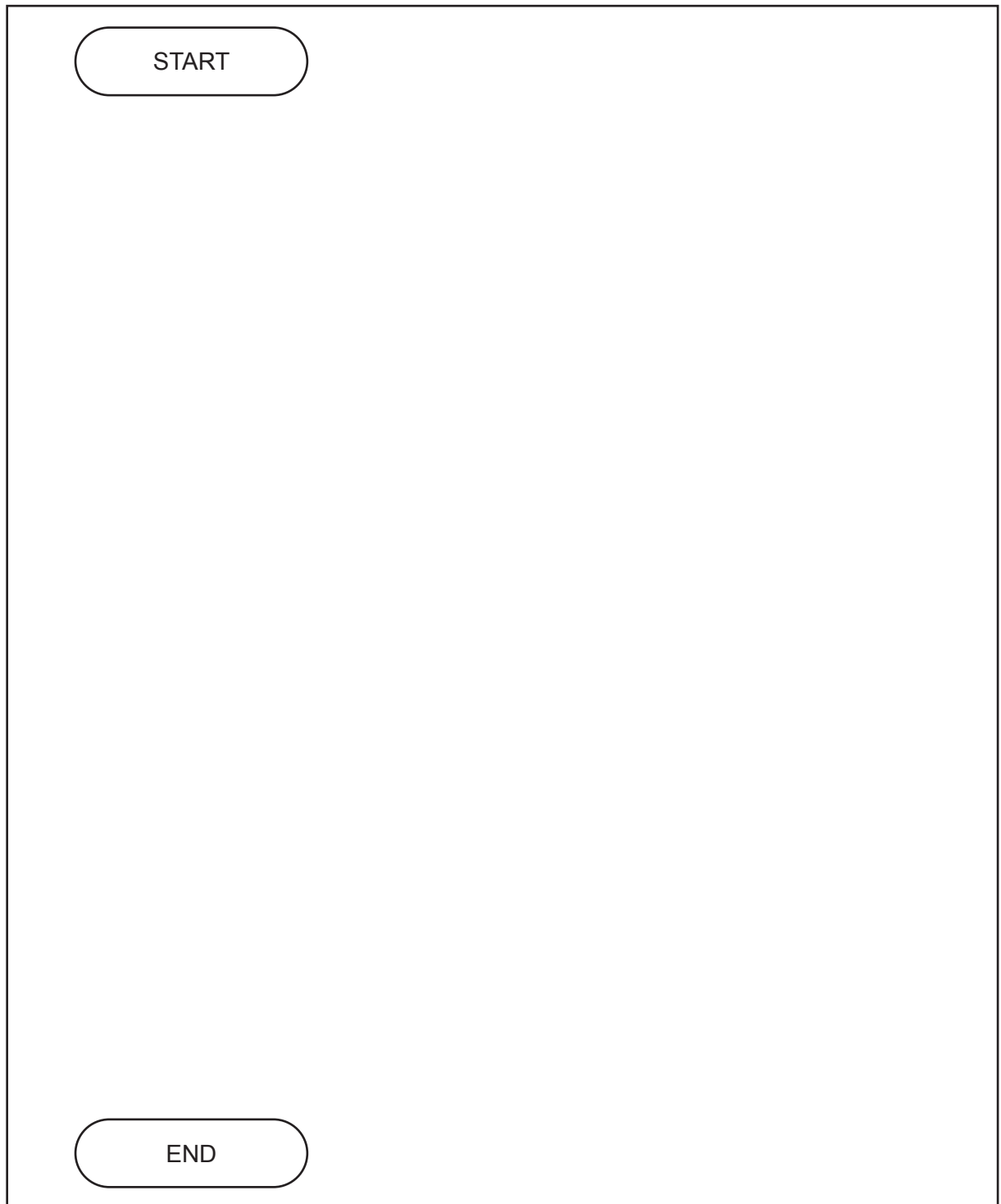
..... [1]

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

Assignment

[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

Advantage

.....

[4]

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

.....

..... [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

.....

.....

2

.....

.....

3

.....

.....

4

.....

.....

5

.....

.....

[5]

- 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

.....

Reason 2

.....

[2]

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

..... [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

Description

.....

.....

.....

.....

[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:

		1	2	3	...	126	127	128
Sample:	1	20	20	20		30	30	2
	2	20	20	30		50	30	3
	3	20	20	40		40	40	4
	4	20	20	50		40	50	20
	5	20	3	5		6	60	4
	6	20	4	2		4	70	30
		↓	↓	↓	⏏	↓	↓	↓
Result:		20	20	35		40	46	25

7 A school has a computerised library system that allows students to borrow books for a fixed length of time. The system uses text files to store details of students, books and loans.

A new module is to be written which will generate emails to each student who has an overdue book.

(a) Decomposition will be used to break down the problem of designing the new module into sub-problems.

Identify **three** program modules that could be used in the design **and** describe their use.

Module 1

Use

.....

.....

.....

Module 2

Use

.....

.....

.....

Module 3

Use

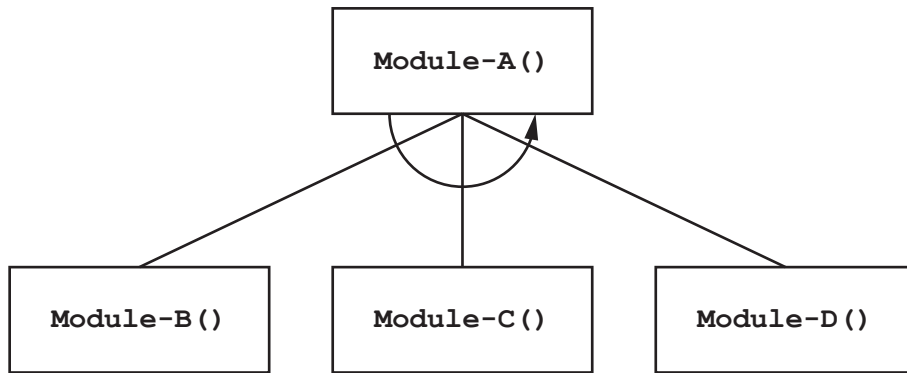
.....

.....

.....

[3]

(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.

.....

.....

.....

..... [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
<code>CheckNewItem()</code>	<ul style="list-style-type: none">called with a parameter of type string representing a line of item informationchecks to see whether an item with the same <code>ItemNum</code> already exists in the filereturns <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()`.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....
.....
.....
..... [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.

..... [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.

.....
.....
.....
..... [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.

..... [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.

.....
.....
.....
.....
..... [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.



Cambridge International AS & A Level

CANDIDATE
NAME

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/22

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 program calculates the postal cost based on the weight of the item and its destination. Calculations occur at various points in the program and these result in the choice of several possible postal costs. The programmer has built these postal costs into the program.

For example, the postal cost of \$3.75 is used in the following lines of pseudocode:

```

IF Weight < 250 AND ValidAddress = TRUE THEN
  ItemPostalCost ← 3.75 // set postal cost for item to $3.75
  ItemStatus ← "Valid" // item can be sent
ENDIF

```

- (a) (i) Identify a more appropriate way of representing the postal costs.

..... [1]

- (ii) Describe the advantages of your answer to **part (a)(i)** with reference to this program.

.....

.....

.....

.....

.....

.....

..... [3]

- (b) The lines of pseudocode contain features that make them easier to understand.

State **three** of these features.

1

2

3 [3]

- (c) Give the **appropriate** data types for the following variables:

ValidAddress

ItemPostalCost

ItemStatus [3]

2 A program stores a user's date of birth using a variable `MyDOB` of type `DATE`.

(a) Write a pseudocode statement, using a function from the **insert**, to assign the value corresponding to 17/11/2007 to `MyDOB`.

..... [1]

(b) `MyDOB` has been assigned a valid value representing the user's date of birth.

Write a pseudocode statement to calculate the number of months from the month of the user's birth until the end of the year and to assign this to the variable `NumMonths`.

For example, if `MyDOB` contains a value representing 02/07/2008, the value 5 would be assigned to `NumMonths`.

..... [2]

(c) The program will output the day of the week corresponding to `MyDOB`.

For example, given the date 22/06/2023, the program will output "Thursday".

An algorithm is required. An array will be used to store the names of the days of the week.

Define the array **and** describe the algorithm in **four** steps.

Do **not** use pseudocode statements in your answer.

Array definition

.....

Step 1

.....

.....

Step 2

.....

.....

Step 3

.....

.....

Step 4

.....

.....

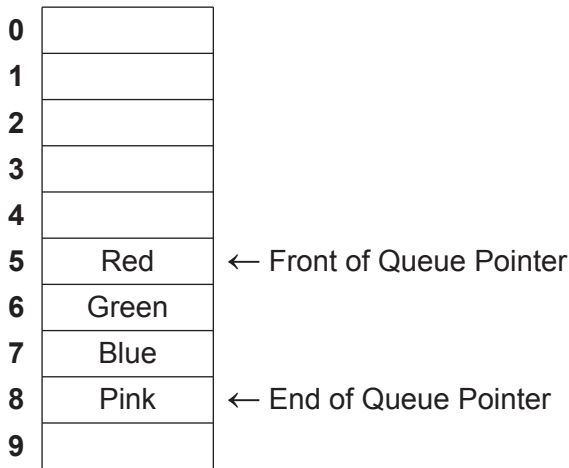
[6]

3 A program stores data in a text file. When data is read from the file, it is placed in a queue.

(a) The diagram below represents an Abstract Data Type (ADT) implementation of the queue. Each data item is stored in a separate location in the data structure. During initial design, the queue is limited to holding a maximum of 10 data items.

The operation of this queue may be summarised as follows:

- The Front of Queue Pointer points to the next data item to be removed.
- The End of Queue Pointer points to the last data item added.
- The queue is circular so that locations can be reused.



(i) Describe how the data items Orange and Yellow are added to the queue shown in the diagram.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

- 5 A programmer has produced the following pseudocode to output the square root of the numbers from 1 to 10.

Line numbers are for reference only.

```

10 DECLARE Num : REAL
11 Num ← 1.0
...
40 REPEAT
41     CALL DisplaySqrt (Num)
42     Num ← Num + 1.0
43 UNTIL Num > 10
...
50 PROCEDURE DisplaySqrt (BYREF ThisNum : REAL)
51     OUTPUT ThisNum
52     ThisNum ← SQRT(ThisNum) // SQRT returns the square root
53     OUTPUT " has a square root of ", ThisNum
54 ENDPROCEDURE

```

The pseudocode is correctly converted into program code.

Function `SQRT()` is a library function and contains no errors.

The program code compiles without errors, but the program gives unexpected results. These are caused by a design error.

- (a) Explain why the program gives unexpected results.

.....

.....

.....

.....

.....

.....

..... [3]

- (b) Explain why the compiler does **not** identify this error.

.....

..... [1]

(c) Describe how a typical Integrated Development Environment (IDE) could be used to identify this error.

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

(d) The pseudocode is converted into program code as part of a larger program.

During compilation, a complex statement generates an error.

The programmer does not want to delete the complex statement but wants to change the statement so that it is ignored by the compiler.

State how this may be achieved.

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

- 6 A procedure `Square()` will take an integer value in the range 1 to 9 as a parameter and output a number square.

The boundary of a number square is made up of the character representing the parameter value. The inside of the number square is made up of the asterisk character (*).

Parameter value	1	2	3	4	...	9
Output	1	22 22	333 3*3 333	4444 4**4 4**4 4444	...	99999999 9***9 9***9 9***9 9***9 9***9 9***9 9***9 9***9 9***9 99999999

The pseudocode `OUTPUT` command starts each output on a new line. For example, the following three `OUTPUT` statements would result in the outputs as shown:

```
OUTPUT "Hello"
OUTPUT "ginger"
OUTPUT "cat"
```

Resulting output:

```
Hello
ginger
cat
```


.....

.....

.....

.....

.....

.....

..... [6]

7 A computer system for a shop stores information about each customer. The items of information include name and address (both postal and email) together with payment details and order history. The system also stores the product categories they are interested in and how they would like to be contacted.

(a) The shop wants to add a program module that will generate emails to be sent to customers who may be interested in receiving details of new products.

(i) State **three** items of information that the new module would need. Justify your choice in each case.

Information

Justification

.....

Information

Justification

.....

Information

Justification

.....

[3]

(ii) Identify **two** items of customer information that would **not** be required by the new module. Justify your choice in each case.

Information

Justification

.....

Information

Justification

.....

[2]

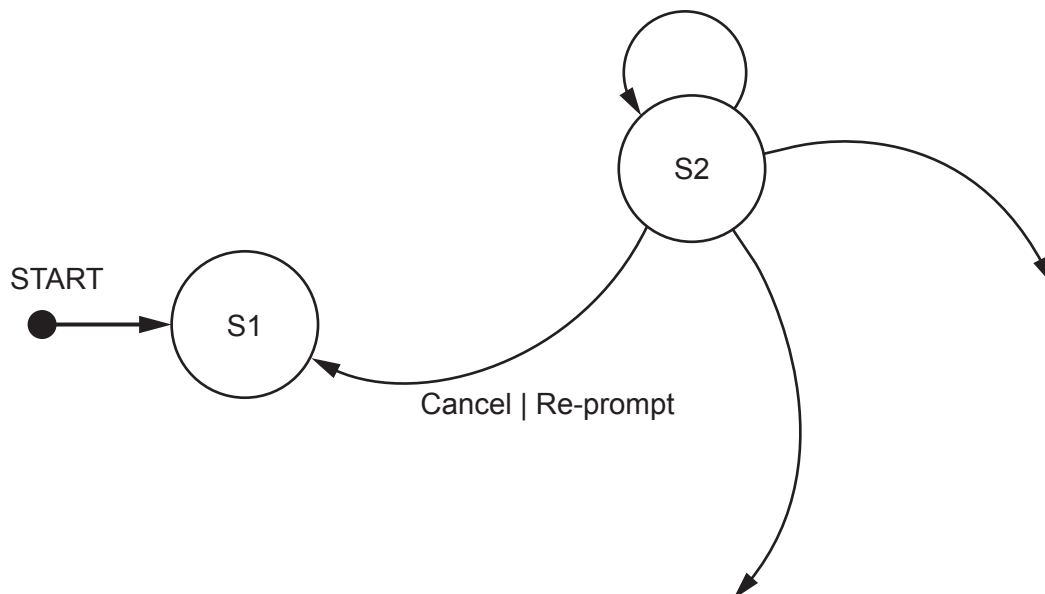
- (b) The program includes a module to validate a Personal Identification Number (PIN). This is used when customers pay for goods using a bank card.

A state-transition diagram has been produced for this module.

The table show the inputs, outputs and states for this part of the program:

Current state	Input	Output	Next state
S1	Input PIN		S2
S2	Re-input PIN	Display error	S2
S2	Cancel	Re-prompt	S1
S2	Valid PIN	Enable payment	S4
S2	Too many tries	Block Account	S3

Complete the state-transition diagram to represent the information given in the table.



[4]

- 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>
```

Valid item information is as follows:

	Format	Comment
ItemNum	4 numeric characters	unique number for each item in the range "0001" to "5999" inclusive
SupplierCode	3 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>OnlyAlpha()</code> (already written)	<ul style="list-style-type: none"> called with a parameter of type string returns <code>TRUE</code> if the string contains only alphabetic characters, otherwise returns <code>FALSE</code>
<code>CheckInfo()</code>	<ul style="list-style-type: none"> called with a parameter of type string representing a line of item information checks to see whether the item information in the string is valid returns <code>TRUE</code> if the item information is valid, otherwise returns <code>FALSE</code>

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.



Cambridge International AS & A Level

CANDIDATE
NAME

--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/23

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 The following pseudocode represents part of the algorithm for a program.

Line numbers are for reference only.

