* 0000800000001 * DFD

CAMBRIDGE
International Education

Cambridge O Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

8587375035

COMPUTER SCIENCE **2210/22**

Paper 2 Algorithms, Programming and Logic **May/June 2026**

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. Do **not** use correction fluid or tape.
- 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.

DC (CJ) 370126
© Cambridge University Press & Assessment 2026

[Turn over

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

Two types of pseudocode statements used for iteration are

A `CASE` and `IF` ☐

B `FOR … NEXT` and `IF` ☐

C `FOR … NEXT` and `WHILE … DO … ENDWHILE` ☐

D `REPEAT … UNTIL` and `CASE` ☐

[1]

2 Tick (✓) **one** box to identify a programming data type used to store a single letter, symbol or number input from the keyboard.

A Boolean ☐

B char ☐

C integer ☐

D real ☐

[1]

3 **(a)** Identify the type of validation check used in each description.

(i) A check to make sure that data has been input and **not** left blank.

..... [1]

(ii) A value placed at the end of a sequence of numbers in the code. The value is calculated from all the other values before it.

..... [1]

(b) Explain the purpose of a format check and give an example of where it could be used.

purpose .....

.....

.....

.....

.....

example .....

.....

[3]

© Cambridge University Press & Assessment 2026

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

(c) Write the pseudocode for an algorithm to perform a validation check on each number as it is input, using the following criteria:

- Input one number at a time.
- If the number is a whole number, store it in the current element of array `Accepted[]`
- If the number is **not** a whole number, ask for it to be input again until it is a whole number.
- When 15 whole numbers have been stored, output a message that states the check has been completed.

Use appropriate input prompts and output messages to make the requirements clear to the user.

You do **not** need to declare any arrays or variables for this algorithm.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [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

4 The program development life cycle is split into several stages.

Outline what happens during the coding and testing stages of the program development life cycle.

coding .....

.....

.....

.....

.....

.....

testing .....

.....

.....

.....

.....

.....

[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

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

Turn over

5 The purpose of this pseudocode algorithm is to find the name of a customer using an account ID, which contains letters and numbers. The name of the customer and the account ID are stored in the prepopulated two-dimensional (2D) array `Accounts[]`. The algorithm would then output the name of the customer.

For example, `Accounts[1, 1]` contains the account ID of the first customer and `Accounts[1, 2]` contains the name of the first customer.

```
01 DECLARE Accounts : ARRAY[1:1000, 1:2] OF STRING
02 DECLARE Row : INTEGER
03 DECLARE AccountID : INTEGER
04 OUTPUT "Please enter an account ID "
05 OUTPUT AccountID
06 FOR Row ← 1 TO 1000
07     CASE OF AccountID = Accounts[Row, 1]
08       THEN
09         OUTPUT "The customer's name is ", Accounts[1, 2]
10     ENDIF
11 NEXT Row
```

(a) Identify the line numbers of **four** errors in the pseudocode and suggest corrections.

error 1 line number .....

correction .....

.....

error 2 line number .....

correction .....

.....

error 3 line number .....

correction .....

.....

error 4 line number .....

correction .....

.....

[4]

(b) State the name given to this type of algorithm.

..... [1]

© Cambridge University Press & Assessment 2026

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

(c) The algorithm stops if the account number is **not** found in the array.

Write the pseudocode statements to change the algorithm to inform the user that the account number has **not** been found in the array.

Clearly state where in the original algorithm your new statements would be placed.

Do **not** rewrite the whole algorithm.

.....

.....

.....

.....

.....

.....

.....

.....

.....

