CAMBRIDGE INTERNATIONAL EXAMINATIONS

Cambridge Ordinary Level

MARK SCHEME for the October/November 2015 series

2210 COMPUTER SCIENCE

2210/22 Paper 2, maximum raw mark 50

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge will not enter into discussions about these mark schemes.

Cambridge is publishing the mark schemes for the October/November 2015 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

CAMBRIDGE
International Examinations

Section A

1 (a) (i) Any **two** variables with matching uses, **one** mark for the variable name and **one** mark for the matching use. The variables and the matching uses must relate to the tasks on the exam paper. There are many possible correct answers these are examples only.

Variable 1 – `Counter(: INTEGER)`
Use – to use as a loop counter when entering the temperatures
Variable 2 – `BabyTemperature(: REAL)`
Use – to store the baby's temperature [4]

(ii) Any **two** constants with matching uses, **one** mark for the constant (name and value) and **one** mark for the matching use. The constants and the matching uses must relate to the tasks on the exam paper. There are several possible correct answers these are examples only.

Constant 1 – `MinBabyTemperature = 36.0`
Use – to keep the lowest acceptable baby temperature
Constant 2 – `MaxBabyTemperature = 37.5`
Use – to keep the highest acceptable baby temperature [4]

(b) Any **five** from
– prompt for baby's temperature
– input baby's temperature
– test for > 37.5
– … then output suitable message if this is the case
– test for < 36.0
– … then output suitable message if this is the case
– output suitable message if temperature between those values [5]

Sample algorithm:

```
PRINT 'Please enter temperature of baby '
INPUT BabyTemperature
IF BabyTemperature > MaxBabyTemperature or 37.5
   THEN Print 'Temperature too high'
   ELSE
   IF BabyTemperature < MinBabyTemperature or 36.0
      THEN Print 'Temperature too low'
      ELSE Print 'Temperature OK'
   ENDIF
ENDIF
```

(c) (i) Explanation
General marks award as seen
Give **one** mark for a mention of any one of the **4** checks below
If a mark is given for a check then mark the corresponding action taken
Maximum of **five** marks overall

General
– check all recorded temperatures (loop 18 times)
– update counter for those out of range
– output suitable message if counter >= 2

1 **check if temperature range <= 1 and highest recorded not out of range and lowest recorded not out of range**
– ... exit
2 **check if temperature range > 1...**
– … output suitable message e.g. "Temperature range greater than one degree"
3 **check if highest recorded temperature out of range…**
– … output a suitable message if at least two recorded temperatures out of range e.g. "Temperature too high on more than one occasion"
4 **check if lowest recorded temperature out of range…**
– … output a suitable message if at least two recorded temperatures out of range e.g. "Temperature too low on more than one occasion" [5]

(ii) Any **two** from
– only checks necessary conditions
– uses results from task 2
– checks for normal values first [2]

Section B

2 **One** mark for each error identified + suggested correction
line 4 or (`Total =) Total + 1`: this should read (`Total =) Total + Num`
line 5 or `Counter = Counter + 1`: delete this line
line 6 or (`Average = )Total / Counter`: swap lines 6 and 7
line 6 or (`Average = )Total / Counter` : this should read (`Average =) Total / 50`
[4]

3 **(a)**

Number 1 Trace table

X	Posn	New	T1	T2	Output
5	1	0			
	10	1	2	1	
2	100	1	1	0	
		101			
					101

← (1 mark) →← (1 mark) →← (1 mark) →

Number 2 Trace table

X	Posn	New	T1	T2	Output
12	1	0			
	10	0	6	0	
6	100	0	3	0	
3	1000	100	1	1	
		1100			
					1100

← (1 mark) →← (1 mark) →← (1 mark) →

[6]

(b) Converts a (denary) number to binary [1]

4 There are many possible correct answers this is an example only.
Normal e.g. 1.7
Extreme 0.5 **or** 2.0 only
Abnormal e.g. one [3]

5 – `IF (… THEN … ELSE … ENDIF)`
– `CASE (… OF … OTHERWISE … ENDCASE)` [2]

6 **(a)** **(i)** **One** mark for every **two** correct types

Title	– text
Artist	– text
Description	– text/memo
Catalogue Number	– text/(auto)number
Size	– number
Price	– currency/number
Arrived	– date
Sold	– "yes/no"/text/Boolean

0, 1 no marks
2, 3 one mark
4, 5 two marks
6, 7 three marks
8 four marks [4]

(ii) Catalogue Number [1]

(b) **One** mark for each correct **different** check

Catalogue Number	Format check/Presence Check/Check Digit/Length check/uniqueness check
Size	Type check/Presence Check/Range Check
Price	Type check/Presence Check/Range Check
Arrived	Type check/Presence Check/Range Check/Format check/Select from calendar length check [4]

(c)

Field:	Catalogue Number	Title	Price	Artist	Sold
Table:	PICTURE	PICTURE	PICTURE	PICTURE	PICTURE
Sort:					
Show:	☑	☑	☑	☐	☐
Criteria:				='Twister'	False
or:					
	(1 mark)	(1 mark)	(1 mark)	(1 mark)	(1 mark)

[5]