```

10  DECLARE Sheet4 : ARRAY[1:2, 1:50] OF INTEGER
...
100 FOR PCount ← 0 TO 49
101     Sheet4[1, PCount] ← 0
102     Sheet4[2, PCount] ← 47
103 NEXT PCount
    
```

(a) The pseudocode contains references to an array.

Complete the table by writing the answer for each row.

Answer

The dimension of the array	
The name of the variable used as an array index	
The number of elements in the array	

[3]

(b) The pseudocode contains two errors. One error is that variable `PCount` has not been declared.

Identify the **other** error **and** state the line number where it occurs.

Error

.....

.....

Line number

[2]

(c) The pseudocode does not include a declaration for `PCount`.

State the data type that should be used in the declaration.

..... [1]

- (d) The pseudocode statements given in the following table are used in other parts of the algorithm.

Complete the table by placing **one or more** ticks (✓) in each row.

The first row has already been completed.

Pseudocode statement	Input	Process	Output
INPUT MyChoice	✓		
OUTPUT FirstName & LastName			
WRITEFILE YourFile, TextLine			
READFILE MyFile, TextLine			
Result ← SQRT (NextNum)			

[4]

2 A program stores a date of birth for a student using a variable, `MyDOB`, of type `DATE`.

(a) `MyDOB` has been assigned a valid value corresponding to Kevin's date of birth.

Complete the pseudocode statement to test whether Kevin was born on a Thursday.

IF THEN [2]

(b) A function `CheckDate()` will take three integer parameters representing a day, month and year of a given date.

The function will validate the date of birth for a student that the parameters passed to it represent.

For a date to be valid, a student must be at least 18 in year 2020.

(i) Two of the parameter values can be checked without reference to the third parameter.

Describe these **two** checks.

Check 1
.....
.....

Check 2
.....
.....
.....

[2]

(ii) Several values of the parameter representing the day can only be checked completely by referring to the value of **one other** parameter.

Describe this check.

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

3 A program processes data using a stack. The data is copied to a text file before the program ends.

(a) The following diagram shows the current state of the stack.

The operation of this stack may be summarised as follows:

- The `TopOfStack` pointer points to the last item added to the stack.
- The `BottomOfStack` pointer points to the first item on the stack.
- The stack grows upwards when items are added.

Stack		Pointer
Memory location	Value	
506		
505	WWW	← <code>TopOfStack</code>
504	YYY	
503	XXX	
502	ZZZ	
501	NNN	
500	PPP	← <code>BottomOfStack</code>

(i) An error will be generated if an attempt is made to POP a value when the stack is empty.

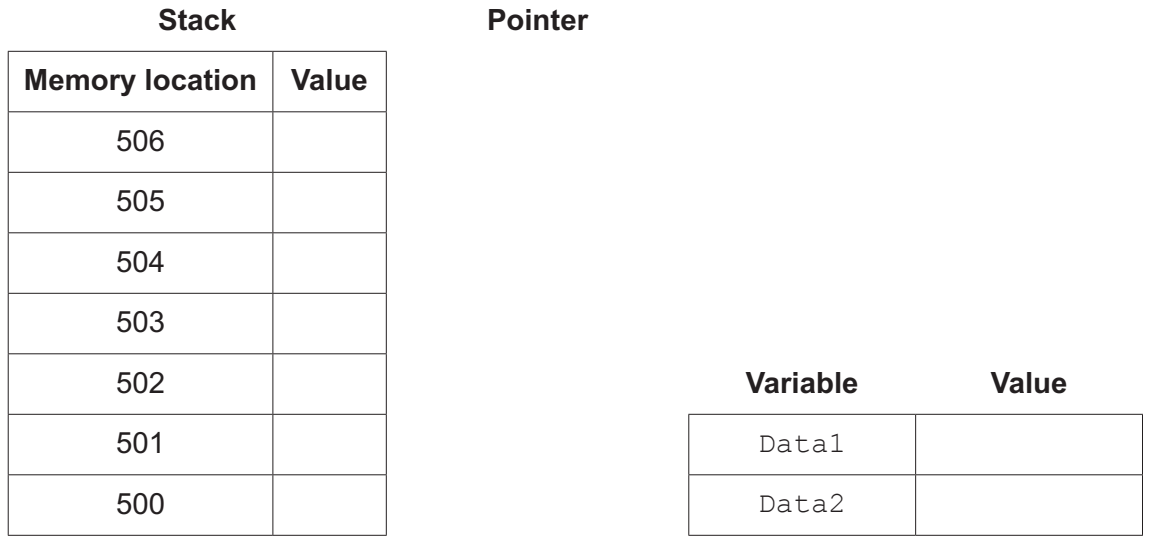
State the maximum number of consecutive POP operations that could be performed on the stack shown above **before** an error is generated.

..... [1]

(ii) The following operations are performed:

1. POP and store value in variable `Data1`
2. POP and store value in variable `Data2`
3. PUSH value AAA
4. PUSH value BBB
5. POP and discard value
6. POP and store value in variable `Data2`

Complete the diagram to show the state of the stack and the variables **after** the given operations have been performed.



[4]

(b) The data is copied to a text file before the program ends.

(i) State an advantage of writing the data from the stack to a text file before the program ends.

.....
 [1]

(ii) A module `SaveStack()` will write the data from the stack to a text file.

Express an algorithm for `SaveStack()` as five steps that could be used to produce pseudocode.

Write the **five** steps.

Step 1

Step 2

Step 3

Step 4

Step 5

[5]

5 A program is designed, coded and compiled without errors. The compiled code is sent for testing.

(a) The program will be tested using the walkthrough method.

Additional information will be needed before this method can be used.

Identify this additional information **and** explain why it is needed.

Additional information

.....

Explanation

.....

.....

.....

[3]

(b) Testing is completed and the program is made available to users.

Some time later, changes are made to the program to improve the speed of response.

State the type of maintenance that has been applied to the program.

..... [1]

(b) The check performed by procedure `Select()` on the last two digits is needed at several places in the program and will be implemented using a new function.

The new function `CheckNum()` will:

- allow the required sum to be specified (not just 6)
- check one number
- return an appropriate value.

Describe the function interface **and two** advantages of this modular approach.

Interface

.....

.....

.....

Advantage 1

.....

Advantage 2

.....

[4]

7 A school has a library system which allows students to borrow books for a length of time. Information relating to students and books is stored in text files. Student information includes name, home address, email address, date of birth, tutor and subject choices. Book information includes author, title, subject category, library location and the date that the book was borrowed.

A program helps the staff to manage the borrowing of books.

(a) A new module needs to be written to generate emails to send to students who have an overdue book. Students who are sent an email are prevented from borrowing any more books until the overdue book is returned.

The process of abstraction has been used when designing the new module.

(i) State the purpose of applying abstraction to this problem.

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

(ii) Identify **one** item of information that is required and **one** item that is **not** required in the new module. Justify your choices.

Item required

Justification

.....
.....

Item not required

Justification

.....
.....

[2]

(iii) Identify **two** operations that would be required to process data when an overdue book is returned.

Operation 1

.....
.....

Operation 2

.....
.....

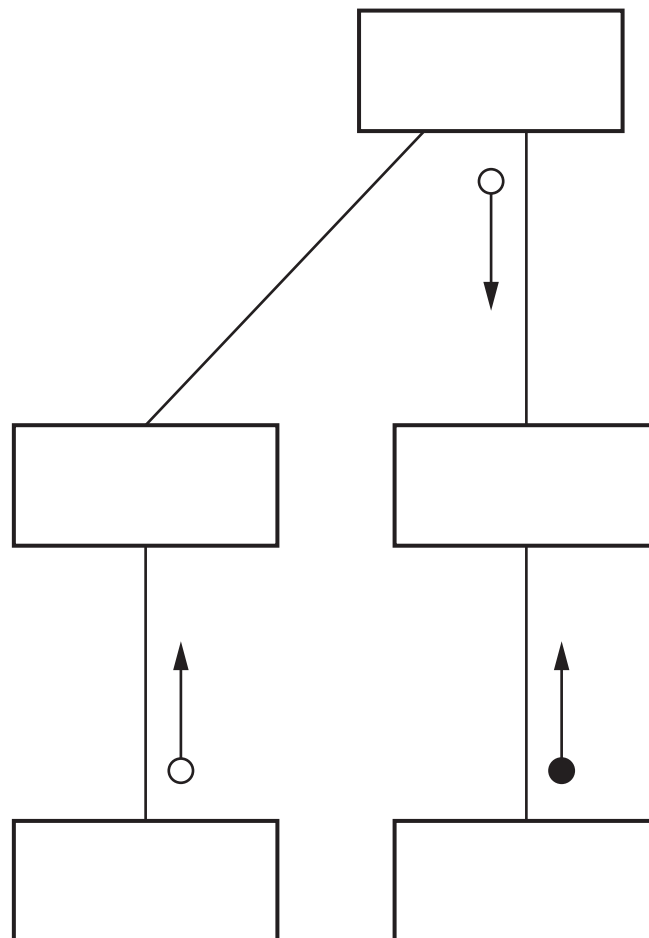
[2]

(b) Part of the library program contains program modules with headers as follows:

Pseudocode module header
PROCEDURE Module-X()
PROCEDURE Module-Y(BYREF RA : INTEGER, SA : REAL)
PROCEDURE Overlay()
FUNCTION Replace(RA : INTEGER, RB : BOOLEAN) RETURNS BOOLEAN
FUNCTION Reset(TA : STRING) RETURNS INTEGER

Module-X() and Module-Y() are both called from module Overlay().

Complete the structure chart.



[3]

- 8 A computer shop assembles desktop 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 number for each item in the range "0001" to "5999" inclusive
SupplierCode	3 alphabetic characters	code 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.

The programmer has defined the first program module as follows:

Module	Description
<code>ChangeSupp()</code>	<ul style="list-style-type: none"> called with two parameters <code>Code1</code> and <code>Code2</code> of type string that represent valid supplier codes creates a new file <code>NewStock.txt</code> from the contents of the file <code>Stock.txt</code> where any reference to <code>Code1</code> is replaced by <code>Code2</code> returns a count of the number of items that have had their supplier code changed

- (c) The format of the output from module `Report_1()` from **part (b)** is changed. The number of items listed is moved to the top of the report as shown in the example:

```
Report for Supplier: DRG
Number of items listed: 3
```

Item	Description
1234	USB Printer Cable 3m
1273	32GB USB Flash Drive
1350	Mouse Mat 320 x 240mm

- (i) Explain why this new layout would increase the complexity of the algorithm.

.....

.....

.....

..... [2]

- (ii) The algorithm will be modified to produce the report in the new format. The modified algorithm will be implemented so that the file `Stock.txt` is only read once.

Describe the modified algorithm.

.....

.....

.....

.....

.....

.....

..... [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.



Cambridge International AS & A Level

CANDIDATE
NAME

--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/21

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 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 **20** pages. Any blank pages are indicated.

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

- 1 Sylvia is testing a program that has been written by her colleague. Her colleague tells her that the program does not contain any syntax errors.

- (a) (i) State what her colleague means by “does not contain any syntax errors”.

.....

 [1]

- (ii) Identify **and** describe **one** other type of error that the program may contain.

Type of error

Description

..... [2]

- (b) Complete the following table by giving the appropriate data type in each case.

Use of variable	Data type
The average mark in a class of 40 students	
An email address	
The number of students in the class	
To indicate whether an email has been read	

[4]

(c) An airline wants to provide passengers with information about individual flights and allow them to book their flight using an online booking system.

(i) Tick (✓) **one** box in each row of the table to indicate whether each item of information would be essential for the customer when making the booking.

Information	Essential	Not essential
Departure time		
Flight number		
Departure airport		
Aircraft type		
Ticket price		
Number of seats in aircraft		

[3]

(ii) Identify the technique used to filter out information that is not essential when designing the booking system **and** state one benefit of this technique.

Technique

Benefit

.....

[2]

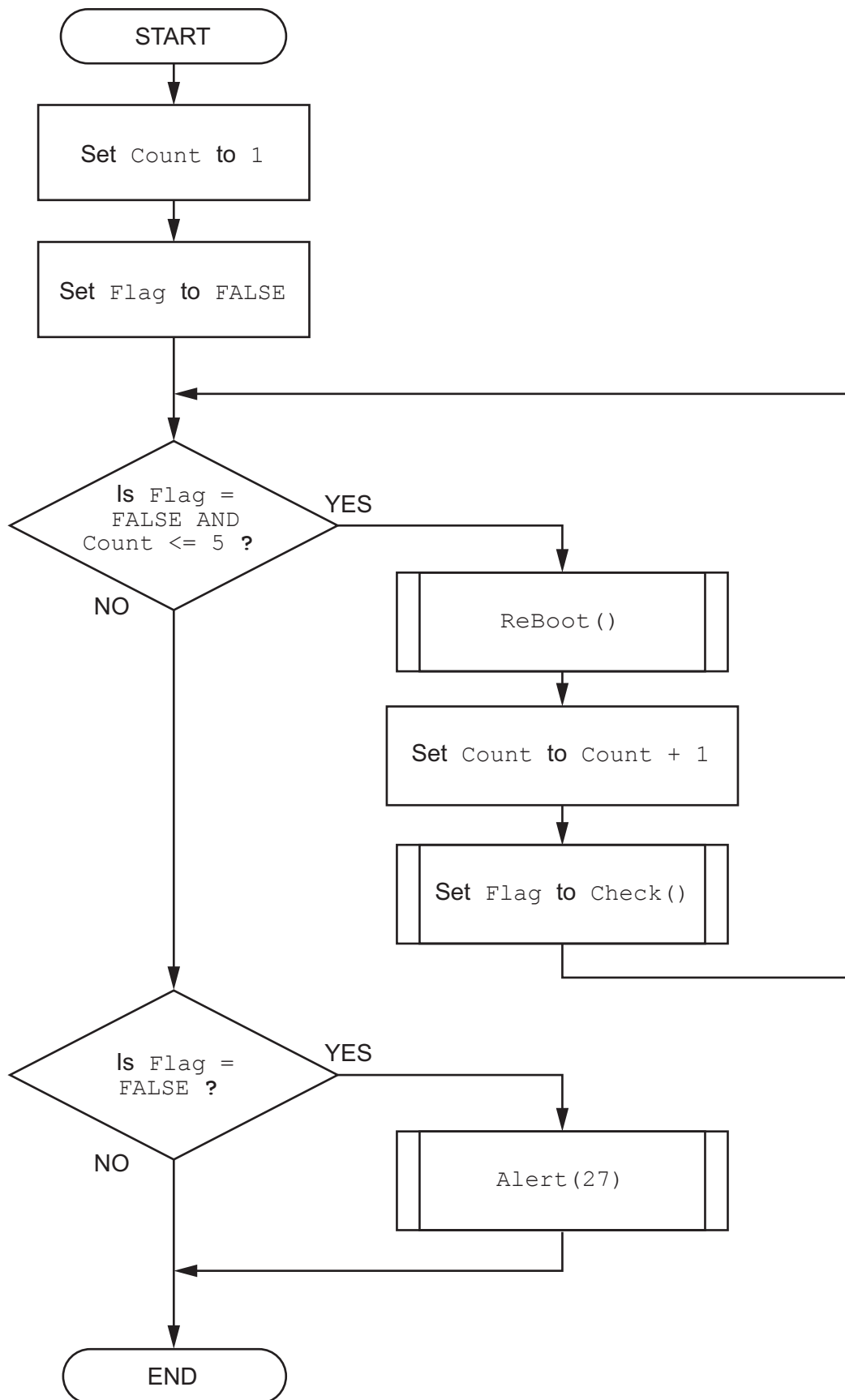
(iii) Identify **two additional** pieces of essential information that a passenger might need when booking a flight.

1

2

[2]

(b) The program flowchart shown describes a simple algorithm.



- 3 (a) The diagram below represents a queue Abstract Data Type (ADT) that can hold a maximum of eight items.

The operation of this queue may be summarised as follows:

- The front of queue pointer points to the next item to be removed.
- The end of queue pointer points to the last item added.
- The queue is circular so that empty storage elements can be reused.

0	Frog	← Front of queue pointer
1	Cat	
2	Fish	
3	Elk	← End of queue pointer
4		
5		
6		
7		

- (i) Describe how “Octopus” is added to the given queue.

.....

.....

.....

..... [2]

- (ii) Describe how the next item in the given queue is removed and stored in the variable `AnimalName`.

.....

.....

.....

..... [2]

- (iii) Describe the state of the queue when the **front of queue** and the **end of queue** pointers have the same value.

.....

..... [1]

(b) Some operations are carried out on the original queue given in **part (a)**.

(i) The current state of the queue is:

0	Frog
1	Cat
2	Fish
3	Elk
4	
5	
6	
7	

Complete the diagram to show the state of the queue after the following operations:

Add “Wasp”, “Bee” and “Mouse”, and then remove two data items.

[3]

(ii) The state of the queue after other operations are carried out is shown:

0	Frog	
1	Cat	
2	Fish	
3	Elk	← Front of queue pointer
4	Wasp	
5	Bee	
6	Mouse	← End of queue pointer
7	Ant	

Complete the following diagram to show the state of the queue after the following operations:

Remove one item, and then add “Dolphin” and “Shark”.

0	
1	
2	
3	
4	
5	
6	
7	

[2]

(c) The queue is implemented using a 1D array.

Describe the algorithm that should be used to modify the **end of queue pointer** when adding an item to the queue.

Your algorithm should detect any potential error conditions.

.....

.....

.....

.....

.....

.....

..... [3]

4 A program controls the heating system in a sports hall.

Part of the program involves reading a value from a sensor. The sensor produces a numeric value that represents the temperature. The value is an integer, which should be in the range 0 to 40 inclusive.

A program function has been written to validate the values from the sensor.

(a) A test plan is needed to test the function.

Complete the table. The first line has been completed for you.

You can assume that the sensor will generate only integer data values.

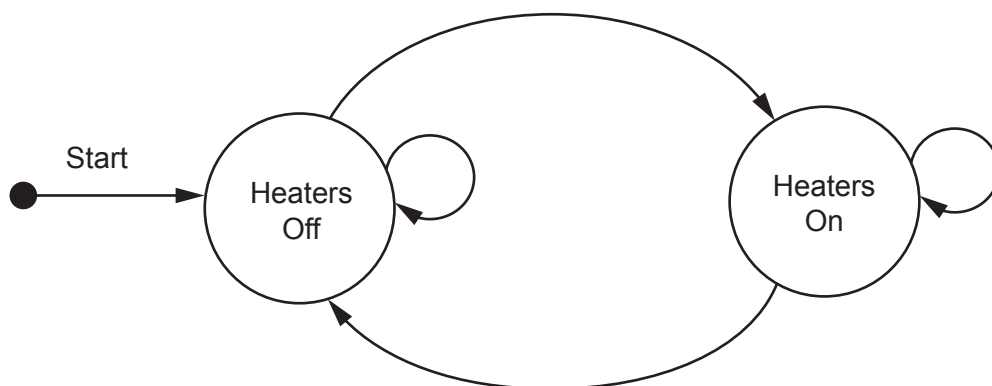
Test	Test data value	Explanation	Expected outcome
1	23	Normal data	Data is accepted
2			
3			
4			
5			

[4]

(b) A program module controls the heaters. This module operates as follows:

- If the temperature is below 10, switch the heaters on.
- If the temperature is above 20, switch the heaters off.

Complete the following state-transition diagram for the heating system:



[3]

- 5 The following data items will be recorded each time a student successfully logs on to the school network:

Data item	Example data
Student ID	"CJL404"
Host ID	"Lib01"
Time and date	"08:30, June 01, 2021"

The Student ID is six characters long. The other two data items are of variable length.

A single string will be formed by concatenating the three data items. A separator character will need to be inserted between items two and three.

For example:

"CJL404Lib01<separator>08:30, June 01, 2021"

Each string represents one log entry.

A programmer decides to store the concatenated strings in a 1D array `LogArray` that contains 2000 elements. Unused array elements will contain an empty string.

- (a) Suggest a suitable separator character **and** give a reason for your choice.

Character

Reason

..... [2]

- (b) The choice of data structure was made during one stage of the program development life cycle.

Identify this stage.

..... [1]

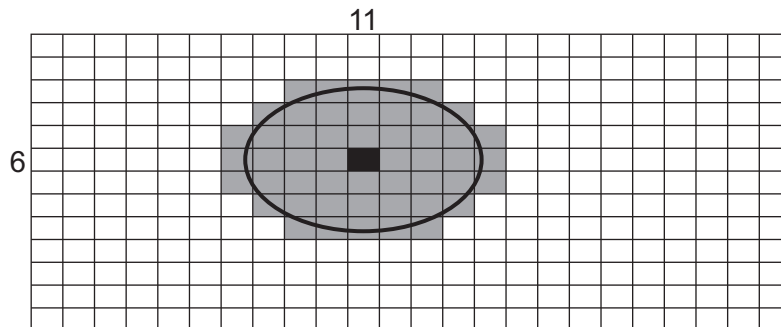
- 6 A mobile phone has a touchscreen. The screen is represented by a grid, divided into 800 rows and 1280 columns.

The grid is represented by a 2D array `Screen` of type `INTEGER`. An array element will be set to 0 unless the user touches that part of the screen.

Many array elements are set to 1 by a single touch of a finger or a stylus.

The following diagram shows a simplified touchscreen. The dark line represents a touch to the screen. All grid elements that are wholly or partly inside the outline will be set to 1. These elements are shaded.

The element shaded in black represents the centre point.



A program is needed to find the coordinates (the row and column) of the centre point. The centre point on the diagram above is row 6, column 11.

Assume:

- the user may only touch one area at a time
- screen rotation does not affect the touchscreen.

The programmer has started to define program modules as follows:

Module	Description
<code>SetRow()</code> (generates test data)	<ul style="list-style-type: none"> • Called with three parameters of type <code>INTEGER</code>: <ul style="list-style-type: none"> ◦ a row number ◦ the number of pixels to be skipped starting from column 1 ◦ the number of pixels that should be set to 1 • Sets the required number of pixels to 1 <p>For example, <code>SetRow(3, 8, 5)</code> will give row 3 as in the diagram shown.</p>
<code>SearchInRow()</code>	<ul style="list-style-type: none"> • Takes two parameters of type <code>INTEGER</code>: <ul style="list-style-type: none"> ◦ a row number ◦ a start column (1 or 1280) • Searches the given row from the start column (either left to right or right to left) for the first column that contains an element set to 1 • Returns the column number of the first element in the given row that is set to 1 • Returns -1 if no element is set to 1
<code>SearchInCol()</code>	<ul style="list-style-type: none"> • Takes two parameters of type <code>INTEGER</code>: <ul style="list-style-type: none"> ◦ a column number ◦ a start row (1 or 800) • Searches the given column from the start row (either up or down) for the first row that contains an element set to 1 • Returns the row number of the first element in the given column that is set to 1 • Returns -1 if no element is set to 1

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.



Cambridge International AS & A Level

CANDIDATE
NAME

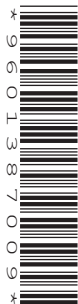
--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/22

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 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 **20** pages. Any blank pages are indicated.

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

- 1 (a) A programmer applies decomposition to a problem that she has been asked to solve.

Describe decomposition.

.....

.....

.....

.....

.....

.....

..... [2]

- (b) The following pseudocode assigns a value to an element of an array:

