* 0000800000001 *

Cambridge Assessment International Education

Cambridge IGCSE™

CANDIDATE NAME			
CENTRE NUMBER	□□□□□	CANDIDATE NUMBER	□□□□

2561265136

COMPUTER SCIENCE **0478/22**

Paper 2 Algorithms, Programming and Logic **February/March 2025**

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.

1 Tick (✓) **one** box to complete this sentence.

A standard method of solution used to find a piece of data in an array is a

A bubble sort. □

B counting routine. □

C linear search. □

D selection search. □

[1]

2 Tick (✓) **one** box to complete this sentence.

The pseudocode to store a hotel name held in the variable `Name` to a text file is

A `READFILE Hotels.txt, Name` □

B `WRITEFILE Hotels.txt, Name` □

C `WRITEFILE Name, Hotels.txt` □

D `STOREFILE Hotels.txt, Name` □

[1]

3 **Four** development life cycle stages and **five** descriptions are shown.

Draw **one** line from each stage to the most appropriate description.
Not all descriptions will be used.

Stage

- testing
- analysis
- coding
- design

Description

- identifying the problem and requirements
- reviewing the final solution to suggest further developments
- making sure the program code works as expected
- using structure diagrams, flowcharts and pseudocode to plan the solution
- using a programming language to create the solution

[4]

4 **(a)** A presence check is a type of validation check.

State the purpose of a presence check.

....

.... [1]

(b) A value input into a computer system must be an integer.

(i) Identify the validation check used to test whether the value input meets this requirement.

.... [1]

(ii) Write the pseudocode for an algorithm to allow a number to be input into an appropriate variable and check whether the number is an integer. If the number is an integer, it is accepted. If the number is **not** an integer, an error message is output. Re-input of the number is required until a valid number is input.

....

....

....

....

....

....

....

....

....

.... [5]

5 A program is written to only accept data that contains more than 9 characters. The program needs to be tested. The table, when completed, shows appropriate test data that matches the type of test data and the purpose of the test data.

Complete the table by inserting the missing information.

Test data	Type of test data	Purpose of test data
ABC		....
	Boundary	....
		to make sure that the program accepts data that is an appropriate length

[6]

6 This pseudocode algorithm is intended to allow 1000 positive integers to be input and stored in a one-dimensional (1D) array. The integers are added together as they are input and the highest value is identified. At the end of the algorithm, the highest number, the total and the average of the numbers are output.

```
01 DECLARE Numbers : ARRAY[1:1000] OF INTEGER
02 DECLARE Highest : STRING
03 DECLARE Count : INTEGER
04 DECLARE Total : INTEGER
05 Highest ← 1500
06 Total ← 0
07 FOR Count ← 1 TO 1000
08     INPUT Numbers[Count]
09     Total ← Total + Count
10     IF Numbers[Count] > Total
11       THEN
12         Highest ← Numbers[Count]
13     ENDIF
14 NEXT Count
15 OUTPUT "The highest number is ", Highest
16 OUTPUT "The total is ", Total
17 OUTPUT "The average is ", Average / 1000
```

(a) Identify the line numbers of **five** 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 ....

....

Error 5 line number ....

Correction ....

....
[5]

(b) Write the pseudocode statement to output the average of the numbers, rounded to **two** decimal places.

....

.... [2]

(c) Explain how you could change the corrected algorithm so that it also finds the smallest number that was input and outputs its value at the end.

Any code used must be fully explained.

....

....

....

....

