Cambridge Assessment International Education

# Cambridge IGCSE™

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| CANDIDATE NAME | | | | | | | | | |
| CENTRE NUMBER | | | | | | CANDIDATE NUMBER | | | |

*4657181087*

**COMPUTER SCIENCE** **0478/23**

Paper 2 Problem-solving and Programming **October/November 2021**

**1 hour 45 minutes**

You must answer on the question paper.

No additional materials are needed.

**INSTRUCTIONS**

- Answer all questions.
- **Do not attempt Tasks 1, 2 and 3** in the copy of the pre-release material on page 2; these are for information only.
- 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 50.
- 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 2021

## 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 following tasks before the examination to answer Question 1.

**Pre-release material**

A cruise ship has a speciality restaurant where tables can be booked for any of the three sessions: lunch, early dinner or late dinner. A booking for a table can only be made on the day. There are twenty tables available to book in the restaurant. Today's date and how many tables are available for lunch, early dinner and late dinner are displayed on a screen at the entrance to the restaurant.

Only one table can be booked at a time. When a booking is made, the name of the passenger making the booking is recorded, together with their cabin number and any special dietary requirements.

Write and test a program or programs for a computer system to manage the daily restaurant bookings.

- Your program or programs must include appropriate prompts for the entry of data; data must be validated on entry.
- 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** – at the start of the day

Write a program to set up the screen display at the start of the day showing the date and how many tables are available for lunch, early dinner and late dinner. Bookings are to be stored in three separate arrays: lunch, early dinner and late dinner. Initialise further arrays to record for each table booked: the name of the passenger making the booking, their cabin number and any special dietary requirements.

**Task 2** – making a table booking at the restaurant

Check if there is a table available for the session requested. If a table is available, record the passenger's name, cabin number and any special dietary requirements. Mark the table as booked for that session. Display the name and cabin number for the passenger to check. Update the screen display and mark a session as fully booked if all the tables are now booked.

**Task 3** – special dietary requirements

The recording of special dietary requirements is confusing the restaurant staff. It has been decided to use fixed options instead of a description. Only one option can be chosen for each booking.

The options are:

- gluten-free
- vegetarian
- vegan
- diabetic
- none.

Update **Task 2** to allow for this.

Update **Task 2** to count and display how many tables have vegetarian or vegan diners during the day.

© UCLES 2021

**1** All variables, constants and other identifiers must have meaningful names.

**(a)** Identify **one** constant that you could have used for **Task 1**. Give the value that would be assigned to this constant. State the use of this constant.

Constant .....

Value .....

Use .....

.....

[3]

**(b)** Describe the arrays that you have set up in **Task 1** to record today's data about the restaurant tables.

.....

.....

.....

.....

.....

.....

.....

..... [4]

**(c)** Sometimes the restaurant has fewer tables available for a session.

Explain how you would change your **Task 1** program to allow input of the number of tables available for a session.

.....

.....

.....

.....

.....

..... [3]

**(d)** Part of **Task 2** is to check if a table is available. Write an algorithm to do this, using either pseudocode, programming statements or a flowchart. Assume that **Task 1** has been completed.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

© UCLES 2021

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [6]

**(e)** Explain how your program for **Task 3** updated **Task 2**.

Include any programming statements used and fully explain the purpose of each statement.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

**Section B starts on Page 8.**

## Section B

**2** An algorithm has been written in pseudocode to generate 50 positive random integers with values less than or equal to 100. These numbers are stored in the array `NumRand[]`

The function `RandUp(X,Y)` generates a random integer greater than `X` and less than or equal to `Y` For example, `RandUp(1,4)` generates `2` or `3` or `4`

```
1 Count ← 0
2 WHILE Counter > 50 DO
3     NumRand[Counter] ← RandUp(1,100)
4     Counter ← Counter – 2
5 ENDWHILE
```

**(a)** Find the **four** errors in the pseudocode and write a correction for each error.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....

.....

[4]

**(b)** The pseudocode for this algorithm could be shortened by the use of a `FOR … NEXT` loop.

Rewrite the algorithm using a `FOR … NEXT` loop.

.....

.....

.....

..... [3]

**3** A program has been written to check the length and content of a password. The password must be eight or more characters long and contain at least one special character, for example, `Secret!*!`

**(a) (i)** State suitable examples of normal and erroneous test data that could be used to test this program. For each example give the reason for your choice of test data.

Normal test data example .....

Reason .....

.....

Erroneous test data example .....

Reason .....

.....

[4]

**(ii)** Explain why two pieces of boundary test data are required for this program.
Give an example of each piece of boundary test data.

.....

.....

.....

.....

..... [3]

**(b)** Describe **two** methods of verification that could be used to verify this data as it is input.

Method 1 .....

.....

.....