```
ThisArray[n] ← 42
```

Complete the following table by writing the answer for each row.

Answer

The number of dimensions of <code>ThisArray</code>	
The technical terms for minimum and maximum values that the variable <code>n</code> may take	
The technical term for the variable <code>n</code> in the pseudocode statement	

[3]

- (c) Complete the pseudocode expressions so that they evaluate to the values shown.

Any functions and operators used must be defined in the **insert**.

Expression	Evaluates to
..... ('C')	67
2 * ("27")	54
..... (27 /)	13
"Sub" & ("Abstraction" , ,)	"Subtract"

[4]

- (d) Evaluate the expressions given in the following table. The variables have been assigned values as follows:

PumpOn ← TRUE
PressureOK ← TRUE
HiFlow ← FALSE

Expression	Evaluates to
PressureOK AND HiFlow	
PumpOn OR PressureOK	
NOT PumpOn OR (PressureOK AND NOT HiFlow)	
NOT (PumpOn OR PressureOK) AND NOT HiFlow	

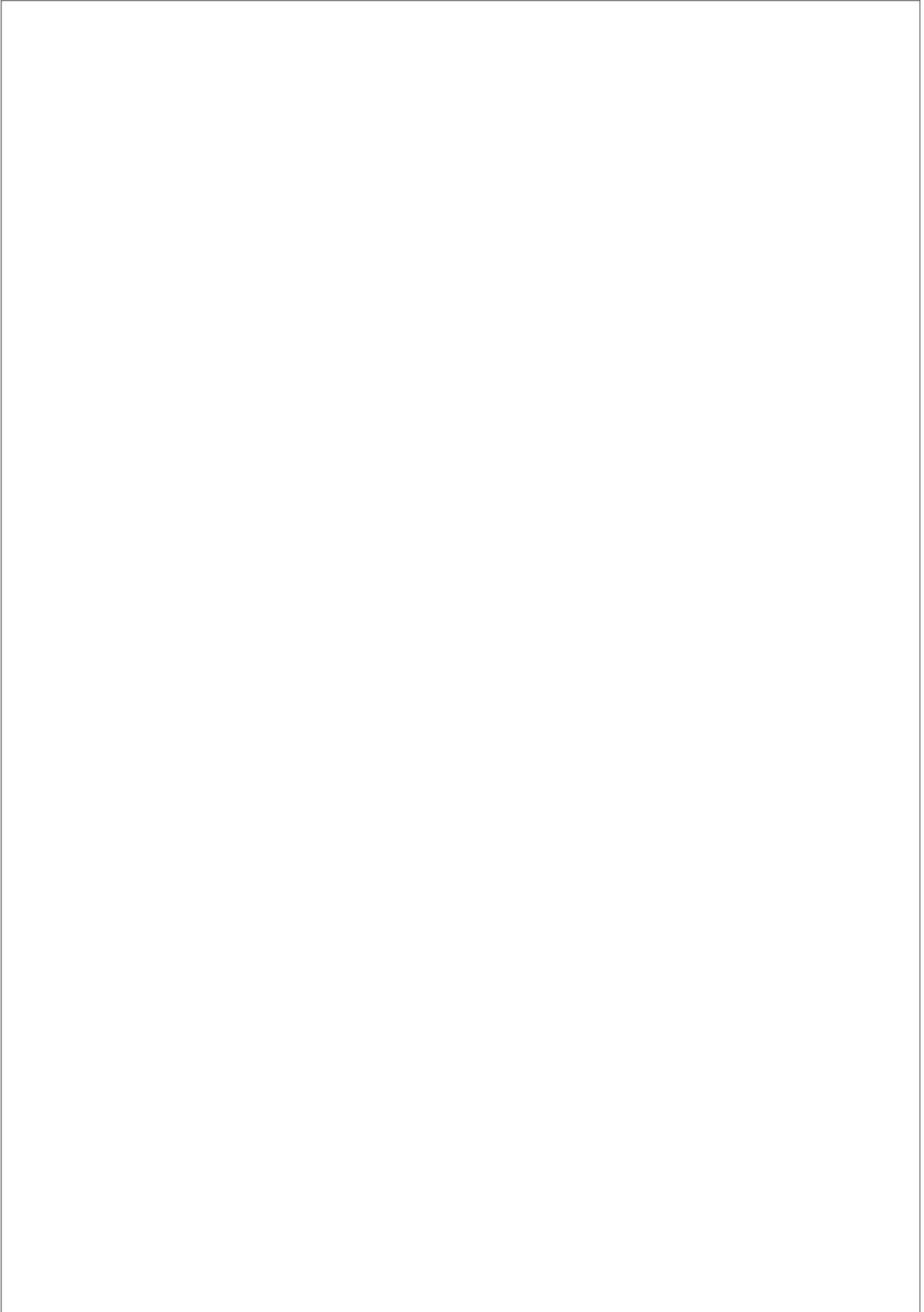
[2]

2 (a) An algorithm will:

1. input an integer value
2. jump to step 6 if the value is zero
3. sum and count the positive values
4. sum and count the negative values
5. repeat from step 1
6. output the two sum values and the two count values.

Draw a program flowchart on the following page to represent the algorithm.

Note that variable declarations are not required in program flowcharts.



[5]

(b) A software company is working on a project to develop a website for a school.

The school principal has some ideas about the appearance of the website but is unclear about all the details of the solution. The principal would like to see an initial version of the website.

(i) Identify a life cycle method that would be appropriate in this case.

Give a reason for your choice.

Life cycle method

Reason

.....

.....

.....

[2]

(ii) The website project has progressed to the design stage.

State **three** activities that will take place during the design stage of the program development life cycle.

1

2

3

[3]

3 A programmer is writing a program to help manage clubs in a school.

Data will be stored about each student in the school and each student may join up to three clubs.

The data will be held in a record structure of type `Student`.

The programmer has started to define the fields that will be needed as shown in the following table.

Field	Typical value	Comment
StudentID	"CF1234"	Unique to each student
Email	"Carmen47@xyzmail.com"	Contains letters, numbers and certain symbols
Club_1	1	Any value in the range 1 to 99 inclusive
Club_2	14	Any value in the range 1 to 99 inclusive
Club_3	27	Any value in the range 1 to 99 inclusive

(a) (i) Write pseudocode to declare the record structure for type `Student`.

.....

.....

.....

.....

.....

.....

.....

.....

..... [3]

(ii) A 1D array `Membership` containing 3000 elements will be used to store the student data.

Write pseudocode to declare the `Membership` array.

.....

..... [2]

(iii) Some of the elements of the array will be unused.

Give an **appropriate** way of indicating an unused array element.

.....

..... [1]

(iv) Some students are members of less than three clubs.

State **one** way of indicating an unused club field.

.....

..... [1]

4 The following is a procedure design in pseudocode.

Line numbers are given for reference only.

```

10 PROCEDURE Check(InString : STRING)
11     DECLARE Odds, Evens, Index : INTEGER
12
13     Odds ← 0
14     Evens ← 0
15     Index ← 1
16
17     WHILE Index <= LENGTH(InString)
18         IF STR_TO_NUM(MID(InString, Index, 1)) MOD 2 <> 0 THEN
19             Odds ← Odds + 1
20         ELSE
21             Evens ← Evens + 1
22         ENDIF
23         Index ← Index + 1
24     ENDWHILE
25
26     CALL Result(Odds, Evens)
27 ENDPROCEDURE
    
```

(a) Complete the following table by giving the answers, using the given pseudocode.

Answer

A line number containing a variable being incremented	
The type of loop structure	
The number of functions used	
The number of parameters passed to STR_TO_NUM()	
The name of a procedure other than Check()	

[5]

(b) The pseudocode includes several features that make it easier to read and understand.

Identify **three** of these features.

- 1
- 2
- 3

[3]

(c) (i) The loop structure used in the pseudocode is not the most appropriate.

State a more appropriate loop structure **and** justify your choice.

Loop structure

Justification

.....

.....

[2]

(ii) The appropriate loop structure is now used. Two lines of pseudocode are changed and two lines are removed.

Write the line numbers of the two lines that are removed.

.....

..... [1]

- 5 A company has several departments. Each department stores the name, email address and the status of each employee in that department in its own text file. All text files have the same format.

Employee details are stored as three separate data strings on three consecutive lines of the file. An example of the first six lines of one of the files is as follows:

File line	Comment
1	First employee name
2	First email address
3	First employee status
4	Second employee name
5	Second email address
6	Second employee status

A procedure `MakeNewFile()` will:

- take three parameters as strings:
 - an existing file name
 - a new file name
 - a search status value
- create a new text file using the new file name
- write all employee details to the new file where the employee status is **not** equal to the search status value
- count the number of sets of employee details that were in the original file
- count the number of sets of employee details that were written to the new file
- produce a summary output.

An example summary output is as follows:

