Cambridge Assessment International Education

Cambridge IGCSE™ (9–1)

CANDIDATE NAME									
CENTRE NUMBER						CANDIDATE NUMBER				

COMPUTER SCIENCE **0984/02**

Paper 2 Problem-solving and Programming **For examination from 2020**

SPECIMEN PAPER

1 hour 45 minutes

Candidates answer on the question paper.

No additional materials are required.

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.
Calculators must not be used in this paper.

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 **13** printed pages and **1** blank page.

[Turn over

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 Section A Question 1.

Pre-release material

A teacher needs a program to record marks for a class of 30 students who have sat three computer science tests.

Write and test a program for the teacher.

- Your program 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 – Set up arrays

Set-up one dimensional arrays to store:

- Student names
- Student marks for Test 1, Test 2 and Test 3
  - Test 1 is out of 20 marks
  - Test 2 is out of 25 marks
  - Test 3 is out of 35 marks
- Total score for each student

Input and store the names for 30 students. You may assume that the students' names are unique.

Input and store the students' marks for Test 1, Test 2 and Test 3. All the marks must be validated on entry and any invalid marks rejected.

TASK 2 – Calculate

Calculate the total score for each student and store in the array.
Calculate the average total score for the whole class.

Output each student's name followed by their total score.
Output the average total score for the class.

TASK 3 – Select

Select the student with the highest total score and output their name and total score.

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

(i) Declare the array to store the students' names.

..... [1]

(ii) Declare the arrays to store each student's marks and total score.

.....

..... [2]

(b) **(i)** Show the design of your algorithm to complete **Task 1** and **Task 2** using pseudocode, programming statements or a flowchart. Do **not** include any of the validation checks or input prompts in your algorithm.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [8]

(ii) Comment on the efficiency of your design.

.....

.....

..... [1]

(c) Show **two** different sets of student data that you could use to check the validation used in **Task 1**. Explain why you chose each data set.

Set 1 .....

Reason for choice .....

.....

.....

Set 2 .....

Reason for choice .....

.....

..... [2]

(d) (i) Explain how you select the student with the highest score (**Task 3**). You may include pseudocode or programming statements to help illustrate your explanation.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

(ii) How does your program work when there is more than one student having the highest score? Explain using your method given in **part (d)(i)**.

.....

.....

..... [1]

Section B

2 Jatinder uses Internet banking.
This pseudocode checks her PIN.

```
c ← 0
INPUT PIN
x ← PIN
REPEAT
   x ← x/10
   c ← c + 1
UNTIL x < 1
IF c <> 5
   THEN
      PRINT "error in PIN entered"
   ELSE
      PRINT "PIN OK"
ENDIF
```

(a) What value of c and what message would be output if the following PINs were entered?

5 1 0 2 0 Value of c: .....

Message: .....

5 1 2 0 Value of c: .....

