* 0000800000001 *

Cambridge Assessment International Education

Cambridge IGCSE™

CANDIDATE NAME

CENTRE NUMBER | | | | | |

CANDIDATE NUMBER | | | | |

6508712232

COMPUTER SCIENCE **0478/21**

Paper 2 Algorithms, Programming and Logic **October/November 2024**

1 hour 45 minutes

You must answer on the question paper.

No additional materials are needed.

INSTRUCTIONS

- Answer **all** questions.
- Use a black or dark blue pen. You may use an HB pencil for any diagrams or graphs.
- 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.
- 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.

This document has **16** pages. Any blank pages are indicated.

© UCLES 2024

[Turn over

1 Tick (✓) **one** box to show which check is used for verification when data is input.

A length check ☐

B range check ☐

C type check ☐

D visual check ☐

[1]

2 Tick (✓) **one** box to identify which option is used to show the inputs and output of a Boolean expression.

A flowchart ☐

B trace table ☐

C truth table ☐

D variable ☐

[1]

3 **Four** operators and **three** types of operator are shown.

Draw **one or more** lines from each operator to its correct operator type.

Operator	Operator type
>=	Boolean
AND	Arithmetic
DIV	Logical
+	

[4]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

4 Identify **three** stages of the program development life cycle from the following list of words.

analysis **decomposition** **design** **input**

pseudocode **testing** **variable**

1 .....

2 .....

3 .....

[3]

5 Describe **three** methods that are used to design and construct a solution to a problem.

Method 1 .....

.....

.....

.....

Method 2 .....

.....

.....

.....

Method 3 .....

.....

.....

.....

[6]

DO NOT WRITE IN THIS MARGIN

6 An incomplete algorithm has been written in pseudocode to count the number of zeros stored in an array and total the non-zero values.

```
01 DECLARE A[1:50] : INTEGER
02 DECLARE C : INTEGER
03 DECLARE I : INTEGER
04 DECLARE T : INTEGER
05 I ← 0
06 .....
07 FOR C ← 1 TO 50
08     IF A[C] .....
09       THEN
10         T ← T + 1
11       ELSE
12         I ← I + A[C]
13     ENDIF
14 .....
```

(a) Complete the given pseudocode algorithm. [3]

(b) Write the pseudocode to display, with suitable messages, the number of zeros stored in the array and the total of the non-zero values.

.....

.....

.....

.....

.....

..... [3]

(c) Meaningful identifiers have **not** been used in the algorithm.
Suggest suitable meaningful identifiers for:

The array:

A .....

The variables:

T .....

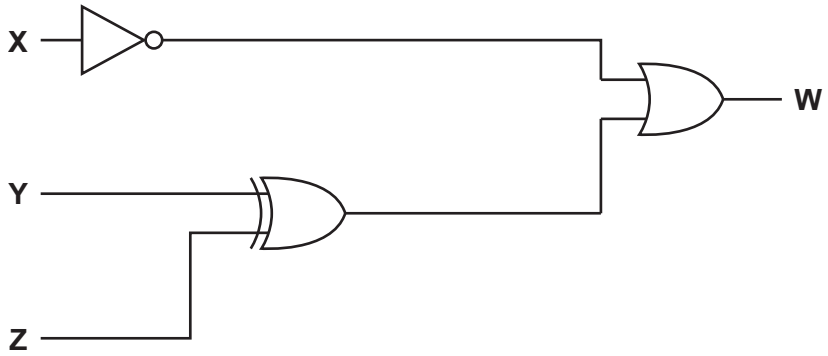
C .....

I .....

[3]

DO NOT WRITE IN THIS MARGIN

7 Consider the logic circuit:



(a) Write a logic expression for the given logic circuit. Do **not** attempt to simplify the logic expression.

W = ...

..... [3]

(b) Complete the truth table from the given logic circuit.

X	Y	Z	Working space	W
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

[4]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

8 A programmer is designing a program to check the length of a password and to check if the password input is the same as the stored password.

The program requirements are:
- input the password, `Password`
- check if there are at least 8 characters in the password
- check that the password is **not** the same as the stored password `OldPass`
- output 'accepted' if both tests are completed successfully
- otherwise, output 'rejected'.

Use the variable names given.

(a) Complete the flowchart for the program.





[6]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

(b) The accepted password, `Password`, is to be written to the file `MyPassword.txt`

Write pseudocode to:

- open the file
- write the accepted password to the file
- close the file.

.....

.....

.....

.....

..... [3]

(c) Explain why the accepted password needs to be stored in a file.

.....

.....

.....

