Cambridge IGCSE

**Cambridge International Examinations**
Cambridge International General Certificate of Secondary Education

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

*1214679873*

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

Paper 2 Problem-solving and Programming **October/November 2018**

**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 approved for use in England, Wales and Northern Ireland as a Cambridge International Level 1/Level 2 Certificate.

This document consists of **11** printed pages and **1** blank page.


© UCLES 2018

## 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 holiday park has a number of log cabins that it rents by the week as shown in the table:

| Name | Capacity | Peak | Off-peak |
|---|---|---|---|
| Hetty | 4 | $400.00 | $250.00 |
| Poppy | 4 | $400.00 | $250.00 |
| Blue Skies | 4 | $500.00 | $350.00 |
| Bay View | 6 | $650.00 | $500.00 |
| Happy Days | 6 | $695.00 | $550.00 |
| Summer Joy | 6 | $800.00 | $600.00 |
| Walkers' Rest | 8 | $950.00 | $750.00 |
| Bertie | 8 | $1050.00 | $850.00 |
| Green Forest Lodge | 10 | $1200.00 | $950.00 |
| Coppice Lodge | 10 | $1500.00 | $1150.00 |

The capacity represents the maximum number of occupants for each log cabin. A program is needed to record and store bookings. Log cabins can only be booked from weeks labelled on the calendar as weeks 23 to 39, inclusive. Peak rates operate for weeks 27 to 35, inclusive, and off-peak rates apply for weeks 23 to 26 and weeks 36 to 39, inclusive.

Write and test a program or programs.
- 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 arrays, 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 – Setting up the bookings system.
Write a program, using arrays, to identify each log cabin, its capacity, cost and whether or not it has been booked for each week. Identify each week by a number ranging from week 23 to 39.

TASK 2 – Taking a booking.
Extend the program to:
- Identify and display which weeks are available for each log cabin, and its capacity.
- Input the log cabin, number of weeks and start week for the booking.
- Generate a unique booking code for the week(s) and log cabin chosen.
- Store the unique booking code in your array
  (multiple week bookings will need the booking code stored multiple times).
- Calculate and output the cost of the booking.

TASK 3 – Applying a special offer.
Amend the program to apply a 10% discount to any booking of three weeks or more. Output the original cost and the discounted cost of the booking.

© UCLES 2018

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

**(i)** State the name, data type and use of **three** arrays you have created for **Task 1**.

Array 1 name .....

Data type .....

Use .....

.....

Array 2 name .....

Data type .....

Use .....

.....

Array 3 name .....

Data type .....

Use .....

.....

[6]

**(ii)** State the name, data type and use for **one** variable you used in **Task 2**.

Variable name .....

Data type .....

Use .....

.....

[2]

© UCLES 2018



**[Turn over**

**(b)** Write an algorithm to show how your program finds and displays the weeks available for each log cabin and its capacity (part of **Task 2**), using **either** pseudocode, programming statements or a flowchart.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

© UCLES 2018

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................

.......................................................................................................................................................[4]

© UCLES 2018

**(c)** Describe how you could validate the input to identify a log cabin in **Task 2**. State **one** valid and **one** invalid item of data to test your validation method.

Validation method .....

.....

.....

.....

.....

.....

Valid test data .....

Invalid test data .....

[4]

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

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[4]

© UCLES 2018

## Section B

**2** Describe, giving an example for each, the following data types used in programming.

**Integer**

Description .....

.....

Example .....

**String**

Description .....

.....

Example .....

[4]

**3** Give an example of a pseudocode statement or statements to perform each of the following functions.

A condition controlled loop .....

.....

.....

A conditional statement .....

.....

.....

Totalling .....

.....

.....

[3]

**4** This is a section of program code.

```
1 Total = 100.00
2 PRINT 'Enter the height of each member of your class, one at a
  time, when prompted'
3 FOR Count = 1 TO 30
4   PRINT 'Enter a height in metres'
5   INPUT Height
6   Total = Total + Height
7   PRINT Total / 30
8   Count = Count + 1
9 NEXT Count
```

**(a)** There are **three** errors in this code.

State the line numbers that contain the errors and describe how to correct each error.