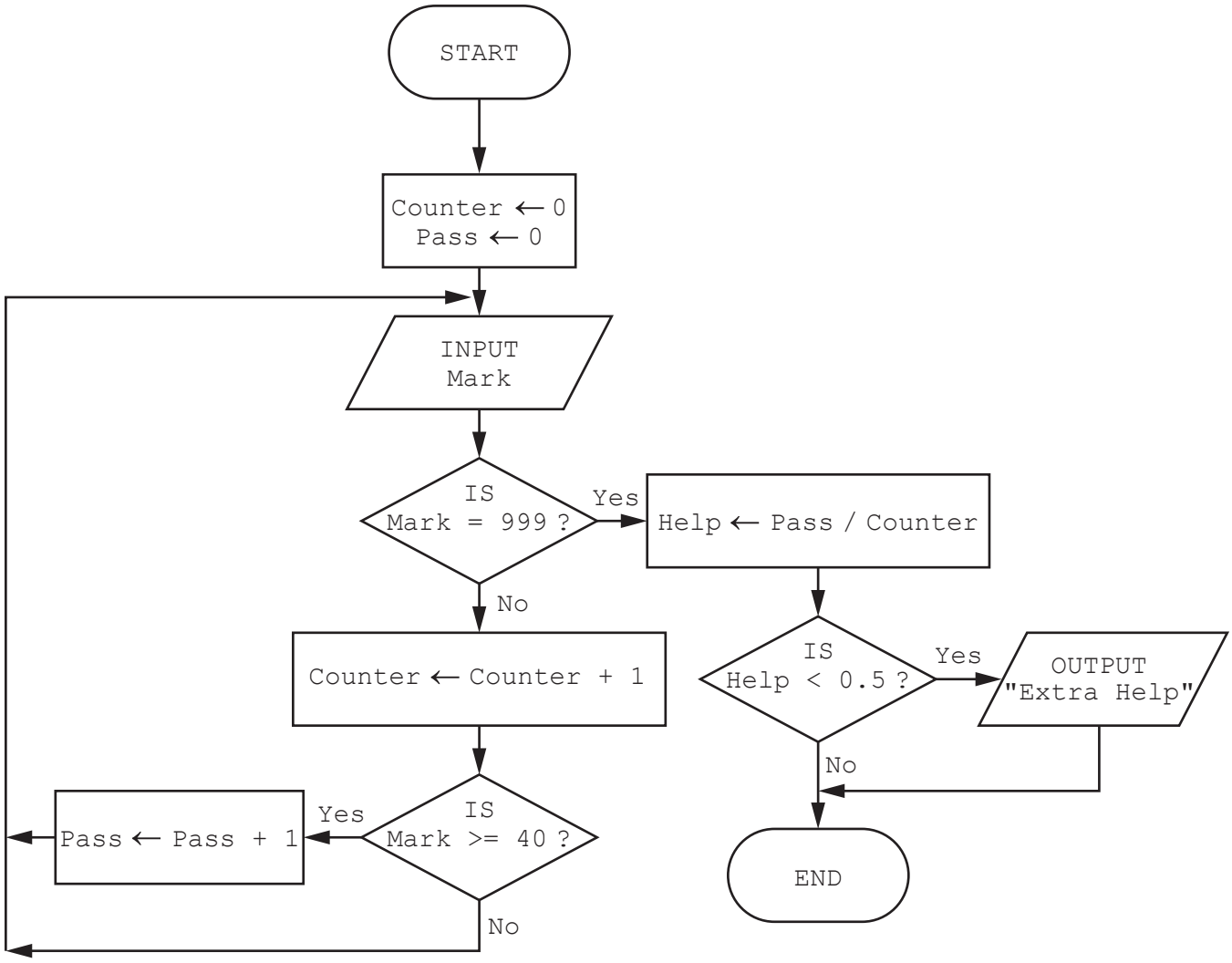
Method 2 .....

.....

.....

[4]

**4** The algorithm, shown by this flowchart, allows the input of examination marks for a class of students. A mark of 999 ends the process. If a mark is 40 or over then a pass grade is awarded. The number of pass grades is calculated for the whole class. If this is under 50% of the class, the class is offered extra help.



Complete a trace table for the algorithm using this input data:
88, 24, 60, 30, 44, 17, 25, 22, 54, 6, 999, −1

| Counter | Pass | Mark | Help | OUTPUT |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

[5]

**5** A database table, COMPUTER, is used to keep a record of computers available for sale.

The following data is stored for each computer:

- CATEGORY – desktop, laptop or tablet
- WEIGHT – weight in kilograms
- MANUFACTURER – ICN, Linoldo, Pear or JoeSing
- PRICE – price in $
- CODE – a unique code allocated by the manufacturer, e.g. P771
- STOCK – quantity in stock.

A database management system uses these data types:

Text Number Currency Boolean

The CATEGORY field and MANUFACTURER field have a data type of text.

**(a)** Select the most appropriate data type for each field from the **four** types shown. State the reason why you chose the data type.

WEIGHT data type .....

Reason .....

.....

PRICE data type .....

Reason .....

.....

CODE data type .....

Reason .....

.....

STOCK data type .....

Reason .....

.....

[4]

**(b)** Complete the query-by-example grid to display only the category, manufacturer, price and code of the computers with weight of less than 2.5 kilograms.

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

[3]

**BLANK PAGE**

© UCLES 2021

**BLANK PAGE**

**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.

© UCLES 2021