..... [2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

9 An algorithm has been written in pseudocode to check if a temperature is in a given range. The temperature values used in the algorithm are correct.

```
01 REPEAT
02     OUTPUT "Please enter temperature "
03     INPUT Temp
04     IF Temperature = 999
05       THEN
06         IF Temperature > 38.0
07           THEN
08             OUTPUT "Temperature too high"
09         ENDIF
10         IF Temperature < 35.0
11           THEN
12             OUTPUT "Temperature too low"
13         ENDIF
14         IF Temperature >= 35.0 OR Temperature <= 38.0
15           THEN
16             OUTPUT "Temperature normal"
17         ENDIF
18     ENDIF
19 WHILE Temperature = 999
```

(a) Identify the line numbers of **four** errors in the pseudocode and suggest a correction for each error.

Error 1 line number .....

Correction .....

.....

Error 2 line number .....

Correction .....

.....

Error 3 line number .....

Correction .....

.....

Error 4 line number .....

Correction .....

.....

[4]

DO NOT WRITE IN THIS MARGIN

(b) Identify the temperature range used.

.....

.....

..... [2]

(c) Complete the trace table for the **corrected** algorithm using this data:

34.22, 36.1, 37.4, 38.0, 999, –1

Temperature	OUTPUT

[2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

10 A shop that sells cheese has set up a new database table called `CheeseStock` to store details of the cheeses available for sale. Part of this table is given.

ChNo	Name	InStock	SupplierCode	PricePerKg	WeightKg
CH01	American	Yes	XYZ	4.50	20.0
CH02	Brie	Yes	XYZ	7.50	21.0
CH03	Burrata	No	IMP	13.75	0.0
CH04	Camembert	No	ABC	16.85	0.0
CH05	Cheddar	Yes	ABC	5.00	50.0
CH06	Comté	No	SPC	7.35	0.0
CH07	Cottage	Yes	XYZ	4.50	3.0
CH08	Cream	Yes	XYZ	5.50	6.5
CH12	Emmental	Yes	IMP	2.75	1.5
CH15	Feta	Yes	IMP	12.75	12.0
CH16	Fontina	Yes	SPC	15.99	1.2
CH17	Gorgonzola	Yes	SPC	15.25	0.3
CH19	Gouda	Yes	SPC	7.99	2.5
CH21	Gruyère	No	SPC	16.75	0.0
CH22	Halloumi	Yes	IMP	4.75	15.0
CH23	Havarti	No	SPC	6.75	0.0
CH27	Manchego	No	IMP	13.99	0.0
CH30	Manouri	No	IMP	18.50	0.0
CH31	Mascarpone	No	SPC	12.99	0.0

(a) State the number of records in this part of the database table.

..... [1]

(b) (i) Give the name of the field that would be used for the primary key.

..... [1]

(ii) State the reason for choosing this field for the primary key.

.....

..... [1]

DO NOT WRITE IN THIS MARGIN

(c) Write the output from this structured query language (SQL) statement.

```
SELECT ChNo, WeightKg
FROM CheeseStock
WHERE SupplierCode = 'ABC';
```

.....

.....

..... [2]

(d) **(i)** Complete this SQL statement to display only the name of all the cheeses that are out of stock.

```
SELECT .....
FROM .....
WHERE ..... ;
```

[3]

(ii) Explain how **one** of the lines in your statement in part **(d)(i)** could be changed to display the same information.

.....

.....

.....

..... [2]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

11 A running club has 200 members who compete in a 1-kilometre running competition every month.
Members' names are stored in the one-dimensional (1D) array `MemberName[]`
Each member's time, in seconds, for the 1-kilometre run will be stored in another one-dimensional (1D) array `MemberTime[]`
The position of each member's data in the two arrays is the same. For example, the member stored at index 10 in `MemberName[]` and at index 10 in `MemberTime[]` is the same.

The running club awards a small prize to the members who have the top three times.
The club also awards certificates to all members with a time under 240 seconds.

Write a program that meets the following requirements:

- allows members' times to be input twice and verifies that the inputs match
- sorts the arrays `MemberTime[]` and `MemberName[]` in ascending order of time
- outputs the member names and times of the members with the top three times and identifies them as First, Second and Third
- stores the names of all the members who will receive a certificate in the array `MemberCertificate[]`
- outputs a message stating the number of certificates to be printed.

You must use pseudocode or program code **and** add comments to explain how your code works.

You do **not** need to initialise the data in the array `MemberName[]`

You do **not** need to declare any arrays or variables; you may assume that this has already been done.

All inputs and outputs must contain suitable messages.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [15]

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

BLANK PAGE

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

BLANK PAGE

DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN
DO NOT WRITE IN THIS MARGIN

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.