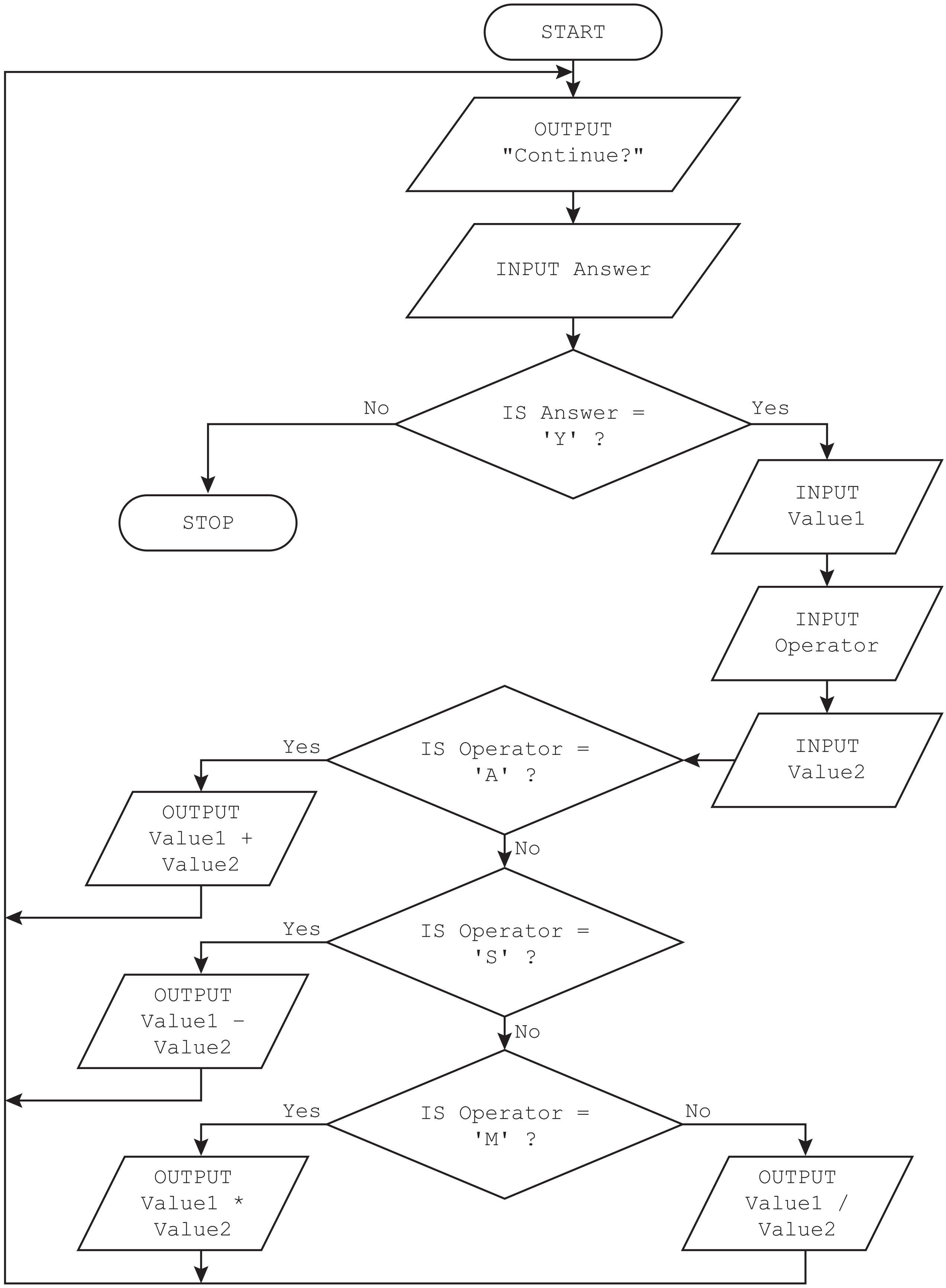
....

....

....

.... [4]

7 This flowchart represents an algorithm.



(a) Complete the trace table for the input data:

Y, 7, S, 9, Y, 5, M, 12, Y, 25, D, 5, N, 10, A, 6

`Answer`	`Value1`	`Operator`	`Value2`	**OUTPUT**

[5]

(b) State the purpose of the algorithm on **page 6**.

.... [1]

(c) There are problems with the given algorithm. If the lower-case letter 'y' is entered for `Answer`, the algorithm stops and if a lower-case letter is entered for `Operator`, the algorithm gives an incorrect output.

(i) Explain how you would change the algorithm to prevent the problems described.

....

....

....

....

....

.... [3]

(ii) Identify **one** more error related to inputs when the algorithm is run.

....

.... [1]

8 A database table, `MajorCity`, stores some details about a number of cities from around the world.

Code	City	Capital	Country	Continent	Population
ASY6	Abu Dhabi	TRUE	United Arab Emirates	Asia	1,566,999
EUY3	Amsterdam	TRUE	Netherlands	Europe	1,174,025
ASY1	Beijing	TRUE	China	Asia	21,766,214
SAY1	Buenos Aires	TRUE	Argentina	South America	15,490,415
AFY1	Cairo	TRUE	Egypt	Africa	22,183,200
EUN1	Frankfurt	FALSE	Germany	Europe	796,437
ASY3	Jakarta	TRUE	Indonesia	Asia	11,248,839
ASN3	Karachi	FALSE	Pakistan	Asia	17,236,230
ASY4	Kuala Lumpur	TRUE	Malaysia	Asia	8,621,724
AFN2	Lagos	FALSE	Nigeria	Africa	15,945,912
SAY2	Lima	TRUE	Peru	South America	11,206,000
EUY1	London	TRUE	UK	Europe	9,648,110
EUY4	Madrid	TRUE	Spain	Europe	6,751,374
ASN2	Mumbai	FALSE	India	Asia	21,296,516
NAN1	New York City	FALSE	USA	North America	7,888,121
EUY2	Paris	TRUE	France	Europe	11,206,000
ASN1	Shanghai	FALSE	China	Asia	29,210,808
ASY5	Singapore	TRUE	Singapore	Asia	6,080,859
AUN1	Sydney	FALSE	Australia	Australia	5,120,894
ASY2	Tokyo	TRUE	Japan	Asia	37,194,104
NAN2	Toronto	FALSE	Canada	North America	6,371,958
EUN2	Valencia	FALSE	Spain	Europe	838,301
SAN1	Valencia	FALSE	Venezuela	South America	1,983,445

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

Fields ....