```
File Marketing contained 54 employee details
52 employee sets of details were written to file NewMarketingList
```

- (a) Write pseudocode for the procedure `MakeNewFile()`.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

(b) An alternative format could be used for storing the data.

A text file will still be used.

(i) Describe the alternative format.

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

(ii) State **one** advantage **and one** disadvantage of the alternative format.

Advantage
.....
Disadvantage
..... [2]

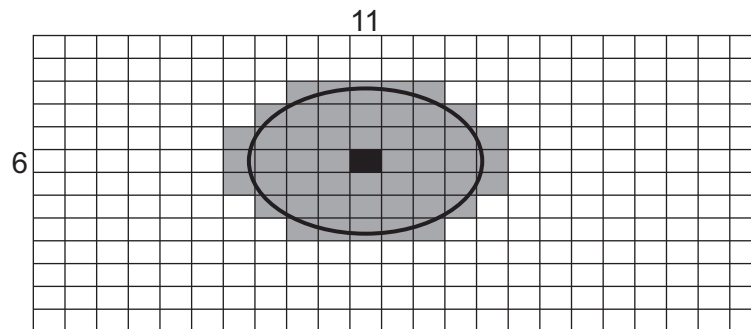
- 6 A mobile phone has a touchscreen. The screen is represented by a grid, divided into 800 rows and 1280 columns.

The grid is represented by a 2D array `Screen` of type `INTEGER`. An array element will be set to 0 unless the user touches that part of the screen.

Many array elements will be set to 1 by a single touch of a finger or a stylus.

The following diagram shows a simplified touchscreen. The dark line represents a touch on the screen. All grid elements that are wholly or partly inside the outline will be set to 1. These elements are shaded.

The element shaded in black represents the centre point of the touch.



A program is needed to find the coordinates (the row and column) of the centre point. The centre point on the diagram shown is row 6, column 11.

Assume:

- the user may only touch one area at a time
- screen rotation does not affect the touchscreen.

The programmer has decided to use global values `CentreRow` and `CentreCol` as coordinate values for the centre point.

The programmer has started to define program modules as follows:

Module	Description
<code>FirstRowSet()</code>	<ul style="list-style-type: none"> • Searches for the first row that has an array element set to 1 • Returns the index of that row (1 is the first row) • Returns -1 if there are no elements set to 1
<code>LastRowSet()</code>	<ul style="list-style-type: none"> • Searches for the last row that has an array element set to 1 • Returns the index of that row • Returns -1 if there are no elements set to 1
<code>FirstColSet()</code>	<ul style="list-style-type: none"> • Searches for the first column that has an array element set to 1 • Returns the index of that column (1 is the first column) • Returns -1 if there are no elements set to 1
<code>LastColSet()</code>	<ul style="list-style-type: none"> • Searches for the last column that has an array element set to 1 • Returns the index of that column • Returns -1 if there are no elements set to 1

(b) Describe a feature of your solution to **part (a)** that indicates the pseudocode represents an efficient algorithm.

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

(c) The programmer decides to produce a **single** search module `FindSet()`, which will be able to perform each of the individual searches performed by the first four modules in the table.

(i) Outline the changes needed to convert one of the existing modules into this single module.

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

(ii) Give one possible advantage **and** one possible disadvantage of combining the four searches into a single module.

Advantage

.....

Disadvantage

..... [2]

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.



Cambridge International AS & A Level

CANDIDATE
NAME

--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/23

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 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 **20** pages. Any blank pages are indicated.

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

1 Sylvia is testing a program that has been written by her colleague. Her colleague tells her that the program does not contain any syntax errors.

(a) (i) State what her colleague means by “does not contain any syntax errors”.

.....

 [1]

(ii) Identify **and** describe **one** other type of error that the program may contain.

Type of error

Description

..... [2]

(b) Complete the following table by giving the appropriate data type in each case.

Use of variable	Data type
The average mark in a class of 40 students	
An email address	
The number of students in the class	
To indicate whether an email has been read	

[4]

(c) An airline wants to provide passengers with information about individual flights and allow them to book their flight using an online booking system.

(i) Tick (✓) **one** box in each row of the table to indicate whether each item of information would be essential for the customer when making the booking.

Information	Essential	Not essential
Departure time		
Flight number		
Departure airport		
Aircraft type		
Ticket price		
Number of seats in aircraft		

[3]

(ii) Identify the technique used to filter out information that is not essential when designing the booking system **and** state one benefit of this technique.

Technique

Benefit

.....

[2]

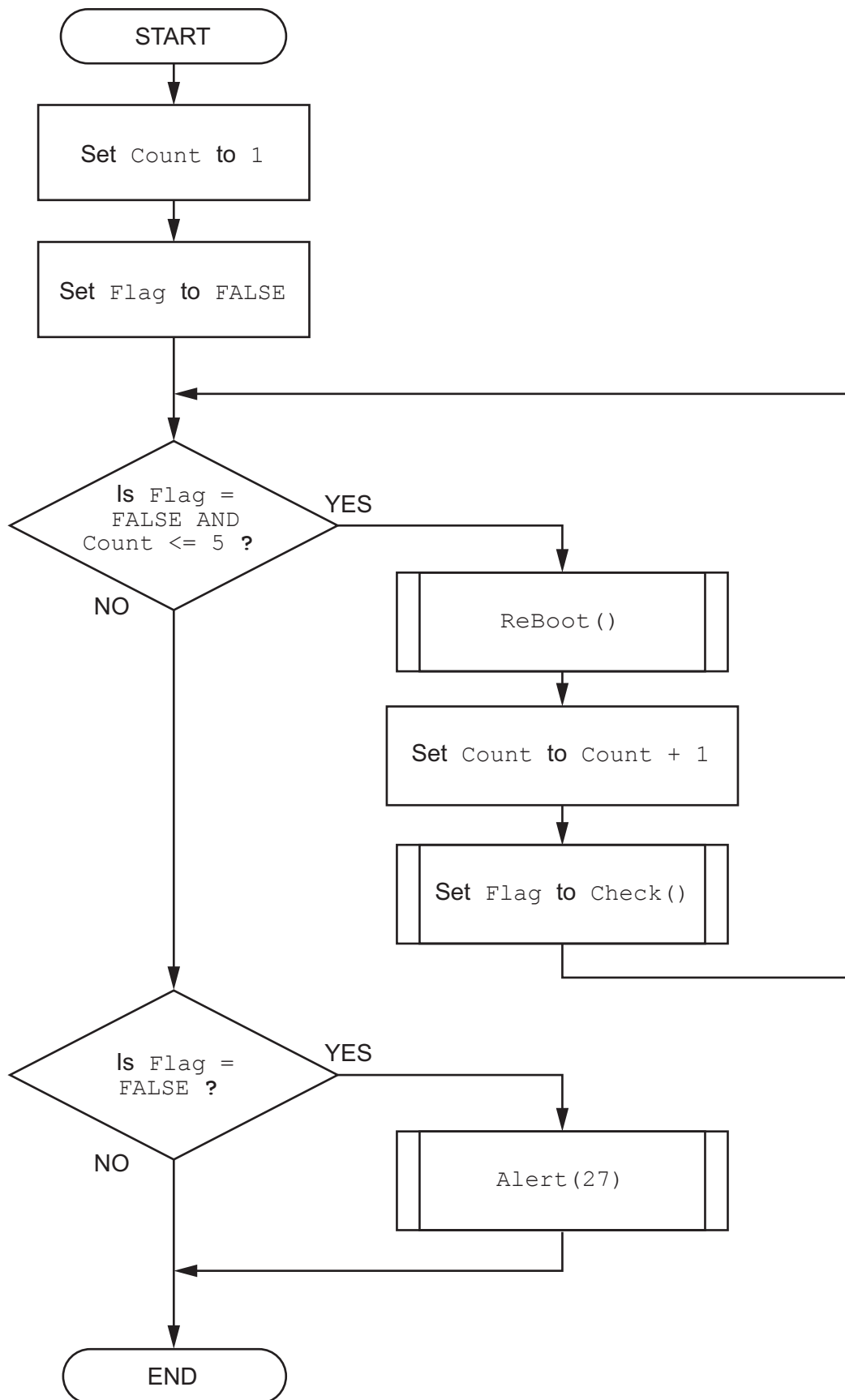
(iii) Identify **two additional** pieces of essential information that a passenger might need when booking a flight.

1

2

[2]

(b) The program flowchart shown describes a simple algorithm.



- 3 (a) The diagram below represents a queue Abstract Data Type (ADT) that can hold a maximum of eight items.

The operation of this queue may be summarised as follows:

- The front of queue pointer points to the next item to be removed.
- The end of queue pointer points to the last item added.
- The queue is circular so that empty storage elements can be reused.

0	Frog	← Front of queue pointer
1	Cat	
2	Fish	
3	Elk	← End of queue pointer
4		
5		
6		
7		

- (i) Describe how “Octopus” is added to the given queue.

.....

.....

.....

..... [2]

- (ii) Describe how the next item in the given queue is removed and stored in the variable `AnimalName`.

.....

.....

.....

..... [2]

- (iii) Describe the state of the queue when the **front of queue** and the **end of queue** pointers have the same value.

.....

..... [1]

(b) Some operations are carried out on the original queue given in **part (a)**.

(i) The current state of the queue is:

0	Frog
1	Cat
2	Fish
3	Elk
4	
5	
6	
7	

Complete the diagram to show the state of the queue after the following operations:

Add “Wasp”, “Bee” and “Mouse”, and then remove two data items.

[3]

(ii) The state of the queue after other operations are carried out is shown:

0	Frog	
1	Cat	
2	Fish	
3	Elk	← Front of queue pointer
4	Wasp	
5	Bee	
6	Mouse	← End of queue pointer
7	Ant	

Complete the following diagram to show the state of the queue after the following operations:

Remove one item, and then add “Dolphin” and “Shark”.

0	
1	
2	
3	
4	
5	
6	
7	

[2]

(c) The queue is implemented using a 1D array.

Describe the algorithm that should be used to modify the **end of queue pointer** when adding an item to the queue.

Your algorithm should detect any potential error conditions.

.....

.....

.....

.....

.....

.....

..... [3]

4 A program controls the heating system in a sports hall.

Part of the program involves reading a value from a sensor. The sensor produces a numeric value that represents the temperature. The value is an integer, which should be in the range 0 to 40 inclusive.

A program function has been written to validate the values from the sensor.

(a) A test plan is needed to test the function.

Complete the table. The first line has been completed for you.

You can assume that the sensor will generate only integer data values.

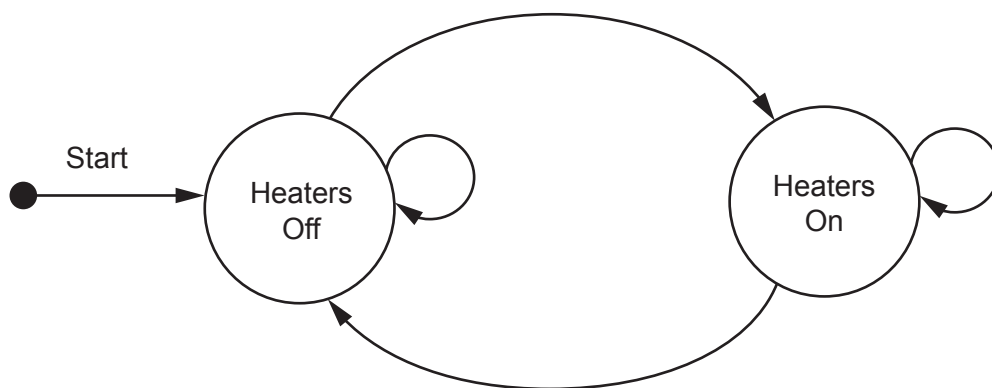
Test	Test data value	Explanation	Expected outcome
1	23	Normal data	Data is accepted
2			
3			
4			
5			

[4]

(b) A program module controls the heaters. This module operates as follows:

- If the temperature is below 10, switch the heaters on.
- If the temperature is above 20, switch the heaters off.

Complete the following state-transition diagram for the heating system:



[3]

- 5 The following data items will be recorded each time a student successfully logs on to the school network:

Data item	Example data
Student ID	"CJL404"
Host ID	"Lib01"
Time and date	"08:30, June 01, 2021"

The Student ID is six characters long. The other two data items are of variable length.

A single string will be formed by concatenating the three data items. A separator character will need to be inserted between items two and three.

For example:

```
"CJL404Lib01<separator>08:30, June 01, 2021"
```

Each string represents one log entry.

A programmer decides to store the concatenated strings in a 1D array `LogArray` that contains 2000 elements. Unused array elements will contain an empty string.

- (a) Suggest a suitable separator character **and** give a reason for your choice.

Character

Reason

..... [2]

- (b) The choice of data structure was made during one stage of the program development life cycle.

Identify this stage.

..... [1]

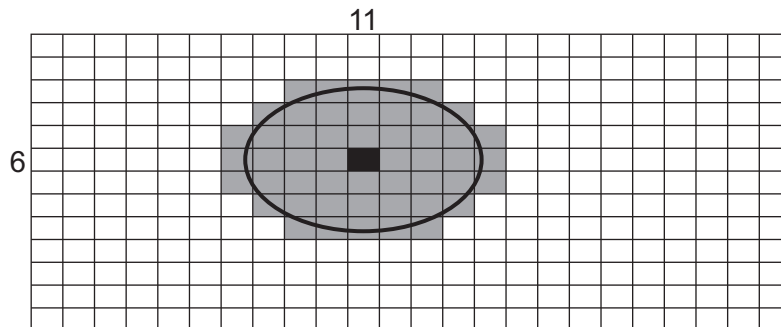
- 6 A mobile phone has a touchscreen. The screen is represented by a grid, divided into 800 rows and 1280 columns.

The grid is represented by a 2D array `Screen` of type `INTEGER`. An array element will be set to 0 unless the user touches that part of the screen.

Many array elements are set to 1 by a single touch of a finger or a stylus.

The following diagram shows a simplified touchscreen. The dark line represents a touch to the screen. All grid elements that are wholly or partly inside the outline will be set to 1. These elements are shaded.

The element shaded in black represents the centre point.



A program is needed to find the coordinates (the row and column) of the centre point. The centre point on the diagram above is row 6, column 11.

Assume:

- the user may only touch one area at a time
- screen rotation does not affect the touchscreen.

The programmer has started to define program modules as follows:

Module	Description
<code>SetRow()</code> (generates test data)	<ul style="list-style-type: none"> • Called with three parameters of type <code>INTEGER</code>: <ul style="list-style-type: none"> ◦ a row number ◦ the number of pixels to be skipped starting from column 1 ◦ the number of pixels that should be set to 1 • Sets the required number of pixels to 1 <p>For example, <code>SetRow(3, 8, 5)</code> will give row 3 as in the diagram shown.</p>
<code>SearchInRow()</code>	<ul style="list-style-type: none"> • Takes two parameters of type <code>INTEGER</code>: <ul style="list-style-type: none"> ◦ a row number ◦ a start column (1 or 1280) • Searches the given row from the start column (either left to right or right to left) for the first column that contains an element set to 1 • Returns the column number of the first element in the given row that is set to 1 • Returns -1 if no element is set to 1
<code>SearchInCol()</code>	<ul style="list-style-type: none"> • Takes two parameters of type <code>INTEGER</code>: <ul style="list-style-type: none"> ◦ a column number ◦ a start row (1 or 800) • Searches the given column from the start row (either up or down) for the first row that contains an element set to 1 • Returns the row number of the first element in the given column that is set to 1 • Returns -1 if no element is set to 1

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.



Cambridge International AS & A Level

CANDIDATE
NAME

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/21

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 2022

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.

(ii) The following pseudocode statement includes array references:

```
OUTPUT "Student ", Name[Count], " scored ", Mark[Count]
```

State the purpose of the variable `Count` and give its data type.

Purpose

.....

Data type

[2]

(c) The pseudocode statements in the following table may contain errors.

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

Assume that any variables used are of the correct type for the given function.

Statement	Error
IF EMPTY ← "" THEN	
Status ← IS_NUM(-23.4)	
X ← STR_TO_NUM("37") + 5	
Y ← STR_TO_NUM("37" + "5")	

[4]

- 2 A system is being developed to help manage a car hire business. A customer may hire a car for a number of days.

An abstract model needs to be produced.

- (a) Explain the process of abstraction **and** state **four** items of data that should be stored each time a car is hired.

Explanation

.....

Item 1

Item 2

Item 3

Item 4

[3]

- (b) Identify **two** operations that would be required to process the car hire data.

Operation 1

.....

Operation 2

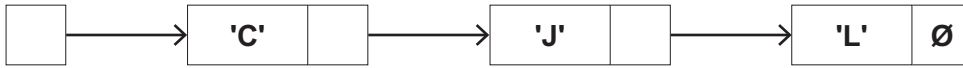
.....

[2]

- 4 (a) The following diagram shows an Abstract Data Type (ADT) representation of an ordered linked list. The data item stored in each node is a single character. The data will be accessed in alphabetical order.

The symbol \emptyset represents a null pointer.

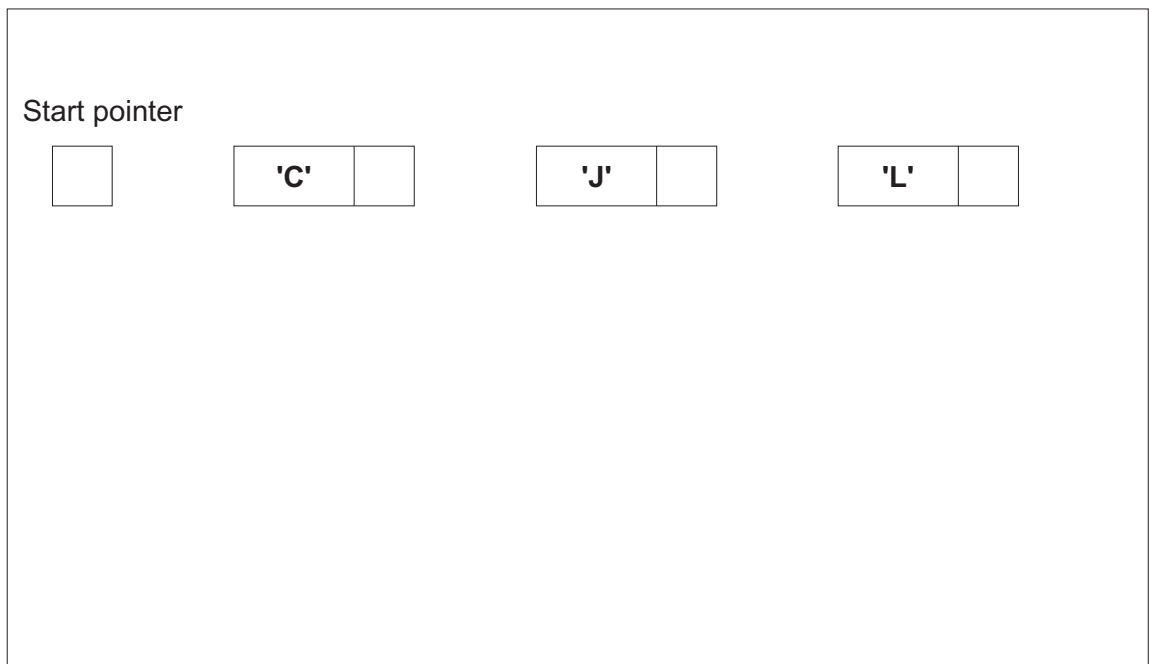
Start pointer



- (i) Nodes with data 'A' and 'K' are added to the linked list. Nodes with data 'J' and 'L' are deleted.

After the changes, the data items still need to be accessed in alphabetical order.

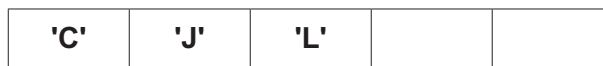
Complete the diagram to show the new state of the linked list.



[4]

- (ii) The original data could have been stored in a 1D array in which each element stores a character.

For example:



Explain the advantages of making the changes described in **part (a)(i)** when the data is stored in the linked list instead of an array.

.....

.....

.....

..... [2]

(iii) Explain the disadvantages of making the changes described in **part (a)(i)** when the data is stored in the linked list instead of an array.

.....

.....

.....

..... [2]

(b) A program will store data using a linked list like the one shown in **part (a)**.

Explain how the linked list can be implemented.

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

BLANK PAGE

- 6 The following pseudocode algorithm attempts to check whether a string is a valid email address.