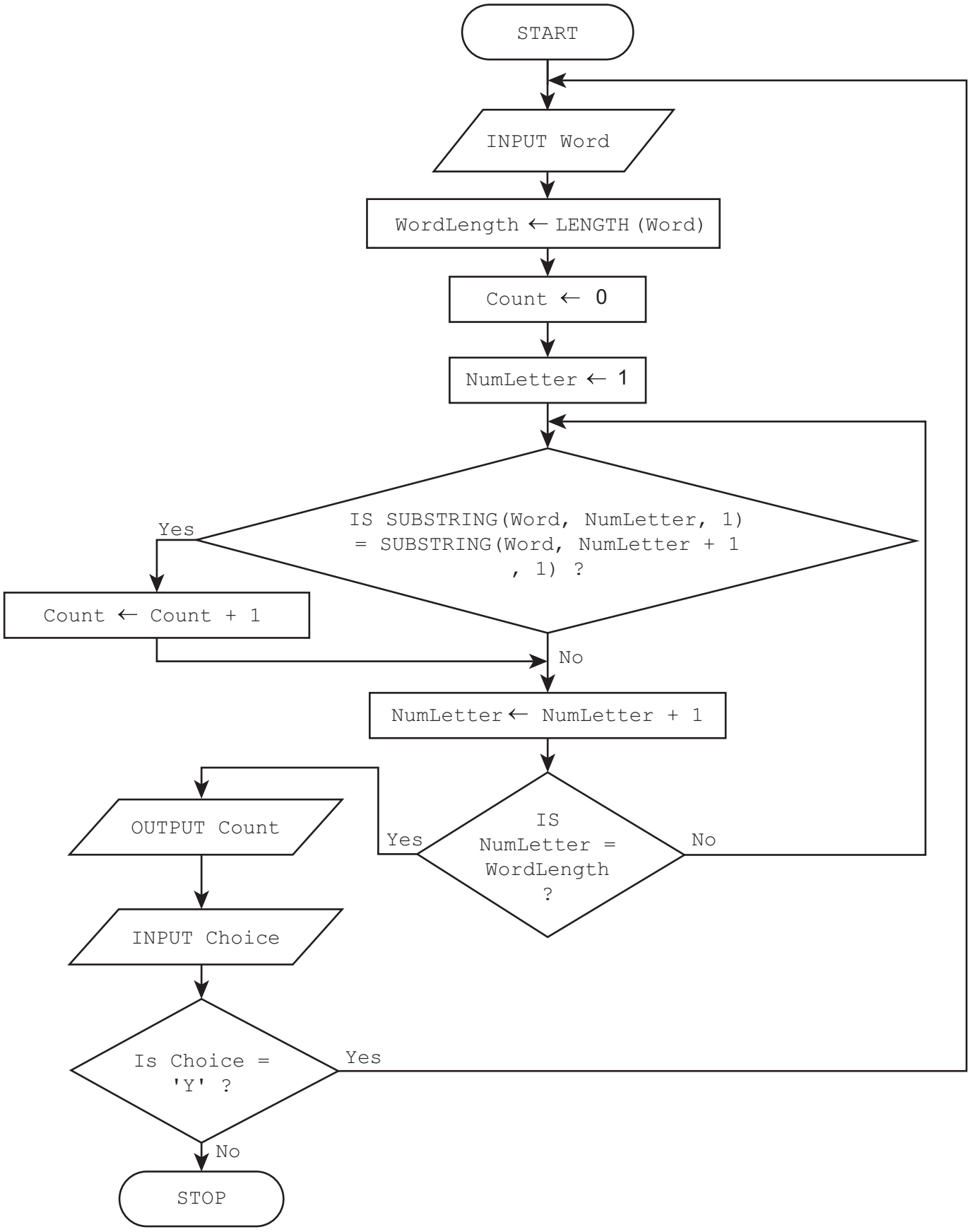
.....

.....

..... [5]

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

6 This flowchart represents an algorithm.



© Cambridge University Press & Assessment 2026

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

(a) Complete the trace table for the algorithm, using the input data:

post, Y, committee, N, bookkeeper

Note: the first character of each word is in position 1.

Word	WordLength	Count	NumLetter	Choice	OUTPUT

[6]

(b) State the purpose of the algorithm shown by the flowchart.

.....

..... [1]

DO NOT WRITE IN THIS MARGIN

7 **(a)** State **one** reason to store data in a file.

.....

..... [1]

(b) A list of names of the members of a club are stored in a file called `Names.txt`

Write the pseudocode statements to load a line of text from `Names.txt` and assign the line of text to an appropriately named variable.

.....

.....

.....

.....

.....

.....

.....