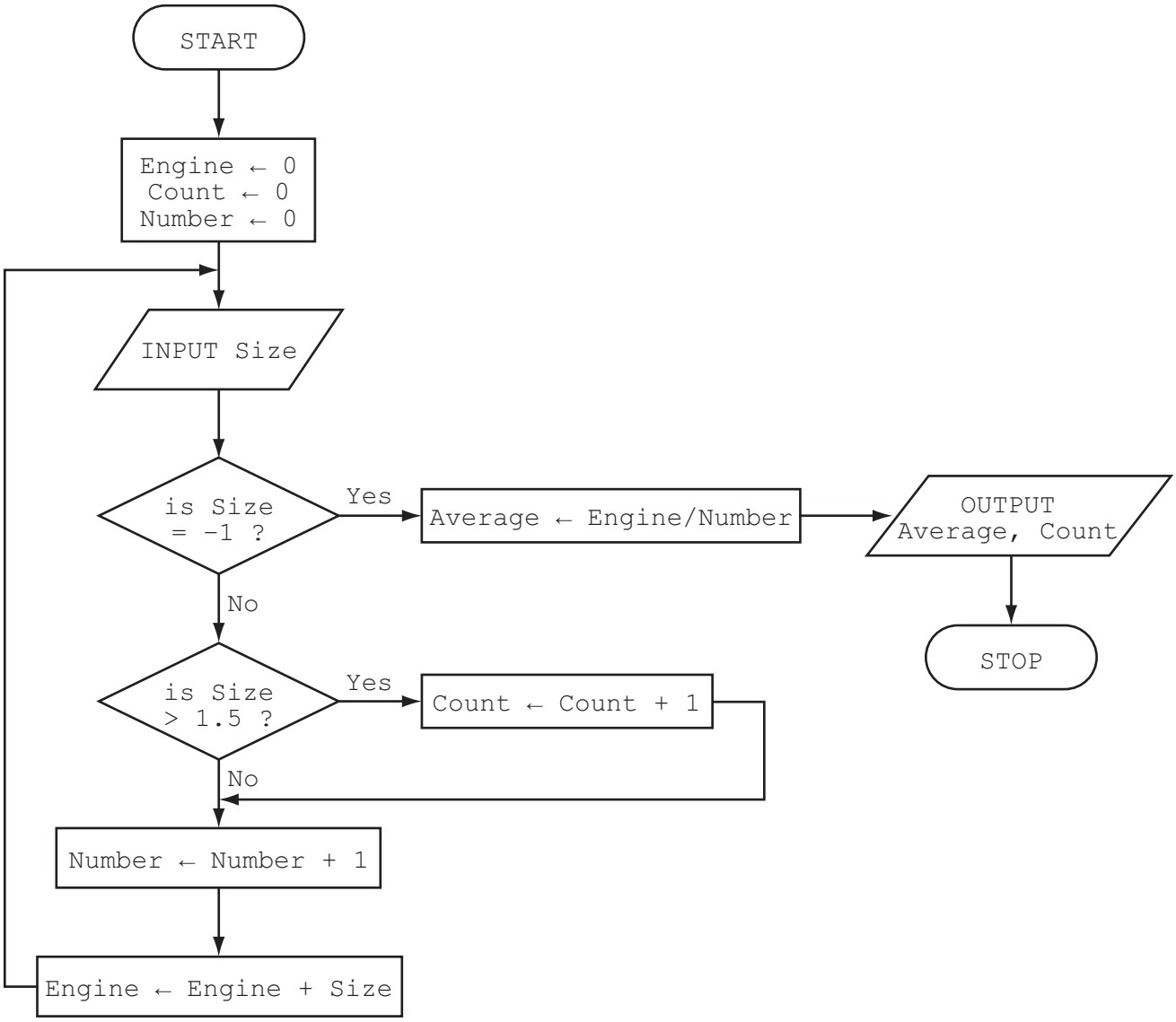
Message: ..... [2]

(b) What type of validation check is being carried out here?

..... [1]

3 The flowchart inputs the size of a number of car engines; a value of –1 stops the input.

This information is output: *average engine size* and *number of engines with size > 1.5*



Complete the trace table for the input data.

1.8, 2.0, 1.0, 1.3, 1.0, 2.5, 2.0, 1.3, 1.8, 1.3, –1

Engine	Count	Number	Size	Average	OUTPUT

[6]

4 Read this section of program code that inputs twenty (20) numbers and then outputs the largest number input.

```
1  h = 0
2  c = 0
3  REPEAT
4     READ x
5     IF x > h THEN x = h
6     c = c + 1
7     PRINT h
8  UNTIL c < 20
```

There are **three** errors in this code.

Locate these errors and suggest a corrected piece of code.

1 .....

.....

.....

2 .....

.....

.....

3 .....

.....

..... [3]

5 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
TN371	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) 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]

6 **(a)** Write an algorithm, using pseudocode or flowchart only, which:

- inputs three numbers
- outputs the **largest** of the three numbers

.....

.....

.....

.....

.....

.....

.....

.....

..... [3]

(b) Write an algorithm, using pseudocode or flowchart only, which:

- inputs 1000 numbers
- outputs how many of these numbers were whole numbers (integers)
  (You may use INT(x) in your answer, e.g. y = INT(3.8) gives the value y = 3)

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [4]

(c) Describe, with examples, **two** sets of test data you would use to test your algorithm.

1 .....

.....

2 .....

..... [2]

7 A database was set up to show the properties of certain chemical elements. Part of the database is shown below.

Name of element	Element symbol	Atomic number	Atomic weight	Melting point (C)	Boiling point (C)	State at room temp
oxygen	O	8	16	–218	–183	gas
iron	Fe	26	56	1538	2861	solid
mercury	Hg	80	201	–38	356	liquid
bromine	Br	35	80	–7	59	liquid
osmium	Os	76	190	3033	5012	solid
caesium	Cs	55	133	28	671	solid
gallium	Ga	31	70	30	2204	solid
argon	Ar	18	40	–189	–186	gas
silver	Ag	47	108	961	2162	solid

(a) How many fields are in each record?

..... [1]

(b) The following search condition was entered:

(**Melting point (C)** < 40) AND (**Atomic weight** > 100)

Using **Element symbol** only, which records would be output?

.....

..... [2]

(c) Which field would be best suited as primary key?

..... [1]

BLANK PAGE