```

FUNCTION IsValid(InString : STRING) RETURNS BOOLEAN
  DECLARE Index, Dots, Ats, Others : INTEGER
  DECLARE NextChar : CHAR
  DECLARE Valid : BOOLEAN

  Index ← 1
  Dots ← 0
  Ats ← 0
  Others ← 0
  Valid ← TRUE

  REPEAT
    NextChar ← MID(InString, Index, 1)
    CASE OF NextChar
      '.' : Dots ← Dots + 1
      '@' : Ats ← Ats + 1
            IF Ats > 1 THEN
              Valid ← FALSE
            ENDIF
      OTHERWISE : Others ← Others + 1
    ENDCASE

    IF Dots > 1 AND Ats = 0 THEN
      Valid ← FALSE
    ELSE
      Index ← Index + 1
    ENDIF

  UNTIL Index > LENGTH(InString) OR Valid = FALSE

  IF NOT (Dots >= 1 AND Ats = 1 AND Others > 8) THEN
    Valid ← FALSE
  ENDIF

  RETURN Valid

ENDFUNCTION

```

- (a) Part of the validation is implemented by the line:

```
IF NOT (Dots >= 1 AND Ats = 1 AND Others > 8) THEN
```

State the values that would result in the condition evaluating to TRUE.

.....

.....

..... [1]

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

Result ← IsValid("Liz.123@big@net")

Index	NextChar	Dots	Ats	Others	Valid

[5]

(ii) State the value returned when IsValid() is called using the expression shown in part (b)(i).

..... [1]

.....

.....

.....

..... [7]

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.



Cambridge International AS & A Level

CANDIDATE
NAME

--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/22

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 2022

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) A programmer is developing an algorithm to solve a problem. Part of the algorithm would be appropriate to implement as a subroutine (a procedure or a function).

(i) State **two** reasons why the programmer may decide to use a subroutine.

1

.....

2

.....

[2]

(ii) A procedure header is shown in pseudocode:

```
PROCEDURE MyProc (Count : INTEGER, Message : STRING)
```

Give the correct term for the identifiers `Count` and `Message` **and** explain their use.

Term

Use

.....

.....

.....

[2]

(b) The algorithm in **part (a)** is part of a program that will be sold to the public. All the software errors that were identified during in-house testing have been corrected.

Identify **and** describe the additional test stage that may be carried out before the program is sold to the public.

Test stage

Description

.....

.....

.....

.....

.....

[4]

(ii) Explain why it may be better to store the names of the students in a file rather than in an array.

.....

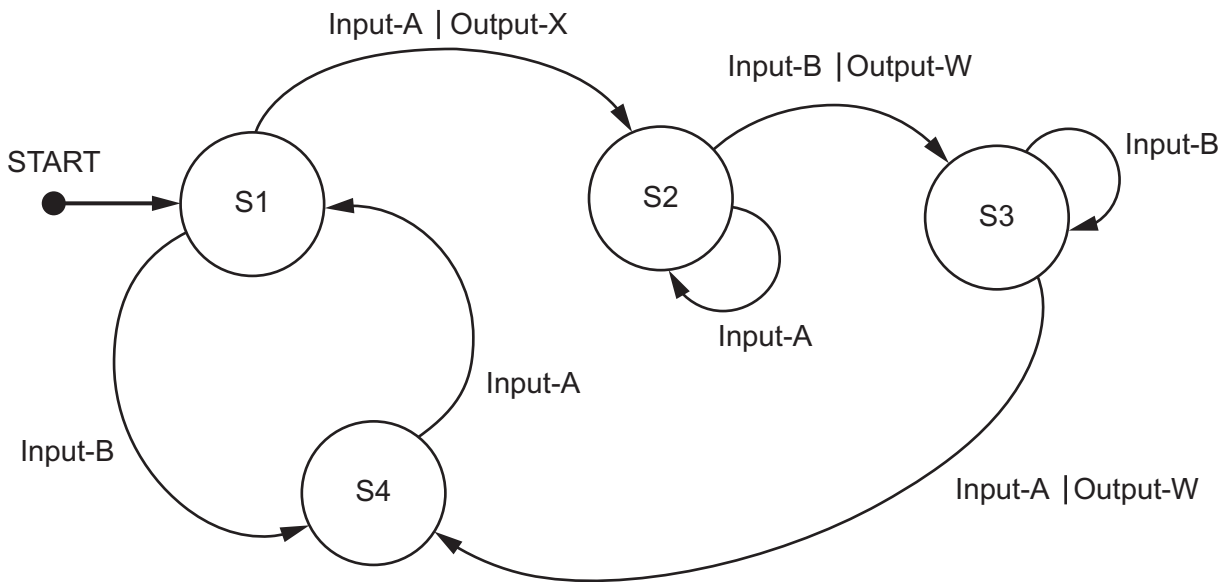
 [1]

(iii) Explain why `WRITE` mode cannot be used in the answer to **part 2(a)(i)**.

.....

 [1]

(b) Examine the following state-transition diagram.



Complete the table to show the inputs, outputs and next states.

Input	Output	Next state
		S1
Input-A		
		S2
	Output-W	
	Output-W	

[4]

- (c) A second stack is used in the program. The diagram below shows the initial state of this stack. Value X is at the top of the stack and was the last item added.

Upper-case letters are used to represent different data values.

Stack operations are performed in three groups as follows:

Group 1:

PUSH D
PUSH E

Group 2:

POP
POP
POP

Group 3:

PUSH A
PUSH B
POP
PUSH C

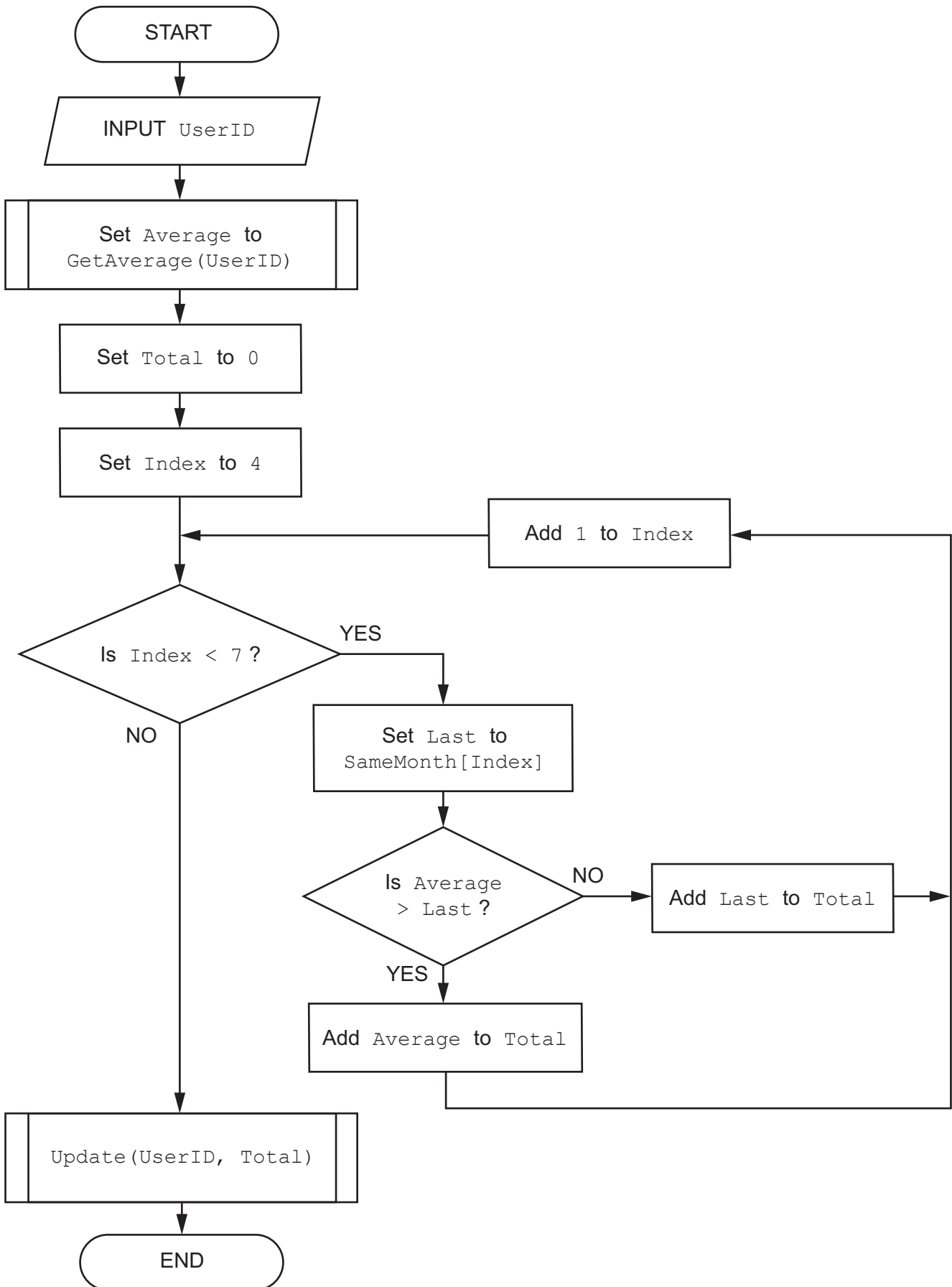
Complete the diagram to show the state of the stack **after** each group of operations has been performed.

Include the current stack pointer (SP) **after** each group.

Memory location	Initial state	After Group 1	After Group 2	After Group 3
957				
956				
955				
954				
953	X ← SP			
952	Y			
951	Z			
950	P			

[5]

4 The program flowchart represents a simple algorithm.



(a) Write the equivalent pseudocode for the algorithm represented by the flowchart.

..... [6]

(b) Give the name of the iterative construct in the flowchart.

..... [1]

5 Examine the following pseudocode.

```

IF A = TRUE THEN
  IF B = TRUE THEN
    IF C = TRUE THEN
      CALL Sub1()
    ELSE
      CALL Sub2()
    ENDIF
  ENDIF
ELSE
  IF B = TRUE THEN
    IF C = TRUE THEN
      CALL Sub4()
    ELSE
      CALL Sub3()
    ENDIF
  ELSE
    IF C = FALSE THEN
      CALL Sub3()
    ELSE
      CALL Sub4()
    ENDIF
  ENDIF
ENDIF
ENDIF

```

A programmer wants to re-write the pseudocode as **four** separate IF...THEN...ENDIF statements, each containing a single CALL statement. This involves writing a single, simplified logic expression as the condition in each statement.

Write the amended pseudocode.

1

.....

.....

.....

2

.....

.....

.....

3

.....

.....

.....

4

.....

.....

.....

[4]

- 6 (a) The factorial of an integer number is the product of all the integers from that number down to 1.

In general, the factorial of n is $n \times (n-1) \times \dots \times 2 \times 1$

For example, the factorial of 5 is $5 \times 4 \times 3 \times 2 \times 1 = 120$

In this question, n will be referred to as the `BaseNumber`.

A function `FindBaseNumber()` will:

- be called with a positive, non-zero integer value as a parameter
- return `BaseNumber` if the parameter value is the factorial of the `BaseNumber`
- return `-1` if the parameter value is **not** a factorial.

For example:

Parameter value	Value returned
120	5
12	-1
6	3
1	1

`FindBaseNumber(12)` will return `-1` because 12 is not a factorial.

You may use the rest of this page for rough working.

(b) A program is written to allow a user to input a sequence of values to be checked using the function `FindBaseNumber()`.

The user will input one value at a time. The variable used to store the user input has to be of type string because the user will input 'End' to end the program.

Valid input will be converted to an integer and passed to `FindBaseNumber()` and the return value will be output.

Complete the table by giving **four** invalid strings that may be used to test distinct aspects of the required validation. Give the reason for your choice in each case.

Input	Reason for choice



[4]

- 7 A teacher is designing a program to perform simple syntax checks on programs written by students.

Two global 1D arrays are used to store the syntax error data. Both arrays contain 500 elements.

- Array `ErrCode` contains integer values that represent an error number in the range 1 to 800.
- Array `ErrText` contains string values that represent an error description.

The following diagram shows an example of the arrays.

Index	ErrCode	ErrText
1	10	"Invalid identifier name"
2	20	"Bracket mismatch"
3	50	"Undeclared variable"
4	60	"Type mismatch in assignment"
...		
500	999	<Undefined>

Note:

- There may be less than 500 error numbers so corresponding elements in both arrays may be unused. Unused elements in `ErrCode` have the value 999. The value of unused elements in `ErrText` is undefined.
- Values in the `ErrCode` array are stored in ascending order but not all values may be present, for example, there may be no error code 31.

The teacher has defined two program modules as follows:

Module	Description
<code>OutputError()</code>	<ul style="list-style-type: none"> • takes two parameters as integers: <ul style="list-style-type: none"> ○ a line number in the student's program ○ an error number • searches for the error number in the <code>ErrCode</code> array: <ul style="list-style-type: none"> ○ if found, outputs the corresponding error description and the line number, for example: "Bracket mismatch on line 34" ○ if not found, outputs the line number and a warning, for example: "Unknown error on line 34"
<code>SortArrays()</code>	sorts the arrays into ascending order of <code>ErrCode</code>

(c) Two 1D arrays were described at the beginning of the question. Both arrays contain 500 elements.

- Array `ErrCode` contains integer values that represent an error number in the range 1 to 800.
- Array `ErrText` contains string values that represent an error description.

The two arrays will be replaced by a single array. A user-defined data type (record structure) has been declared as follows:

```

TYPE ErrorRec
    DECLARE ErrCode : STRING
    DECLARE ErrText : STRING
ENDTYPE
    
```

(i) State the error in the record declaration.

.....
 [1]

(ii) State **two** benefits of using the single array of the user-defined data type.

1

 2
 [2]

(iii) Write the declaration for the single array in pseudocode.

..... [1]

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.



Cambridge International AS & A Level

CANDIDATE
NAME

--

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/23

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 2022

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 program is required for a shopping website.

(a) Part of the program requires four variables. The following table describes the use of each variable.

Complete the table by adding the most appropriate data type for each variable.

Variable use	Data type
Store the number of days in the current month	
Store the first letter of the customer's first name	
Store an indication of whether a year is a leap year	
Store the average amount spent per customer visit	

[4]

(b) The designer considers the use of a development life cycle to split the development of the website into several stages.

(i) State **one** benefit of a development life cycle when developing the website.

.....
 [1]

(ii) Analysis is one stage of a development life cycle.

State **one** document that may be produced from the analysis stage of the website project.

.....
 [1]

(c) The program will be developed using the Rapid Application Development (RAD) life cycle.

(i) State **one** principle of this life cycle.

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

(ii) Give **two** benefits and **one** drawback of its use compared to the waterfall life cycle.

Benefit 1
.....
Benefit 2
.....
Drawback
..... [3]

(d) Adaptive maintenance needs to be carried out on the website program.

Give **two** reasons why adaptive maintenance may be required.

1
.....
2
..... [2]

- 2 A program is being designed for a smartphone to allow users to send money to the charity of their choice.

Decomposition will be used to break the problem down into sub-problems.

Identify **three** program modules that could be used in the design **and** describe their use.

Module 1

Use

.....

.....

.....

Module 2

Use

.....

.....

.....

Module 3

Use

.....

.....

.....

[3]

4 (a) A program contains a 1D array `DataItem` with 100 elements.

State the **one additional** piece of information required before the array can be declared.

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

(b) A programmer decides to implement a queue Abstract Data Type (ADT) in order to store characters received from the keyboard. The queue will need to store at least 10 characters and will be implemented using an array.

(i) Describe **two** operations that are typically required when implementing a queue. State the check that must be carried out before each operation can be completed.

Operation 1
.....
Check 1
.....
Operation 2
.....
Check 2
..... [4]

(ii) Describe the declaration and initialisation of the variables and data structures used to implement the queue.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
..... [5]

(b) The design changes and a record structure is defined to store the three data items.

A user-defined data type `StockItem` is created as shown:

```
TYPE StockItem
  DECLARE StockID : STRING
  DECLARE Description : STRING
  DECLARE Cost : REAL
ENDTYPE
```

(i) A variable `LineData` of type `StockItem` is declared.

Write the pseudocode statement to assign the value 12.99 to the `Cost` field of `LineData`.

..... [1]

(ii) Procedure `Unpack()` is modified and converted to a function which takes the original text string as the only parameter.

Explain the other changes that need to be made to convert the procedure into a function.

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

(c) `Unpack()` is part of a program made up of several modules. During the design stage, it is important to follow good programming practice. One example of good practice is the use of meaningful identifier names.

Give the reason why this is good practice. Give **two other** examples of good practice.

Reason

.....

.....

Example 1

.....

.....

Example 2

.....

.....

[3]

(d) The program that includes `Unpack()` is tested using the walkthrough method.

Describe this method **and** explain how it can be used to identify an error.

.....

.....

.....

.....

.....

.....

.....

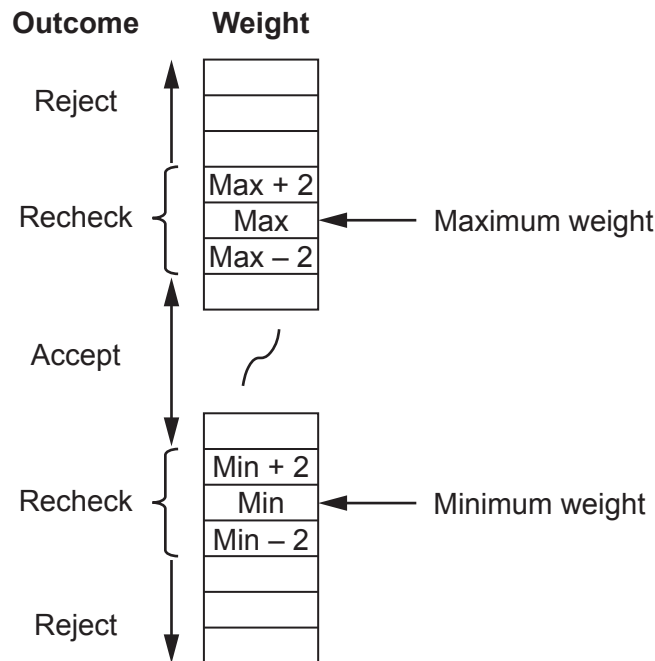
[3]

6 Components are weighed during manufacture. Weights are measured to the nearest whole gram.

Components that weigh at least 3 grams more than the maximum weight, or at least 3 grams less than the minimum weight, are rejected.

A component is rechecked if it weighs within 2 grams of either the maximum or minimum weight.

The final outcome of weighing each component is shown below:



A function `Status()` will be called with three parameters. These are integers representing the weight of an individual component together with the minimum and maximum weights.

The value returned from the function will be as follows:

Outcome	Return value
Accept	'A'
Reject	'R'
Recheck	'C'

(a) Complete the following test plan for **five** tests that could be performed on function `Status()`. The tests should address all possible outcomes.



Test number	Component weight	Min	Max	Expected return value
1				'A'
2				
3				
4				
5				

- 7 A teacher is designing a program to perform simple syntax checks on programs written by students.

Two global 1D arrays are used to store the syntax error data. Both arrays contain 500 elements.

- Array `ErrCode` contains integer values that represent an error number in the range 1 to 800.
- Array `ErrMsgText` contains string values that represent an error description.

The following diagram shows an example of the arrays.

Index	ErrCode	ErrMsgText
1	10	"Invalid identifier name"
2	20	"Bracket mismatch"
3	50	""
4	60	"Type mismatch in assignment"
...		
500	999	<Undefined>

Note:

- There are less than 500 error codes so corresponding elements in both arrays may be unused. Unused elements in `ErrCode` have the value 999. These will occur at the end of the array. The value of unused elements in `ErrMsgText` is undefined.
- Values in the `ErrCode` array are stored in ascending order but not all values may be present. For example, there may be no error code 31.
- Some error numbers are undefined. In these instances, the `ErrCode` array will contain a valid error number but the corresponding `ErrMsgText` element will contain an empty string.

The teacher has defined one program module as follows:

Module	Description
<code>OutputRange()</code>	<ul style="list-style-type: none"> • Prompts for input of two error numbers • Outputs a list of error numbers between the two numbers input (inclusive) together with the corresponding error description • Outputs a warning message when the error description is missing as for error number 50 in the example • Outputs a suitable header and a final count of error numbers found <p>Output based on the example array data above:</p> <pre>List of error numbers from 1 to 60 10 : Invalid identifier name 20 : Bracket mismatch 50 : Error Text Missing 60 : Type mismatch in assignment 4 error numbers output</pre>

Question 7 continues on the next page.

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]

(ii) A new Module `RemoveError()` will remove a given error number from the array.

Describe the algorithm that would be required. Do **not** include pseudocode statements in your answer.

.....

.....

.....

.....

.....

.....

.....

.....

..... [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.



Cambridge International AS & A Level

CANDIDATE
NAME

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/21

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 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 The following pseudocode represents part of the algorithm for a program:

```

CASE OF ThisValue
  < 30      : Level ← "Low"           // less than 30
             Check ← 1
  < 20      : Level ← "Very Low"     // less than 20
             Check ← ThisValue / 2
  30 TO 40  : Level ← "Medium"       // between 30 and 40
             Check ← ThisValue / 3
             Data[ThisValue] ← Data[ThisValue] + 1
  > 40      : Level ← "High"
ENDCASE
    
```

(a) Complete the table by writing the answer for each row:

	Answer
The value assigned to <code>Level</code> when <code>ThisValue</code> is 40	
The value assigned to <code>Check</code> when <code>ThisValue</code> is 36	
The value assigned to <code>Level</code> when <code>ThisValue</code> is 18	
The number of elements in array <code>Data</code> that may be incremented	

[4]

(b) The pseudocode contains four assignments to variable `Level`. One of these assignments will never be performed.

Identify this assignment **and** explain why this is the case.

.....

.....

.....

.....

.....

.....

..... [3]

(c) The following line is added immediately before the `ENDCASE` statement:

```
OTHERWISE : Level ← "Undefined"
```

State why this assignment is never performed.

.....

..... [1]

(d) Give the appropriate data types for the variables `ThisValue`, `Check` and `Level`.

`ThisValue`

`Check`

`Level`

[3]

2 (a) An algorithm is expressed as follows:

- input 100 numbers, one at a time
- keep a total of all numbers input that have a value between 30 and 70 inclusive and output this total after the last number has been input.

Outline, using stepwise refinement, the five steps for this algorithm which could be used to produce pseudocode.

Do **not** use pseudocode statements in your answer.

Step 1

.....

Step 2

.....

Step 3

.....

Step 4

.....

Step 5

.....

[5]

(b) Sequence is one programming construct.

Identify **two other** programming constructs that will be required when the algorithm is converted into pseudocode.

Construct 1

.....

Construct 2

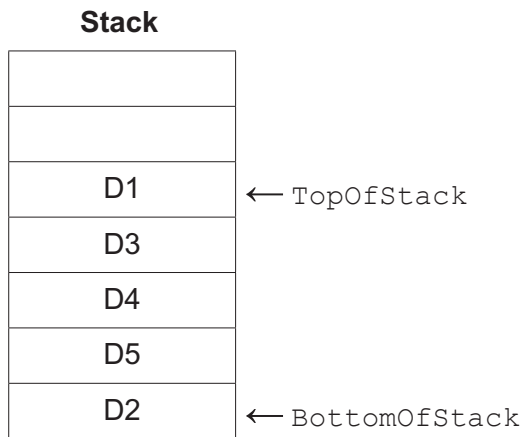
.....

[2]

3 The diagram represents an Abstract Data Type (ADT).

The operation of this stack may be summarised as follows:

- The `TopOfStack` pointer points to the last item added to the stack.
- The `BottomOfStack` pointer points to the first item on the stack.

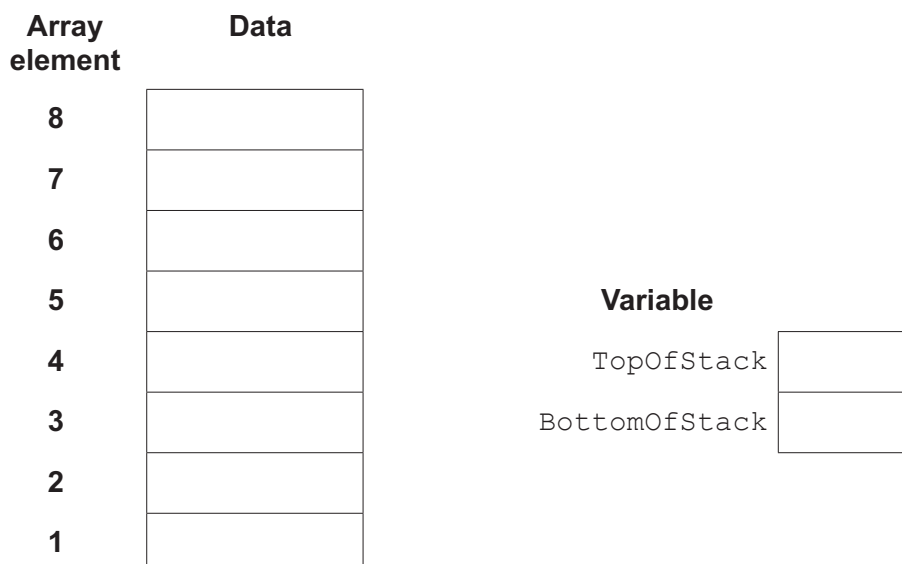


(a) The stack is implemented using two variables and a 1D array of 8 elements as shown.

The variables are used to reference individual elements of the array, in such a way that:

- the array is filled from the lowest indexed element towards the highest
- all the elements of the array are available for the stack.

Complete the diagram to represent the state of the stack as shown above.



[3]

- (b) A function `Push()` will add a value onto the stack by manipulating the array and variables in part (a).

Before adding a value onto the stack, the algorithm will check that space is available.

If the value is added to the stack, the function will return `TRUE`, otherwise it will return `FALSE`.

The algorithm is expressed in five steps.

Complete the steps.

1. If then return `FALSE`
2. Otherwise `TopOfStack`
3. Use `TopOfStack` as an to the array.
4. Set the element at this to the being added.
5. Return

[5]

(b) The global array is changed to a 2D array, organised as 150 rows by 2 columns. It is declared in pseudocode as follows:

```
DECLARE Data : ARRAY[1:150, 1:2] OF STRING
```

The algorithm for the function in **part (a)** is changed. Strings will only be counted if **both** of the following conditions are true:

- The current row is an even number.
- The search string exactly matches the value in **either** column.

Write pseudocode to check these conditions.

Assume that the row index is contained in variable `Row` and the search string in variable `Search`.

.....

.....

.....

.....

.....

.....

.....

..... [3]

- 5 An algorithm is designed to find the smallest numeric value from an input sequence and count how many numeric values have been input.
An example of an input sequence is:

23, AB56, 17, 23ZW, 4, 10, END

Numeric input values are all integers and non-numeric input is ignored, except for the string "END" which is used to terminate the sequence.

The algorithm is expressed in pseudocode as shown:

```

DECLARE NextInput : STRING
DECLARE Min, Count, Num : INTEGER

Min ← 999
Count ← 0

REPEAT
  INPUT NextInput
  IF IS_NUM(NextInput) = TRUE THEN
    Num ← STR_TO_NUM(NextInput)
    IF Num > Min THEN
      Min ← Num
    ENDIF
    Count ← Count & 1
  ENDIF
UNTIL NextInput ← "END"

OUTPUT "The minimum value is ", Min, " and the count was ", Count

```

- (a) The pseudocode contains three errors due to the incorrect use of **operators**.

Identify each error **and** state the correction required.

1

.....

2

.....

3

.....

[3]

(b) The operator errors are corrected and the algorithm is tested as follows:

The input sequence:

18, 4, ONE, 27, 189, ERIC, 3, 65, END

produces the output:

The minimum value is 3 and the count was 6

The algorithm is tested with a different test data sequence. The sequence contains a mix of integer and non-numeric values. It is terminated correctly but the algorithm produces unexpected results.

(i) Explain the problem with the algorithm.

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

(ii) Give a sequence of **four** test data values that could be input to demonstrate the problem.

Value 1
Value 2
Value 3
Value 4 [2]

6 The pseudocode `OUTPUT` command starts each output on a new line.

(a) A new procedure `MyOutput()` will take a string and a Boolean parameter.

`MyOutput()` may be called repeatedly and will use concatenation to build a string using a global variable `MyString`, up to a maximum length of 255 characters.

`MyString` will be output in either of these two cases:

1. The Boolean parameter value is `TRUE`
2. The resulting string (after concatenation) would be longer than 255 characters.

If `MyString` is not output, the string is concatenated with `MyString`.

For example, the calls to `MyOutput()` given below would result in the output as shown:

```
MyOutput("Hello ", FALSE)
MyOutput("ginger ", FALSE)
MyOutput("cat", TRUE)
MyOutput("How are you?", TRUE)
```

Resulting output:

```
Hello ginger cat
How are you?
```

Notes:

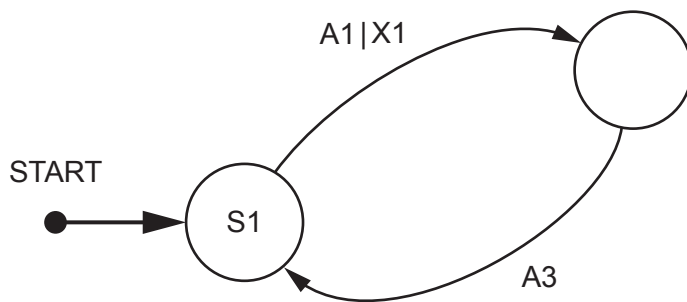
- `MyString` is initialised to an empty string before `MyOutput()` is called for the first time.
- No string passed to `MyOutput()` will be longer than 255 characters.

7 An algorithm is represented by a state-transition diagram.

The table shows the inputs, outputs and states for the algorithm:

Current state	Input	Output	Next state
S1	A1	X1	S2
S2	A3	none	S1
S2	A2	X4	S5
S5	A1	X1	S5
S5	A4	X2	S2
S5	A3	none	S3
S1	A9	X9	S3
S3	A9	X9	S4

Complete the state-transition diagram to represent the information given in the table.



[5]

- 8 A class of students are developing a program to send data between computers. Many computers are connected together to form a wired network. Serial ports are used to connect one computer to another.

Each computer:

- is assigned a unique three-digit ID
- has three ports, each identified by an integer value
- is connected to between one and three other computers.

Messages are sent between computers as a string of characters organised into fields as shown:

```
<STX><DestinationID><SourceID><Data><ETX>
```

Field number	Field name	Description
n/a	STX	a single character marking the start of the message (ASCII value 02)
1	DestinationID	three numeric characters that identify the destination computer
2	SourceID	three numeric characters that identify the source computer
3	Data	a variable length string containing the data being sent (Minimum length is 1 character)
n/a	ETX	a single character marking the end of the message (ASCII value 03)

For example, the following message contains the data "Hello Kevin" being sent from computer "101" to computer "232":