Records ....

[2]

(b) Identify the field in the database table that is most suitable to be the primary key **and** give **one** reason for your choice.

Primary key field ....

Reason ....

.... [2]

(c) Give the output from the structured query language (SQL) statement:

```
SELECT City, Country, Population
FROM MajorCity
WHERE Continent = "South America"
ORDER BY Population;
```

....

....

....

....

....

.... [3]

(d) Complete the SQL statement to list only the code, city, country and continent of all the cities in the database table that are capital cities.

`SELECT` ....

....

....

.... [4]

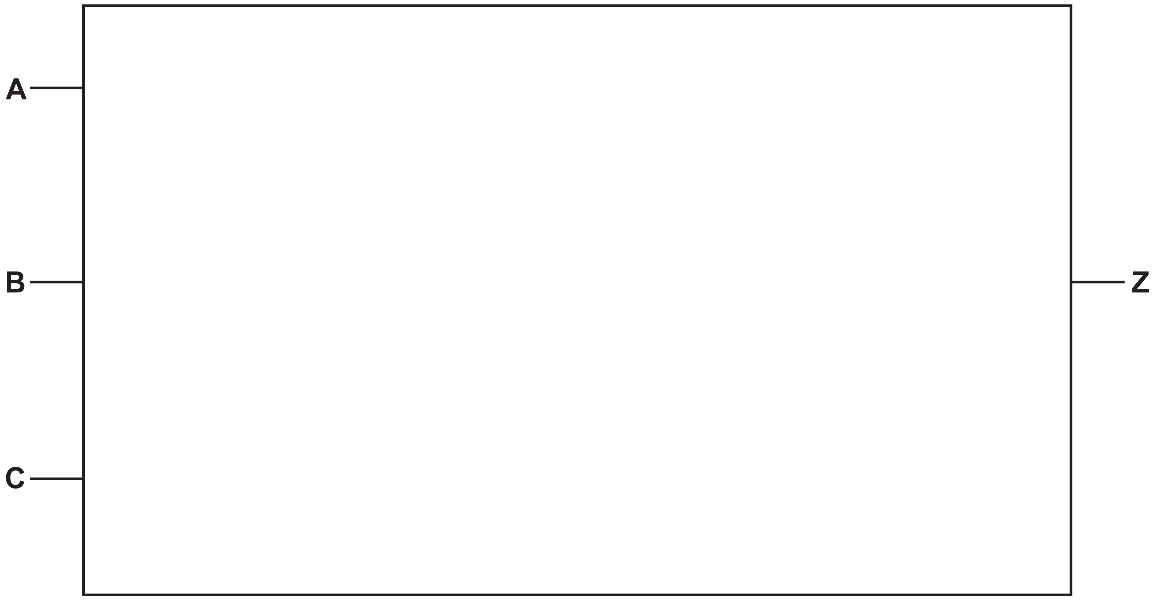
9 Consider this logic expression:

Z = (**A** NAND **B**) XOR (NOT (NOT **B** NAND **C**))

(a) Draw a logic circuit for this logic expression.

Each logic gate must have a maximum of **two** inputs.

Do **not** simplify the logic expression.



[5]

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

A	B	C	Working space	Z
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]

BLANK PAGE

10 A sports club uses a six-character alphanumeric membership code to identify each member of the club.

The one-dimensional (1D) array `MemberID[]` is used to store the unique membership codes for club members.

The two-dimensional (2D) array `Name[]` is used to store the names of the club members. The first and last name of each member will be stored in separate array elements.

The system can store details for a maximum of 1000 members.

The position of any member's data is the same in both arrays. For example, the data in index 2 of `MemberID[]` belongs to the member in index 2 of `Name[]`

The variable `NewID` is used to input a new membership code.

Write a program that meets the following requirements:

- Provide a menu that offers the choices: inputting a new member, outputting a list of membership codes and first and last names, or stopping.
- Input and validate a response to the menu.
- When inputting a new member, input a new membership code and check that it contains six characters:
  - If the new code is six characters, check it against all the previously stored membership codes to make sure it is unique.
  - If the code is **not** unique, a new code must be entered and checked.
  - If the code is unique, it is stored in the first available space in the appropriate array and the new member is required to enter their first name and last name, which are also stored in the corresponding location of the appropriate array.
- When outputting a list of membership codes and names, output for each member: their membership code, first name and last name.
- The program will continue until the stop option on the menu is selected.

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

You do **not** need to declare any arrays, variables or constants; assume this has already been done.

You do **not** need to initialise the data in the arrays.

You do need to initialise any variables or constants used if appropriate.

All inputs and outputs must contain suitable messages.

....

....

....

....

....

....

....

[15]

BLANK PAGE

BLANK PAGE