..... [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
DO NOT WRITE IN THIS MARGIN

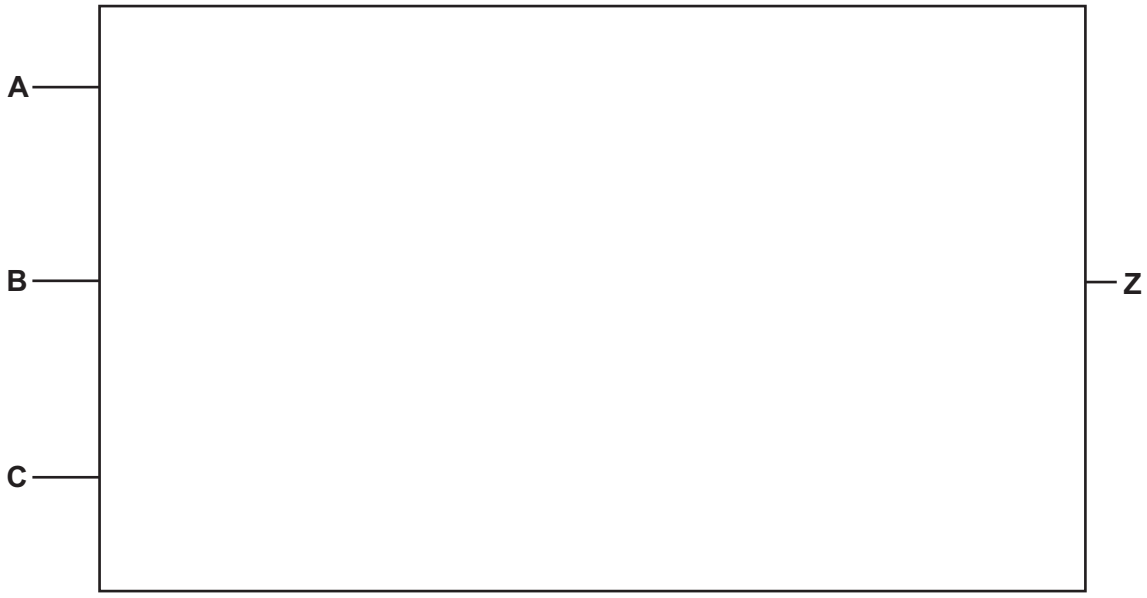
8 Consider the logic expression:

Z = NOT (**B** OR NOT **C**) XOR (**A** 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]

9 **(a)** A database table `CAR_PARTS` stores details of car parts that a business has in stock.

Complete the table by giving the most appropriate data types for each field.

Only the data types: text, character, Boolean, integer, real and date/time are available in this database.

field	data type	description
`PartID`		unique code to identify the car part with letters and numbers
`PartName`		name of the car part, for example windscreen wiper
`CarType`		four-digit number code to represent the type of car
`Price`		selling price to 2 decimal places
`Size`		specifies the size if relevant, otherwise stores 'one size'
`Colour`		single letter code for the colour of the car part
`NumberInStock`		the quantity of car parts in stock
`InStock`		whether or not the car part is in stock

[4]

(b) Complete the structured query language (SQL) statement to list only the part ID, type of car and the quantity in stock of all the car parts that are brake pads.

`SELECT` .....

`FROM` .....

`WHERE` ..... ;

[4]

(c) Describe **one** reason why the `InStock` field is **not** needed in the table.

.....

.....

.....

..... [2]

© Cambridge University Press & Assessment 2026

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

Turn over

10 A program is required to record and analyse the daily rainfall in millimetres for one year. Assume a year has 365 days.

The one-dimensional (1D) array `Rainfall[]` is used to store the daily rainfall measurements.

Write a program for the following requirements:

- Initialise the array `Rainfall[]` to zero.
- Input the rainfall measurement in millimetres for each day and store it in the array.
- Calculate the following statistics after all the inputs have been made for the year:
  - the total rainfall for the year in centimetres (1 centimetre = 10 millimetres), rounded to two decimal places
  - the mean average daily rainfall in millimetres, rounded to four decimal places
  - the total number of days with no rainfall
  - the longest number of consecutive days in which there was no rainfall.
- Output all the calculated statistics with clear and appropriate messaging.
- If the longest number of consecutive days in which there was no rainfall is 15 or more, output a message to indicate there was a drought (a long period when there is little or no rainfall).

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

All arrays, variables or constants used must be declared for this algorithm.

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

.....

.....

.....