```
<STX>"232101Hello Kevin"<ETX>
```

Each computer will run a copy of the same program. Each program will contain a global variable, `MyID` of type string, that contains the unique ID of the computer in which the program is running.

The programmer has defined the first two program modules as follows:

Module	Description
<code>Transmit()</code> (already written)	<ul style="list-style-type: none"> • takes two parameters: <ul style="list-style-type: none"> ○ a string containing a message ○ an integer containing a port number • transmits the message using the given port
<code>SendFile()</code>	<ul style="list-style-type: none"> • takes three parameters: <ul style="list-style-type: none"> ○ a string containing a text file name ○ a string containing a Destination ID ○ an integer containing a Port number • transmits the file one line at a time • transmits a final message with data string "*****"

(b) Module `SendFile()` is used to copy a file from one computer to another.

A module within the program running on the destination computer will receive the data and write it to a new file.

Explain why module `SendFile()` transmits the message with data string "*****" after the last line of the file.

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

(c) One of the text files to be sent contains several blank lines (lines that do not contain any text).

(i) Explain why this is a problem.

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

(ii) Explain how the message format could be changed to allow a blank line to be sent.

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

Question 8(d) starts on page 18.

(d) A new module has been defined:

Module	Description
GetField()	<ul style="list-style-type: none"> • takes two parameters: <ul style="list-style-type: none"> ○ a string containing a message ○ an integer containing a field number • If the field number is valid (in the range 1 to 3, inclusive), it returns a string containing the required field, otherwise it returns an empty string.

As a reminder, a message is defined as follows:

<STX><DestinationID><SourceID><Data><ETX>

Field number	Field name	Description
Not applicable	STX	a single character marking the start of the message (ASCII value 02)
1	DestinationID	three numeric characters that identify the destination computer
2	SourceID	three numeric characters that identify the source computer
3	Data	a variable length string containing the data being sent (Minimum length is 1 character)
Not applicable	ETX	a single character marking the end of the message (ASCII value 03)

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.



Cambridge International AS & A Level

CANDIDATE
NAME

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/22

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 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 shop sells car accessories. A customer order is created if an item cannot be supplied from current stock. A program is being developed to create and manage the customer orders.

(a) The following identifier table shows some of the data that will be stored for each order.

Complete the identifier table by adding meaningful variable names and appropriate data types.

Example value	Explanation	Variable name	Data type
"Mr Khan"	The name of the customer		
3	The number of items in the order		
TRUE	To indicate whether this is a new customer		
15.75	The deposit paid by the customer		

[4]

(b) Other variables in the program have example values as shown:

Variable	Example value
Total	124.00
DepRate	2.00
Description	"AB12345:Cleaning Brush (small)"

Complete the table by evaluating each expression using the example values.

Expression	Evaluates to
$(\text{Total} * \text{DepRate}) + 1.5$	
<code>RIGHT(Description, 7)</code>	
$(\text{LENGTH}(\text{Description}) - 8) > 16$	
<code>NUM_TO_STR(INT(DepRate * 10)) & '%'</code>	

[4]

(c) The data that needs to be stored for each customer order in part (a) is not all of the same type.

Describe an effective way of storing this data for many customer orders while the program is running.

.....

.....

.....

.....

.....

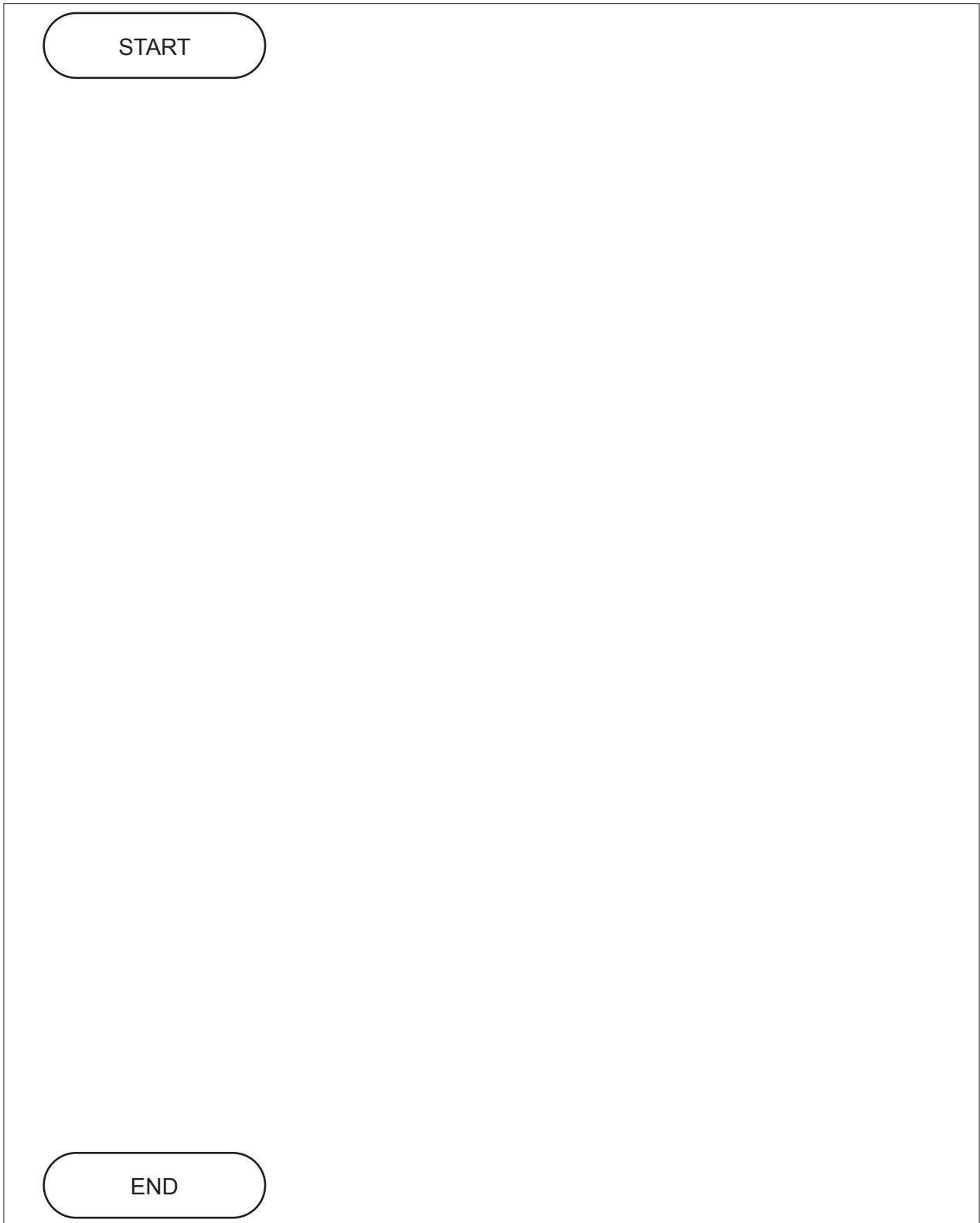
.....

..... [3]

2 An algorithm will:

1. input a sequence of integer values, one at a time
2. ignore all values until the value 27 is input, then sum the remaining values in the sequence
3. stop summing values when the value 0 is input and then output the sum of the values.

(a) Draw a program flowchart to represent the algorithm.



[5]

(b) The solution to the algorithm includes iteration.

Give the name of a suitable loop structure that could be used.

Justify your answer.

Name

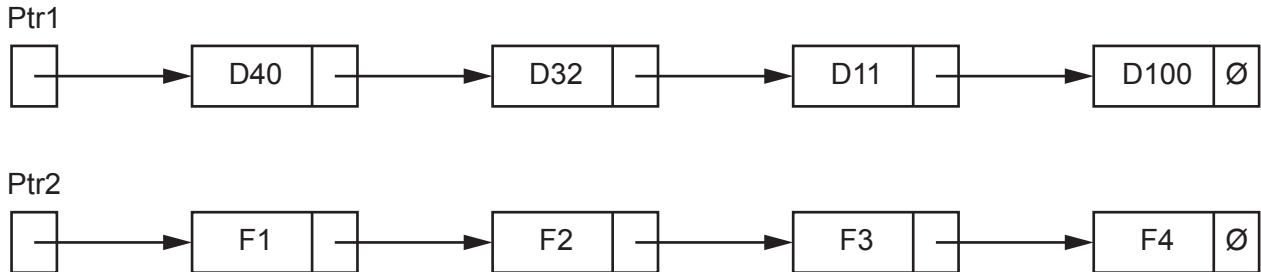
Justification

.....

[2]

3 The diagram represents a linked list Abstract Data Type (ADT).

- Ptr1 is the start pointer. Ptr2 is the free list pointer.
- Labels D40, D32, D11 and D100 represent the data items of nodes in the list.
- Labels F1, F2, F3 and F4 represent the data items of nodes in the free list.
- The symbol \emptyset represents a null pointer.



(a) The linked list is implemented using two variables and two 1D arrays as shown.

The pointer variables and the elements of the Pointer array store the indices (index numbers) of elements in the Data array.

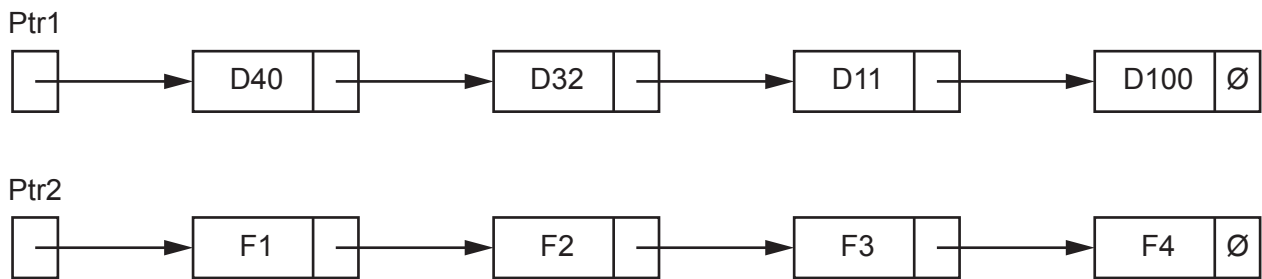
Complete the diagram to show how the linked list as shown above may be represented using the variables and arrays.

Variable	Value
Start_Pointer	
Free_List_Pointer	5

Index	Data array	Pointer array
1	D32	2
2		3
3		
4	D40	
5		
6	F2	7
7		
8		

[5]

- (b) The original linked list is to be modified. A new node D6 is inserted between nodes D32 and D11.



The algorithm required is expressed in four steps as shown.

Complete the steps.

1. Assign the data item to
2. Set the of this node to point to
3. Set Ptr2 to point to
4. Set pointer of to point to

[4]

(b) The procedure `Count()` is to be tested.

Typical test data would consist of odd and even values, for example:

23, 5, 64, 100, 2002, 1, 8, 900, 99

The purpose of this test would be to test a typical mix of even and odd values and check the totals.

Give **three** test data sequences that may be used to test **different** aspects of the procedure.

Do **not** include invalid data.

Sequence 1:

Test data

Purpose of test.

.....

Sequence 2:

Test data

Purpose of test.

.....

Sequence 3:

Test data

Purpose of test.

.....

[3]

- 5 A global 1D array of integers contains four elements, which are assigned values as shown:

```
Mix[1] ← 1
Mix[2] ← 3
Mix[3] ← 4
Mix[4] ← 2
```

A procedure `Process()` manipulates the values in the array.

The procedure is written in pseudocode:

```
PROCEDURE Process(Start : INTEGER)
  DECLARE Value, Index, Count : INTEGER

  Index ← Start
  Count ← 0

  REPEAT
    Value ← Mix[Index]
    Mix[Index] ← Mix[Index] - 1
    Index ← Value
    Count ← Count + 1
  UNTIL Count = 5

  Mix[4] ← Count * Index

ENDPROCEDURE
```

Complete the trace table on the opposite page by dry running the procedure when it is called as follows:

```
CALL Process(3)
```


(b) A module `CheckFiles()` will count the number of files produced by `CreateFiles()` in part (a).

`CheckFiles()` will take a string representing a file name and return the number of files found.

(i) Identify the type of module that should be used for `CheckFiles()`.

..... [1]

(ii) Write the module header for `CheckFiles()`.

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

(iii) State the file mode that should be used in `CheckFiles()`.

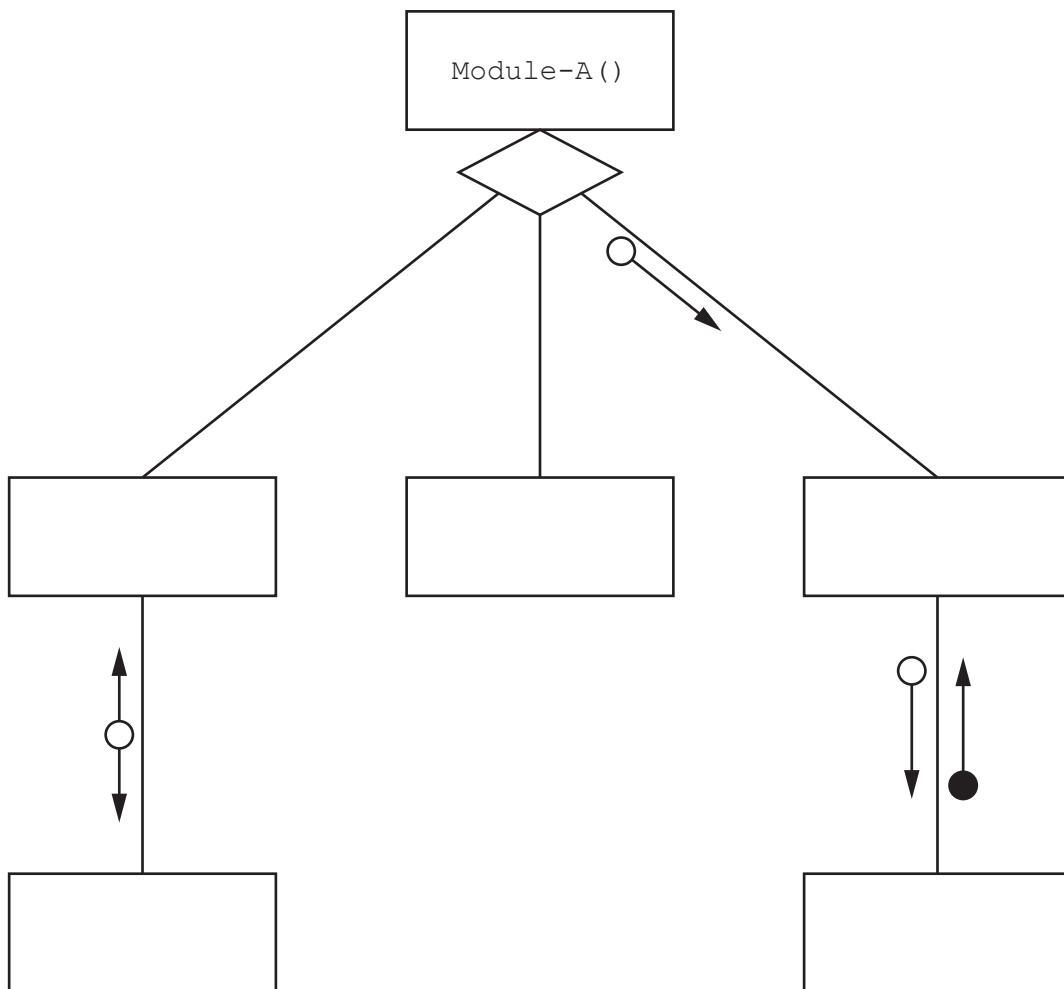
..... [1]

7 A program contains six modules:

Pseudocode module header
PROCEDURE Module-A()
PROCEDURE Module-X(T1 : INTEGER, S2 : REAL)
PROCEDURE Reset(BYREF Code : INTEGER)
FUNCTION Restore(OldCode : INTEGER) RETURNS BOOLEAN
FUNCTION Module-Y(RA : INTEGER, RB : BOOLEAN) RETURNS BOOLEAN
FUNCTION Module-Z(SA : INTEGER) RETURNS INTEGER

Module-X() **calls** Reset()
 Module-Y() **calls** Restore()

(a) Complete the structure chart for these modules.



[4]

(b) Explain the meaning of the diamond symbol as used in the diagram in part (a).

.....
 [2]

- 8 A class of students are developing a program to send data between computers. Many computers are connected together to form a wired network. Serial ports are used to connect one computer to another.

Each computer:

- is assigned a unique three-digit ID
- has three ports, each identified by an integer value
- is connected to between one and three other computers.

Data is sent as individual message strings.

Each string contains the destination ID (the ID of the computer that is to receive the message) followed by the data:

```
<DestinationID><Data>
```

Messages may pass through several computers on the way to their destination.

When a message arrives at a computer, that is **not** the destination, the program needs to forward it on to another computer using one of its serial ports.

The port to use is obtained from information that is stored in an array `RouteTable`.

`RouteTable` is a global 2D array of integers. It is declared in pseudocode as follows:

