Cambridge
O Level

Cambridge International Examinations
Cambridge Ordinary Level

CANDIDATE NAME	
CENTRE NUMBER	
CANDIDATE NUMBER	

6543207907

COMPUTER SCIENCE **2210/23**

Paper 2 Problem-solving and Programming **October/November 2016**

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.

DC (LK/SW) 125804/1
© UCLES 2016

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

The headteacher of a school needs a program to record and count the votes for a class captain. Each student in the class will be allowed one vote. There can be up to 30 students in a class.

Write and test a program for the headteacher.

- Your program must include appropriate prompts for the entry of data.
- Error messages and other output need to be set out clearly.
- 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 the voting

Each class can choose from up to four different students as candidates for their class captain. Set up a routine that allows:

- between two and four different candidate names to be input and stored
- the candidate names to be displayed with a number (1, 2, 3 or 4) beside each name
- a choice of 1, 2, 3 or 4 to be entered to record a vote; all other entries are rejected
- up to four totals set to zero ready to record the votes
- a maximum of 30 votes to be cast

TASK 2 – Record and count each vote

Input a number 1, 2, 3 or 4; add 1 to the appropriate total.

Output the name of the candidate voted for or output 'invalid vote' if a vote is rejected.

TASK 3 – Show the result

When all the votes have been cast, display the candidates' names with their totals in descending order of totals. If there is a clear winner, display the candidate's name with the words 'NEW CLASS CAPTAIN' beside it; otherwise display 'NO OVERALL WINNER'.

© UCLES 2016

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

For **four** of the variables, constants or arrays that you used in **Task 1**, state the name, data structure, data type and its use.

Name 1 .....

Data structure .....

Data type .....

Use .....

Name 2 .....

Data structure .....

Data type .....

Use .....

Name 3 .....

Data structure .....

Data type .....

Use .....

Name 4 .....

Data structure .....

Data type .....

Use .....[8]

(b) Write an algorithm to complete **Task 2**, using **either** pseudocode, programming statements **or** a flowchart**.** You can assume that Task 1 has been completed and that there are three candidates for class captain.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[4]

(c) Explain how you show the result (**Task 3**). You may include programming statements as part of your explanation.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....[6]

(d) Explain how you could extend your solution to deal with the case of 'NO OVERALL WINNER'.

.....

.....

.....

.....[2]

Section B

2 Read this section of program code that:

- inputs 10 numbers
- checks whether each number is within a specified range
- totals the numbers within the range and outside the range

```
1   InRange = 0
2   OutRange = 1000
3   FOR Count = 1 TO 10
4     INPUT Num
5     IF Num > 10 AND Num < 20 THEN InRange = InRange + 1
6       ELSE OutRange = OutRange - 1
7   Count = Count + 1
8   NEXT X
9   PRINT InRange, OutRange
```

(a) There are four errors in this code.

Locate these errors and suggest a correction to remove each error.

Error 1 .....

Correction .....

.....

Error 2 .....

Correction .....

.....

Error 3 .....

Correction .....

.....

Error 4 .....

Correction .....

.....[4]

