Cambridge Assessment International Education
Cambridge International General Certificate of Secondary Education

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

COMPUTER SCIENCE **0478/22**

Paper 2 Problem-solving and Programming **February/March 2019**

1 hour 45 minutes

Candidates answer on the Question Paper.

No Additional Materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your centre number, candidate number and name in the spaces at the top of this page.
Write in dark blue or black pen.
You may use an HB pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.

DO NOT ATTEMPT TASKS 1, 2 AND 3 in the pre-release material; these are for information only.

You are advised to spend no more than **40 minutes** on **Section A** (Question 1).

No marks will be awarded for using brand names of software packages or hardware.

Any businesses described in this paper are entirely fictitious.

At the end of the examination, fasten all your work securely together.

The number of marks is given in brackets [ ] at the end of each question or part question.

The maximum number of marks is 50.

This syllabus is regulated for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **14** printed pages and **2** blank pages.

DC (LK/CB) 162269/1

Cambridge Assessment International Education

[Turn over

Section A

You are advised to spend no longer than 40 minutes answering this section.

Here is a copy of the pre-release material.

DO NOT attempt Tasks 1, 2 and 3 now.

Use the pre-release material and your experience from attempting the tasks before the examination to answer Question 1.

Pre-release material

A pizza ordering service allows customers to design their own pizza. There are three sizes: small, medium and large. A pizza can have a thick or thin base. All pizzas come with tomato and cheese toppings as standard and there are six additional types of topping available:

- Pepperoni
- Chicken
- Extra cheese
- Mushrooms
- Spinach
- Olives

Pizzas always come with tomato and cheese toppings as standard, and can have up to three additional toppings. Customers need to be able to design their own pizza and then confirm or change it. Records are kept showing the number of pizzas sold for each base and size. The number of sales for each additional topping is also recorded.

Write and test a program or programs for the pizza ordering service.

- Your program or programs must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

TASK 1 – Design your pizza.

The customer is given choices of size, base and additional toppings (number and type) as stated above. Only valid choices can be accepted. The customer is asked to confirm their order or alter their choices or not proceed. If the customer confirms their order they are given a unique order number.

TASK 2 – Record the choices.

Extend TASK 1 to record totals for the choices made for ordered pizzas only and calculate the total number of pizzas ordered.

TASK 3 – Find the most and least popular additional pizza toppings.

Using your results from TASK 2, display the most popular and least popular additional toppings as a percentage of the total number of additional toppings ordered.

1 **(a)** All variables, constants and other identifiers should have meaningful names.

State **one** constant and **one** variable that you could have used for **Task 1**. Give the value that would be assigned to the constant. Give the data type for the variable. Explain what each one could be used for.

Constant name .....

Value .....

Use .....

.....

Variable name .....

Data type .....

Use .....

.....

[6]

(b) Explain how you would need to change your program for **Task 1** if there were three bases to choose from (thick, thin and extra crispy).

.....

.....

.....

..... [2]

(c) (i) Write an algorithm for choosing the additional toppings in **Task 1**, using **either** pseudocode, programming statements **or** a flowchart. Your algorithm must only include this part of **Task 1**.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

[5]

(ii) Explain how your algorithm in **part (c)(i)** only allowed valid choices for the additional pizza toppings.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [3]

(d) Explain how your program completed **Task 3**. Any programming statements used in your answer must be fully explained.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

Section B

2 **(a)** An algorithm has been written in pseudocode to input 50 numbers and total only the positive numbers.

```
Count ← 1
Total ← Count
REPEAT
  INPUT Number
  IF Number <> 0
  THEN
    Total ← Total + Count
  ENDIF
  Count ← Count + 1
UNTIL Count < 50
PRINT Total
```

Find the **four** errors in the pseudocode and suggest a correction for each error.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....

[4]

(b) Show how you would change the corrected algorithm to only total numbers greater than 0 and less than 20.

