Cambridge
O Level

Cambridge International Examinations
Cambridge Ordinary Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

COMPUTER SCIENCE **2210/22**

Paper 2 Problem-solving and Programming **May/June 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.

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

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 farmer records the milk production of a herd of cows. Every cow has a unique 3-digit identity code. Each cow can be milked twice a day, seven days a week. The volume of milk from each cow is recorded in litres correct to one decimal place (yield) every time the cow is milked. The size of the herd is fixed. At the end of the week the total and the average yield for each cow for that week is calculated.

The farmer identifies the cow that has produced the most milk that week. The farmer also identifies any cows that have produced less than 12 litres of milk on four or more days that week.

A program is required to record the yield for each cow every time it is milked, calculate the total weekly volume of milk for the herd and the average yield per cow in a week. The program must also identify the cow with the best yield that week and identify any cows with a yield of less than 12 litres of milk for four or more days that week.

Write and test a program or programs for the farmer.

- 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 – Record the yield.

Write a program for TASK 1 to record the milk yields for a week. The program records and stores the identity code number and the yield every time a cow is milked.

TASK 2 – Calculate the statistics.

Using your recorded data from TASK 1, calculate and display the total weekly volume of milk for the herd to the nearest whole litre. Calculate and display the average yield per cow in a week to the nearest whole litre.

TASK 3 – Identify the most productive cow and cows that are producing a low volume of milk.

Extend TASK 2 to identify and display the identity code number and weekly yield of the cow that has produced the most milk. Also identify and display the identity code numbers of any cows with a yield of less than 12 litres of milk for four days or more in the week.

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

(i) State the name, the data type and the use of **two** variables that you have used in **Task 2**.

Variable 1 name .....

Data type .....

Use .....

Variable 2 name .....

Data type .....

Use .....
[2]

(ii) Describe, with the aid of some sample data, the data structures that you have used to record the data for the cows in **Task 1**.

.....

.....

.....

.....

.....

.....

.....

.....[4]

(b) Explain how your program for **Task 1** ensures that each 3-digit identity code is unique.

.....

.....

.....

.....[2]

[Turn over

(c) Write an algorithm for **Task 2**, using **either** pseudocode, programming statements **or** a flowchart**.** Assume that **Task 1** has been completed.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[5]

(d) (i) Explain how your program for **Task 3** finds the cows with a daily yield of less than 12 litres of milk for four days or more in the week. Any programming statements used in your answer must be fully explained.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[5]

(ii) Explain how you would extend your program for **Task 3** to **store** the identity code number(s) of those cows with a yield of less than 12 litres of milk for four days or more in the week.

.....

.....

.....

.....

.....

.....[2]

Section B

2 **(a)** Draw a flowchart for an algorithm to input numbers. Reject any numbers that are negative and count how many numbers are positive. When the number zero is input, the process ends and the count of positive numbers is output.

[6]

(b) Explain the changes you will make to your algorithm to also count the negative numbers.

.....

.....

.....

.....[2]

Question 3 starts on Page 8.

3 This pseudocode algorithm inputs two non-zero numbers and a sign, and then performs the calculation shown by the sign. An input of zero for the first number terminates the process.

```
INPUT Number1, Number2, Sign
WHILE Number1 <> 0
   IF Sign = '+' THEN Answer ← Number1 + Number2 ENDIF
   IF Sign = '-' THEN Answer ← Number1 - Number2 ENDIF
   IF Sign = '*' THEN Answer ← Number1 * Number2 ENDIF
   IF Sign = '/' THEN Answer ← Number1 / Number2 ENDIF
   IF Sign <> '/' AND Sign <> '*' AND Sign <> '-' AND Sign <> '+'
      THEN Answer ← 0
   ENDIF
   IF Answer <> 0 THEN OUTPUT Answer ENDIF
   INPUT Number1, Number2, Sign
ENDWHILE
```

(a) Complete the trace table for the input data:
5, 7, +, 6, 2, –, 4, 3, *, 7, 8, ?, 0, 0, /

Number1	Number2	Sign	Answer	OUTPUT

[3]

(b) Show how you could improve the algorithm written in pseudocode by writing an alternative type of conditional statement in pseudocode.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[3]

4 A programmer has written a routine to store the name, email address and password of a contributor to a website's discussion group.

(a) The programmer has chosen to verify the name, email address and password.

Explain why verification was chosen and describe how the programmer would verify this data.

.....

.....

.....

.....

.....

.....

.....

.....

.....[4]

(b) The programmer has also decided to validate the email address and the password.

Describe validation checks that could be used.

Email address.....

.....

.....

Password .....

.....

.....

[2]

5 A program checks that the weight of a basket of fruit is over 1.00 kilograms and under 1.10 kilograms. Weights are recorded to an accuracy of two decimal places and any weight not in this form has already been rejected.

Give **three** weights as test data and for each weight state a reason for choosing it. All your reasons must be different.

Weight 1 .....

Reason .....

.....

Weight 2 .....

Reason .....

.....

Weight 3 .....

Reason .....

.....

[3]

6 A database table, TREES, is used to keep a record of the trees in a park. Each tree is given a unique number and is examined to see if it is at risk of dying. There are over 900 trees; part of the database table is shown.

Tree Number	Type	Map Position	Age in Years	At Risk
TN091	Acacia	A7	250	Y
TN172	Olive	C5	110	N
TN913	Cedar	B9	8	N
TN824	Banyan	A3	50	Y
TN021	Pine	D5	560	Y
TN532	Teak	C8	76	Y
TN043	Yew	B1	340	N
TN354	Spruce	D4	65	N
TN731	Elm	B10	22	Y
TN869	Oak	C9	13	N
TN954	Pine	E11	3	N

(a) State the number of fields in the table.

.....[1]

(b) The tree numbering system uses TN followed by three digits. The numbering system will not work if there are over 1000 trees.
Describe, with the aid of an example, how you could change the tree numbering system to allow for over 1000 trees. Existing tree numbers must not be changed.

.....

.....

.....

.....[2]

(c) Using the query-by-example grid, write a query to identify at risk trees over 100 years old. Display only the type and the position on the map.

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

[4]

BLANK PAGE