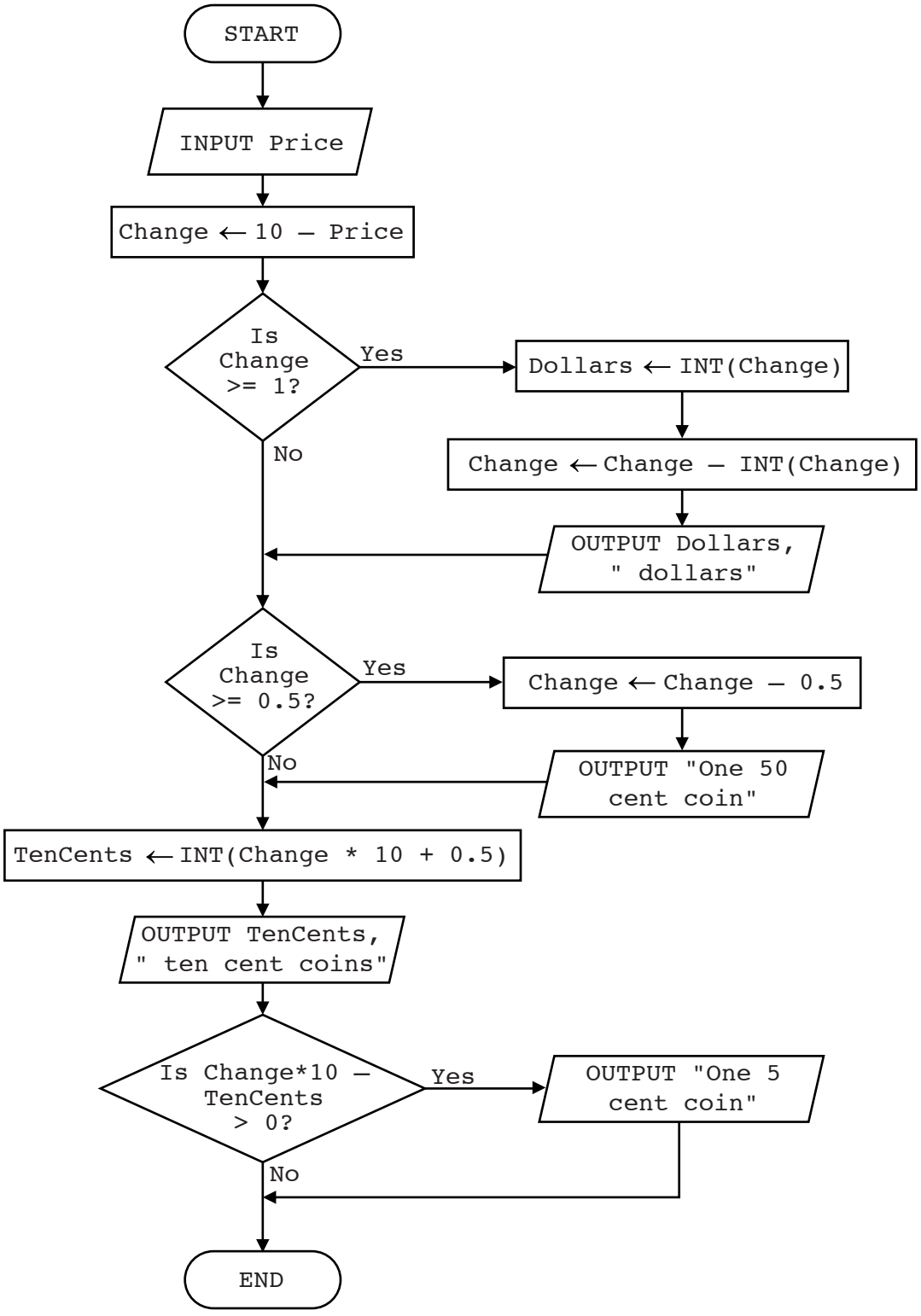
(b) Decide, with reasons, whether the numbers 10 and 20 are within or outside the range.

Number	Within range (✓)	Outside range (✓)	Reason
10			…… ……
20			…… ……

[4]

3 The flowchart below inputs the price of an item under $10. The change from a $10 note is output. Any amount less than 5 cents is rounded up to 5 cents.

The predefined function `INT` rounds a number down to the nearest whole number; for example `Z ← INT(5.7)` gives the value `Z = 5`



Complete the trace table for the input data: 6.29

Price	Change	Dollars	TenCents	OUTPUT

[5]

4 Four validation checks and four descriptions are shown below.

Draw a line to link each validation check to the correct description.

Validation check	Description
Presence check	Numbers between two given values are accepted
Range check	Data is of a particular specified type
Type check	Data contains an exact number of characters
Length check	Ensures that some data have been entered

[3]

5 `REPEAT ... UNTIL` and `WHILE ... DO ... ENDWHILE` are two different loop structures you can use when writing pseudocode.

Explain, using examples, why you would choose to use each type of loop.

Example 1 .....

.....

.....

.....

Reason for choice .....

.....

.....

Example 2 .....

.....

.....

.....

Reason for choice .....

.....

.....[6]

6 A database, THEATRETOURS, was set up to show the tour dates, towns, number of seats and prices in local currency for a Shakespeare play.

Town	Tour Date	Number of Seats	Price Local Currency
Wigan	18/08/2016	120	15.00
Dumfries	20/08/2016	160	12.50
Turin	25/08/2016	200	17.00
Macon	27/08/2016	75	18.00
Bordeaux	29/08/2016	170	20.00
Algiers	01/09/2016	125	1350.00
Windhoek	05/09/2016	65	90.00
Windhoek	06/09/2016	65	90.00
Port Elizabeth	10/09/2016	200	110.00

(a) Explain why none of the fields in the database can be used as a primary key.

.....

.....

.....[2]

(b) State a field that could be added as a primary key.

.....

Give a reason for choosing this field.

.....

.....[2]

(c) Use the query-by-example grid below to provide a list of tour dates and seat prices in alphabetical order of town.

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

[4]

© UCLES 2016

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 2016