Error 1 .....

.....

.....

Error 2 .....

.....

.....

Error 3 .....

.....

.....

[3]

**(b)** State the purpose of this program.

.....

.....

.....[1]

© UCLES 2018

**5** The algorithm allows a number to be entered. It then calculates and outputs the next number in the mathematical series.

```
Fib ← 1
Prev2 ← 0
Prev1 ← 1
INPUT Number
IF Number = 0
    THEN Fib ← 0
ENDIF
WHILE Number > 2
    Fib ← Prev2 + Prev1
    Prev2 ← Prev1
    Prev1 ← Fib
    Number ← Number - 1
ENDWHILE
OUTPUT Fib
```

**(a)** Complete the trace table for the input data: **7**

| Fib | Prev2 | Prev1 | Number | OUTPUT |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

[4]

**(b)** Complete the trace table for the input data: **2**

| Fib | Prev2 | Prev1 | Number | OUTPUT |
|---|---|---|---|---|
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

[2]

**6** An online fruit tree specialist sells fruit trees in various sizes. A database table, TREETAB, shows the tree type and, for each size, the price and whether they are in stock.

| Tree Type | Size1 | Size1 In | Size2 | Size2 In | Size3 | Size3 In |
|---|---|---|---|---|---|---|
| Apple | 10.95 | Yes | 14.95 | Yes | 29.95 | Yes |
| Apple | 12.95 | Yes | 14.95 | Yes | 29.95 | Yes |
| Cherry | 24.95 | No | 34.95 | No | 59.95 | Yes |
| Fig | 19.95 | Yes | 29.95 | No | 49.95 | Yes |
| Guava | 19.95 | No | 29.95 | No | 59.95 | No |
| Nectarine | 8.50 | Yes | 11.95 | Yes | 19.95 | Yes |
| Olive | 19.95 | No | 39.95 | Yes | 59.95 | Yes |
| Peach | 9.25 | No | 11.95 | Yes | 19.95 | Yes |
| Pear | 10.95 | Yes | 14.95 | Yes | 29.95 | Yes |
| Plum | 8.95 | Yes | 11.95 | Yes | 19.95 | Yes |
| Pomegranate | 12.95 | No | 18.95 | Yes | 34.95 | No |
| Quince | 34.95 | Yes | 44.95 | Yes | 84.95 | No |

**(a)** State whether any of the fields shown would be suitable as a primary key.

.....

Explain your answer .....

.....

.....

[2]

**(b)** Complete the table to show the most appropriate data type for each of the fields based on the data shown in the table at the start of question 6.

| Field | Data type |
|---|---|
| Tree Type | |
| Size3 | |
| Size2 In | |

[3]

© UCLES 2018

**(c)** Show the output that would be given by this query-by-example.

| Field: | Tree Type | Size1 | Size1 In | | |
|---|---|---|---|---|---|
| Table: | TREETAB | TREETAB | TREETAB | | |
| Sort: | | Descending | | | |
| Show: | ☑ | ☑ | ☑ | ☐ | ☐ |
| Criteria: | | <10.00 | | | |
| or: | | | | | |

.....

.....

.....

.....[4]

**(d)** Using the following query-by-example grid, write a query to identify all types of the fruit trees that are out of stock for all three sizes. Make sure the type of the tree and the various 'in stock' fields are shown. The trees should be listed in alphabetical order by type.

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

[4]

© UCLES 2018

**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 International Examinations Copyright Acknowledgements Booklet. This is produced for each series of examinations and is freely available to download at www.cie.org.uk after the live examination series.

Cambridge International Examinations is part of the Cambridge Assessment Group. Cambridge Assessment is the brand name of University of Cambridge Local Examinations Syndicate (UCLES), which is itself a department of the University of Cambridge.

© UCLES 2018