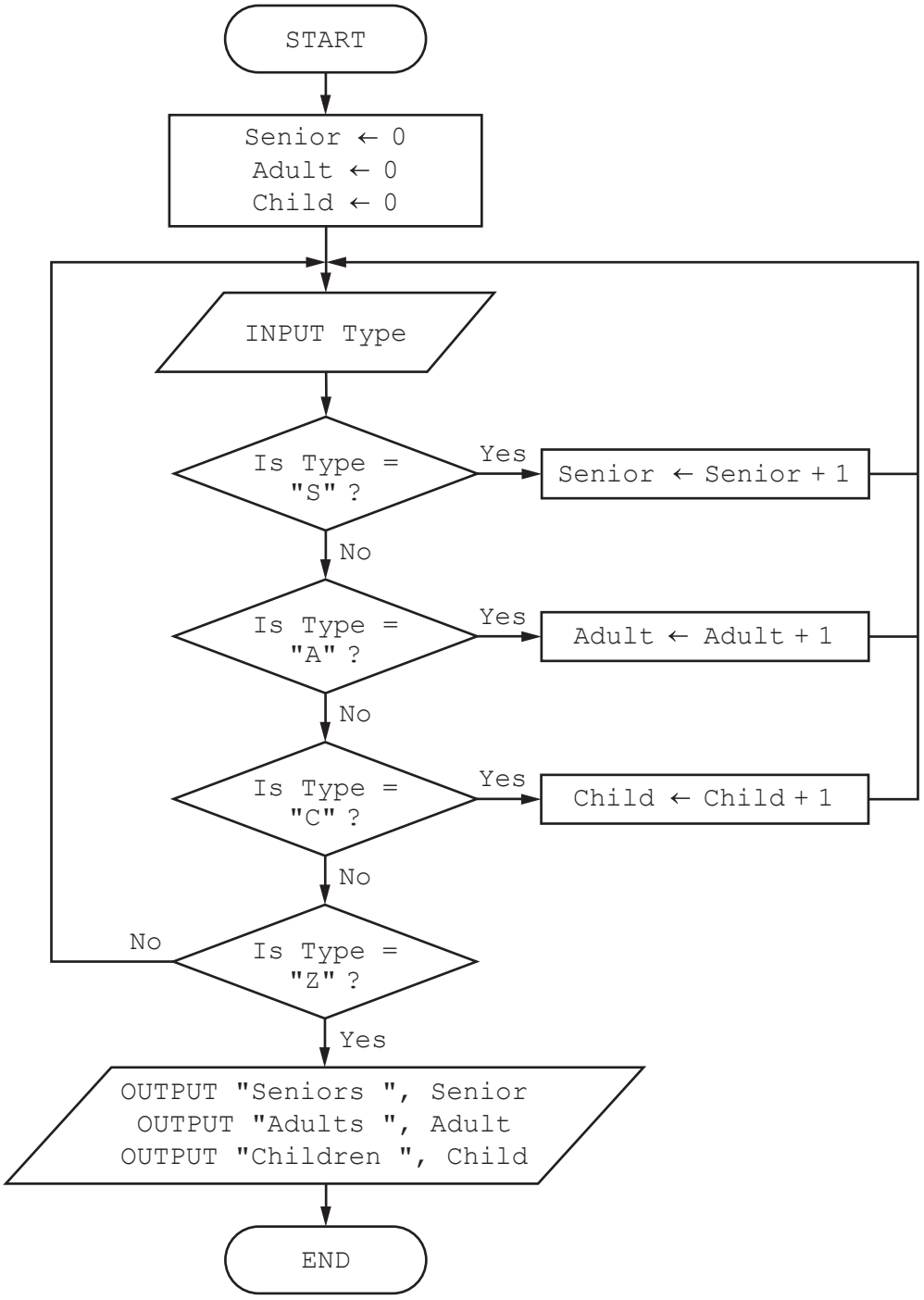
.....

.....

..... [2]

Question 3 starts on page 10.

3 This flowchart inputs the type of passenger for a survey: S for Senior, A for Adult, C for Child. All other values are ignored, apart from Z which ends the process.



Complete the trace table for the passenger input data:
S, S, S, A, C, C, C, A, A, A, A, W, S, S, D, C, Z, D, S

Senior	Adult	Child	Type	OUTPUT

[5]

4 For each of the **four** groups of statements in the table, place a tick in the correct column to show whether it is an example of **Selection** or **Repetition**.

Statements	Selection	Repetition

`FOR X ← 1 TO 10`
`  SUM ← SUM + 1`
`NEXT X`		

`WHILE X > 10 DO`
`  SUM ← SUM + 1`
`  X ← X - 1`
`ENDWHILE`		

`IF X > 10`
`THEN`
`  SUM ← SUM + 1`
`  X ← X - 1`
`ENDIF`		

`REPEAT`
`  SUM ← SUM + 1`
`  X ← X - 1`
`UNTIL X > 10`		

[4]

5 A programmer restricts input values to less than 90 and greater than 60.

(a) State whether this is called validation or verification.

.....

Name the check that needs to be used.

.....

[2]

(b) State **three** different types of test data the programmer would need to use. Give an example of each type and the reason that the programmer chose that test data.

Type 1 .....

Example .....

Reason .....

.....

Type 2 .....

Example .....

Reason .....

.....

Type 3 .....

Example .....

Reason .....

.....

[9]

6 A database table, BIKETYRES, is used to keep a record of tyres for sale in a cycle shop. Tyres are categorised by width and diameter in millimetres, whether they have an inner tube and the type of terrain for which they are designed.

Tyre Code	Width	Diameter	Tube	Terrain	Stock Level
SLTT	23	700	YES	Asphalt	18
MLNT	24	700	NO	Asphalt	23
LLNT	28	700	NO	Asphalt	19
SLTM	23	700	YES	Mixed	22
MLTM	24	700	YES	Mixed	14
LLTM	28	700	YES	Mixed	12
SLTH	23	700	YES	Hard	10
MLTH	24	700	YES	Hard	5
LLNH	28	700	NO	Hard	7
SLNM	23	700	NO	Mixed	12
MLNM	24	700	NO	Mixed	22
LLNM	28	700	NO	Mixed	18
SSNT	23	650	NO	Asphalt	10
MSNT	24	650	NO	Asphalt	8
SSTM	23	650	YES	Mixed	5
MSNM	24	650	NO	Mixed	4

The query-by-example grid below displays the tyre code and the stock level of all 28 mm width tyres suitable for mixed terrain.

Field:	Tyre Code	Stock Level	Width	Terrain
Table:	BIKETYRES	BIKETYRES	BIKETYRES	BIKETYRES
Sort:				
Show:	☑	☑	☐	☐
Criteria:			= 28	= 'Mixed'
or:				

Alter the query to show the tyre code and stock level in ascending order of stock level for all 24 mm asphalt terrain tyres. Write the new query in the following query-by-example grid.

Field:				
Table:				
Sort:				
Show:	☐	☐	☐	☐
Criteria:				
or:				

[4]

BLANK PAGE

BLANK PAGE