```
DECLARE RouteTable : ARRAY[1:6,1:3] OF INTEGER
```

The values in the first two columns of `RouteTable` define a range of ID values.

Column 3 gives the corresponding port number to use when forwarding the message to a computer with an ID within this range.

For example, the contents of `RouteTable` could be:

	Column 1	Column 2	Column 3
Row 1	100	199	1
Row 2	200	259	2
Row 3	-1	<undefined>	<undefined>
Row 4	260	399	2
Row 5	400	599	3
Row 6	600	999	1

In this example, a message that arrives with a `DestinationID` of "283" will be forwarded using port 2.

Row 3 in the example shows an unused row. These may occur anywhere. Unused rows have the column 1 element set to -1. The value of unused elements in the other two columns is undefined.

- (b) Copies of the same program will run on each computer. The program contains a global variable `MyID` of type string, which contains the unique ID of the computer in which the program is running.

When messages are received, they are placed on one of two stacks. Stack 1 is used for messages that have reached their destination and stack 2 is used for messages that will be forwarded on to another computer.

Additional modules are defined:

Module	Description
<code>StackMsg()</code> (already written)	<ul style="list-style-type: none"> • takes two parameters: <ul style="list-style-type: none"> ○ a string representing a message ○ an integer representing the stack number • adds the message to the required stack • returns <code>TRUE</code> if the message is added to the required stack, otherwise returns <code>FALSE</code>
<code>ProcessMsg()</code>	<ul style="list-style-type: none"> • takes a message as a parameter of type string • ignores any message with a zero-length data field • extract the <code>DestinationID</code> from the message • checks whether the <code>DestinationID</code> is this computer or whether the message is to be forwarded • uses <code>StackMsg()</code> to add the message to the appropriate stack • outputs an error if the message could not be added to the stack

- (c) The program contains a module `GetFile()` which receives text files sent from another computer.

Lines from the file are sent one at a time. Each message contains one line and `ProcessMsg()` from part (b) adds each message as it is received onto stack 1.

Module `GetFile()` removes messages from stack 1 and writes the data to a text file.

There is a problem. Under certain circumstances, the received file does not appear as expected.

Assume that while a file is being received `ProcessMsg()` receives only messages containing lines from the file.

- (i) Describe the circumstances and explain the problem.

Circumstances

.....

Explanation

.....

.....

.....

[3]

- (ii) Suggest a more appropriate Abstract Data Type that could be used to store the messages that would not have the same problem.

..... [1]

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.



Cambridge International AS & A Level

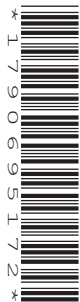
CANDIDATE
NAME

CENTRE
NUMBER

--	--	--	--	--

CANDIDATE
NUMBER

--	--	--	--



COMPUTER SCIENCE

9618/23

Paper 2 Fundamental Problem-solving and Programming Skills

October/November 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.

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

- 1 A program is being developed in pseudocode before being converted into a programming language.

- (a) The following table shows four valid pseudocode assignment statements.

Complete the table by giving the data type that should be used to declare the variable underlined in each assignment statement.

Assignment statement	Data type
<u>MyVar1</u> ← Total1 / Total2	
<u>MyVar2</u> ← 27/10/2023	
<u>MyVar3</u> ← "Sum1 / Sum2"	
<u>MyVar4</u> ← Result1 AND Result2	

[4]

- (b) Other variables in the program have example values as shown:

Variable	Value
Active	TRUE
Fraction	0.2
Code	"Ab12345"

Complete the table by evaluating each expression using the example values.

Expression	Evaluates to
Fraction >= 0.2 AND NOT Active	
INT((Fraction * 100) + 13.3)	
STR_TO_NUM(MID(Code, 4, 2)) + 5	
LENGTH("TRUE" & Code)	

[4]

- (c) The program makes use of complex statistical functions. The required functions are not built-in to the programming language and are too complicated for the programmer to write.

One solution would be to employ another programmer who has experience of writing these functions, as there is no time to train the existing programmer.

State **one other** way that these functions may be provided for inclusion in the program.

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

- (d) The hardware that runs the program is changed and the program needs to be modified so that it works with the new hardware.

Identify the type of maintenance that this represents **and** give **one other** reason why this type of maintenance may be needed.

Type

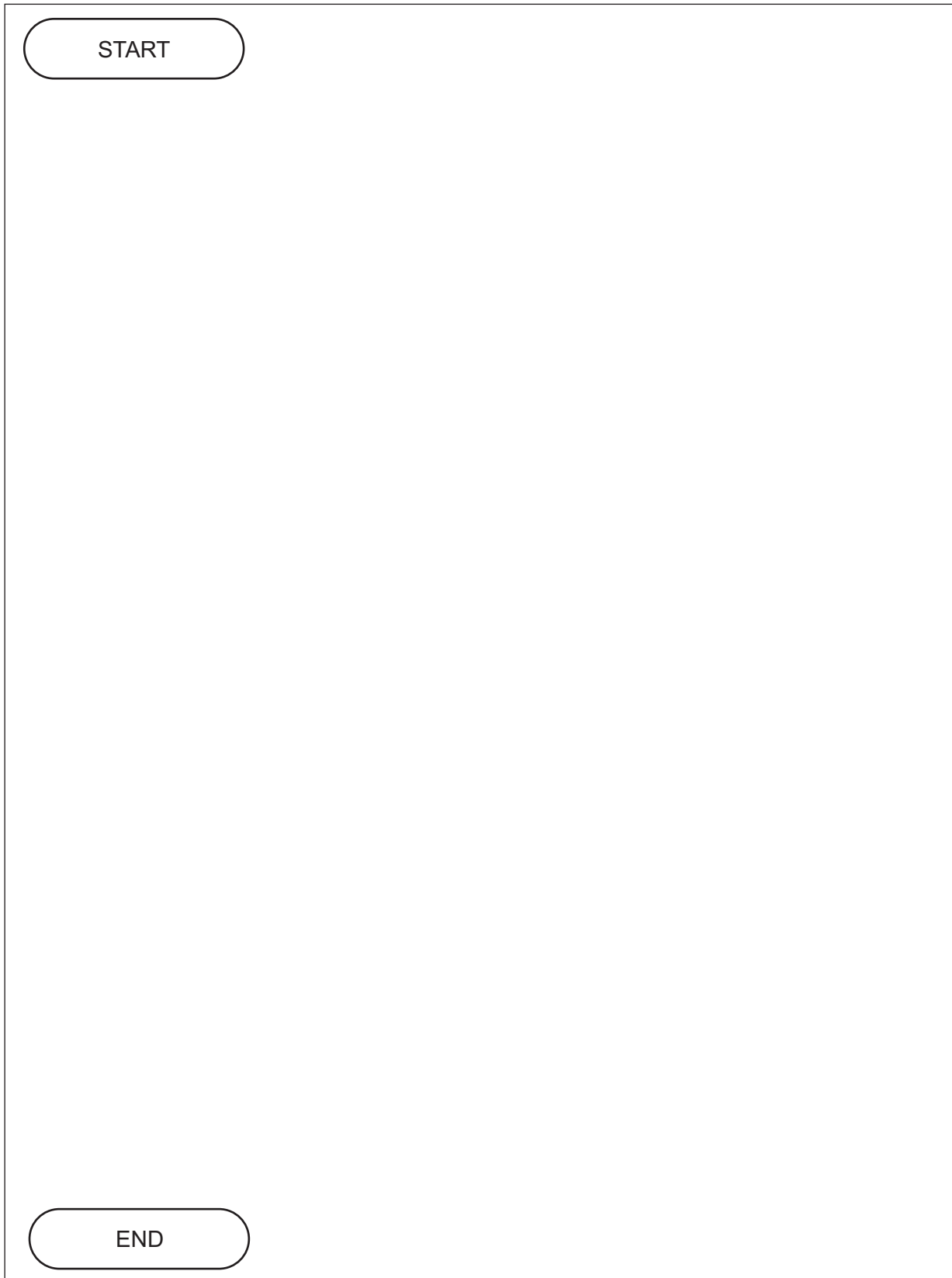
Reason

..... [2]

2 Data is a 1D array of integers, containing 30 elements. All element values are unique.

(a) An algorithm will output the index of the element with the smallest value.

Draw a program flowchart to represent the algorithm.



[5]

(b) The 30 data values could have been stored in separate variables rather than in an array.

Explain the benefits of using an array when designing a solution to part (a).

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

(c) The requirement changes. Array `Data` needs to hold 120 elements and each value may include a decimal place.

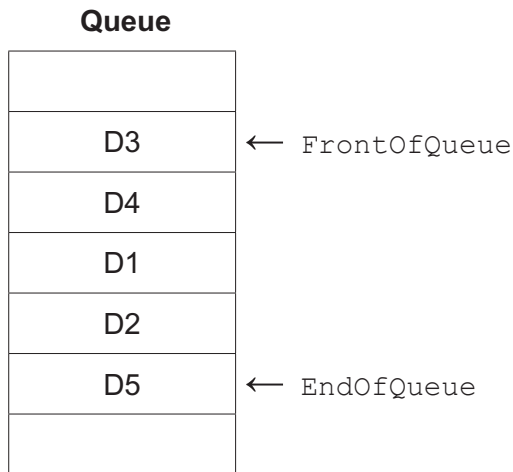
Write a pseudocode statement to declare the modified array.

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

3 The diagram represents a queue Abstract Data Type (ADT).

The organisation of this queue may be summarised as follows:

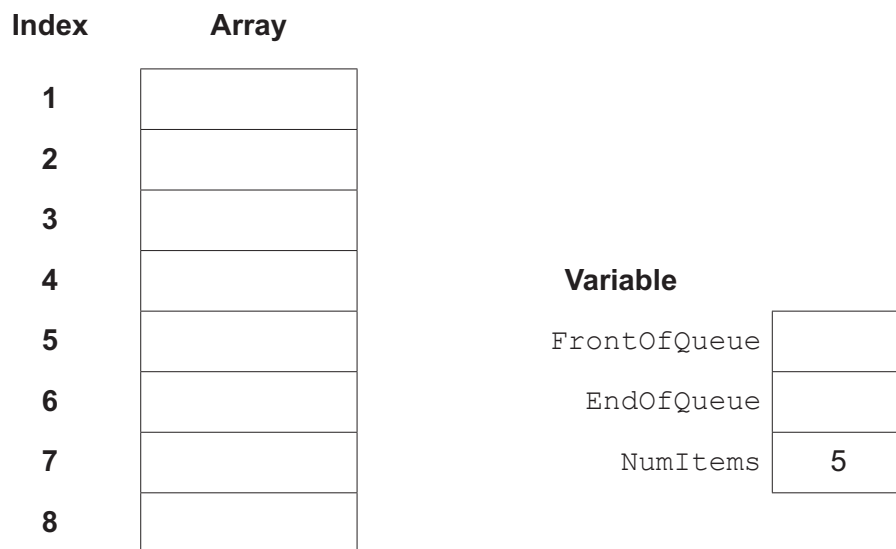
- The `FrontOfQueue` pointer points to the next data item to be removed.
- The `EndOfQueue` pointer points to the last data item added.



The queue is implemented using three variables and a 1D array of eight elements as shown. The variable `NumItems` stores the number of items in the queue.

The pointer variables store indices (index numbers) of the array.

(a) Complete the diagram to represent the state of the queue as shown above.



[3]

- (b) A module `AddTo()` will add a value to the queue by manipulating the array and variables in part (a).

The queue implementation is circular. When pointers reach the end of the queue, they will 'wrap around' to the beginning.

Before a value can be added to the queue, it is necessary to check the queue is not full.

The algorithm to add a value to the queue is expressed in six steps.

Complete the steps.

1. If `NumItems` then jump to step 6.
2. Increment
3. If then set `EndOfQueue` to
4. Increment
5. Set the at the index stored in to the being added.
6. Stop.

[6]

(b) Procedure `RandList()` is modified so that the random numbers are also written into a 1D array `Result`.

A new module is written to confirm that the numbers in the array are in ascending order.

This module contains an `IF` statement that performs a comparison between elements:

```
IF (Result[x + 1] = Result[x]) OR (Result[x] > Result[x + 1]) THEN
    Sequence ← FALSE
ENDIF
```

Write a simplified version of the conditional clause.

.....

.....

..... [1]

- 5 A global 1D array of integers contains four elements, which are assigned values as shown:

```
Mix[1] ← 4
Mix[2] ← 2
Mix[3] ← 3
Mix[4] ← 5
```

A procedure `Process()` manipulates the values in the array.

The procedure is written in pseudocode as follows:

```
PROCEDURE Process(Start : INTEGER)
  DECLARE Value, Index, Total : INTEGER

  Index ← Start
  Total ← 0

  WHILE Total < 20
    Value ← Mix[Index]
    Total ← Total + Value

    IF Index < 4 THEN
      Mix[Index] ← Mix[Index] + Mix[Index+1]
    ELSE
      Mix[Index] ← Mix[Index] + Mix[1]
    ENDIF
    Index ← (Value MOD 4) + 1

  ENDWHILE

  Mix[1] ← Total * Index

ENDPROCEDURE
```

Complete the trace table on the opposite page by dry running the procedure when it is called as follows:

```
CALL Process(2)
```


6 A function `TestNum()` will take a six-digit string as a parameter.

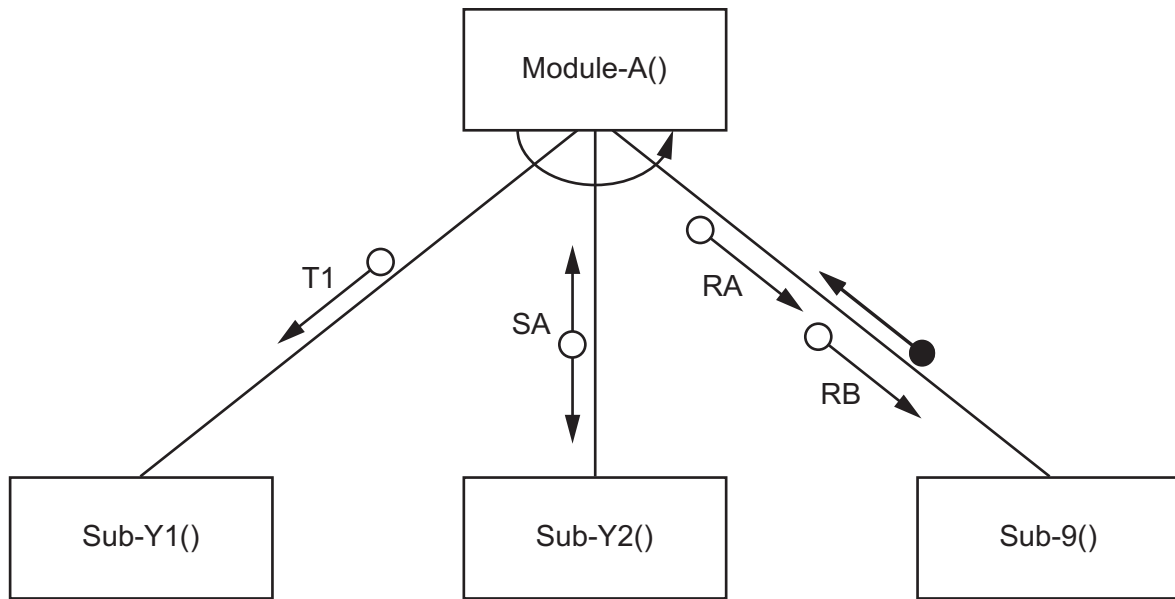
The function will test whether the string meets certain conditions and will return an integer value as follows:

Return value	Condition	Example
1	The last three digits are the same but non-zero.	"253444"
2	The last three digits are zero.	"253000"
3	The first three and last three digits are the same.	"410410"

The function will return the highest possible value for the given string.

If the string does not meet any of the conditions, zero is returned.

7 A structure chart shows the modular structure of a program:



(a) Explain the meaning of the curved arrow symbol which begins and ends at Module-A().

.....

.....

.....

..... [2]

(b) The structure chart shows that Sub-9() is a function.

A Boolean value is returned by Sub-9() for processing by Module-A().

The original parameter RA is of type integer and RB is of type string.

A record type `MyType` will be defined with three fields to store the values passed between the two modules.

(i) Write pseudocode to define `MyType`.

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

(ii) The design is modified and Sub-9() is changed to a procedure.

The procedure will be called with a single parameter of type `MyType`.

Write the pseudocode header for procedure `Sub-9 ()`.

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

- 8 A class of students are developing a program to send data between computers. Many computers are connected together to form a wired network. Serial ports are used to connect one computer to another.

Each computer:

- is assigned a unique three-digit ID
- has three ports, each identified by an integer value
- is connected to between one and three other computers.

Messages are sent between computers as a string of characters organised into fields as shown:

```
<STX><DestinationID><SourceID><Data><ETX>
```

Field name	Description
STX	a single character marking the start of the message (ASCII value 02)
DestinationID	three numeric characters identifying the destination computer
SourceID	three numeric characters identifying the source computer
Data	a variable length string containing the data being sent (Minimum length is 1 character)
ETX	a single character marking the end of the message (ASCII value 03)

For example, the following message contains the data "Hello Jack" being sent from computer "202" to computer "454":

```
<STX>"454202Hello Jack"<ETX>
```

Each computer will run a copy of the same program. Each program will contain a global variable `MyID` of type string which contains the unique ID of the computer in which the program is running.

The first two program modules are defined as follows:

Module	Description
<code>GetData()</code> (already written)	<ul style="list-style-type: none"> • returns the data field from a message that has been received • If no message is available, the module waits until one has been received.
<code>ReceiveFile()</code>	<ul style="list-style-type: none"> • takes a file name as a parameter of type string • creates a text file with the given file name (no checking required) • writes the data field returned by <code>GetData()</code> to the file • repeats until the data field is "****", which is not written to the file • outputs a final message giving the total number of characters written to the file, for example: 132456 characters were written to newfile.txt

- (b) The use of the string "*****" as explained in the module description for `ReceiveFile()` may cause a problem.

Explain the problem and suggest a solution.

Problem

.....

.....

.....

Solution

.....

.....

.....

[3]

- (c) Two new modules are defined, which will allow two users to exchange messages.

Module	Description
<code>Transmit()</code> (already written)	<ul style="list-style-type: none"> • takes two parameters: <ul style="list-style-type: none"> ○ a string representing a message ○ an integer representing a port number • transmits the message using the given port
<code>Chat()</code>	<ul style="list-style-type: none"> • takes two parameters: <ul style="list-style-type: none"> ○ a string representing a Destination ID ○ an integer representing a port number • extracts data from a received message using <code>GetData()</code> and outputs it • forms a message using data input by the user and sends it using <code>Transmit()</code> • repeats until either the output string or the sent string is "Bye"

Reminders:

- Each program contains a global variable `MyID` of type string which contains the unique ID of the computer in which the program is running.
- Messages are sent between computers as a string of characters organised into fields as shown:

<STX><DestinationID><SourceID><Data><ETX>

- (d) Module `GetData()` returns the data field from a message that has been received. If no message is available, the module waits until one has been received.

Explain the limitation of this on module `Chat()` from part (c).

Describe a modification to `GetData()` to address this limitation.

Limitation

.....

.....

.....

Modification

.....

.....

.....

[3]

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.