

**MARK SCHEME for the October/November 2011 question paper  
for the guidance of teachers**

**9691 COMPUTING**

**9691/21**

Paper 2 (Written Paper), maximum raw mark 75

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 must be read in conjunction with the question papers and the report on the examination.

- Cambridge will not enter into discussions or correspondence in connection with these mark schemes.

Cambridge is publishing the mark schemes for the October/November 2011 question papers for most IGCSE, GCE Advanced Level and Advanced Subsidiary Level syllabuses and some Ordinary Level syllabuses.

<b>Page 2</b>	<b>Mark Scheme: Teachers' version</b>	<b>Syllabus</b>	<b>Paper</b>
	<b>GCE AS/A LEVEL – October/November 2011</b>	<b>9691</b>	<b>21</b>

1 (a) (i) -Valid data entered CAO [1]

(ii) -Invalid data. Try again CAO [1]

(b) e.g. Pascal

```

READLN (Position);
IF Position = 'd'
  THEN WRITELN('Valid data entered')
  ELSE
    IF Position = 'f'
      THEN WRITELN('Valid data entered')
      ELSE
        IF Position = 'g'
          THEN WRITELN('Valid data entered')
          ELSE WRITELN('Invalid data. Try again');

```

e.g. VB6

```

Position = txtBox.Text
IF Position = "d" THEN
  MsgBox "Valid data entered"
ELSEIF Position = "f" THEN
  MsgBox "Valid data entered"
ELSEIF Position = "g" THEN
  MsgBox "Valid data entered"
ELSE
  MsgBox "Invalid data. Try again"
END IF

```

e.g. VB 2005

```

Position = Console.ReadLine
IF Position = "d" THEN
  Console.WriteLine("Valid data entered")
ELSEIF Position = "f" THEN
  Console.WriteLine("Valid data entered")
ELSEIF Position = "g" THEN
  Console.WriteLine("Valid data entered")
ELSE
  Console.WriteLine("Invalid data. Try again")
END IF

```

<b>Page 3</b>	<b>Mark Scheme: Teachers' version</b>	<b>Syllabus</b>	<b>Paper</b>
	<b>GCE AS/A LEVEL – October/November 2011</b>	<b>9691</b>	<b>21</b>

e.g. C#

```

position = Console.ReadLine()
if (position = "d")
{
    Console.WriteLine("Valid data entered");
}
else if (position = "f")
{
    Console.WriteLine("Valid data entered");
}
else if (position = "g")
{
    Console.WriteLine("Valid data entered");
}
else
{
    Console.WriteLine("Invalid data. Try again");
}

```

*1 mark for correct input*

*1 mark for 1st condition correct*

*1 mark for all conditions correct*

*1 mark for correct output for Valid data entered*

*1 mark for correct output for Invalid data. Try again*

*1 mark for conditions indented*

*No marks for pseudocode (give marks for clear code, even if unidentified)*

Max [5]

**(c)** -Sequence, selection (*in any order, these words only*)

[1]

**(d)** -A process of repeating  
 -A block of statements/number of steps  
 -Until some condition is met

Max [2]

**(e)** *1 mark for a counter variable*  
*1 mark for initialising counter variable*  
*1 mark for addition*  
*1 mark for condition at 3*  
*1 mark for correct symbols*

[5]

(f)

Field Name	Data Type	Field Size (bytes)	
PlayerID	Integer/byte/shortint	a value within 1–6	<i>NOT a range</i>
Sex	Boolean/character	1	
PlayerName	String/Text	a value within 10–50	<i>NOT a range</i>
Position	Character/string	1	
DateOfBirth	Date/Integer/String	2/4/6/8/10	

1 mark per cell

[10]

(g) (i) -logic (error)

[1]

	(ii)	(iii)
EITHER:	ArrayPosition ← 1	ArrayPosition ← 0
OR:	UNTIL ArrayPosition = 30	UNTIL ArrayPosition >30 or UNTIL ArrayPosition = 31

[1]

[1]

(h) Dtotal ← 0  
 ArrayPosition ← 0  
 WHILE ArrayPosition ≤ 30  
   IF Squad[ArrayPosition].Position = 'd'  
   THEN  
     Dtotal ← Dtotal + 1  
   ENDIF  
   ArrayPosition ← ArrayPosition + 1  
 ENDWHILE

1 mark for WHILE at top of loop and correct condition dependent on initial values

1 mark for correct content of IF statement and condition

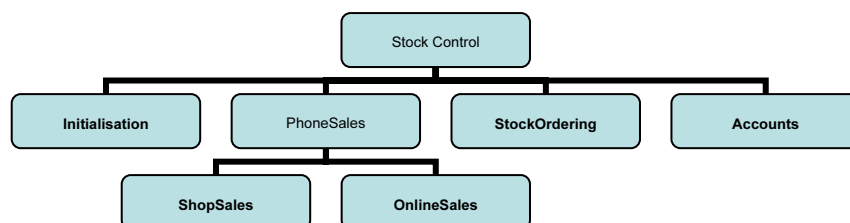
1 mark for ENDWHILE in correct position or equivalent structure

[3]

<b>Page 5</b>	<b>Mark Scheme: Teachers' version</b>	<b>Syllabus</b>	<b>Paper</b>
	<b>GCE AS/A LEVEL – October/November 2011</b>	<b>9691</b>	<b>21</b>

- 2 (a) e.g.  
 -each can work on individual modules  
 -modules can be written in parallel  
 (answer must be specific to this scenario) Max [1]

(b) Each box correctly labelled (Initialisation, StockOrdering, Accounts) Order significant



[1]

(c) 1 mark for 2 boxes under PhoneSales  
 1 mark for correct labelling

[2]

- (d) -these will be local variables  
 -that only have effect in the module they are in // local scope  
 -stored in different memory locations  
 -and have no meaning outside that module Max [2]

(e) (i) -keywords/reserved words  
 -a word in the vocabulary of the language  
 -that can only have the meaning defined in that language Max [1]

(ii) e.g. Visual Basic:  
 -names must begin with a letter  
 -must not contain a space/punctuation characters/certain characters  
 -must be unique in their block/scope  
 -can't be more than 64 characters  
 -can't be a keyword [3]

(iii) Any keyword // word breaking a rule given by the candidate [1]

(f) (i) 17 CAO [1]

(ii) 5\*(a-b) [1]

(iii) Black box CAO [1]

<b>Page 6</b>	<b>Mark Scheme: Teachers' version</b>	<b>Syllabus</b>	<b>Paper</b>
	<b>GCE AS/A LEVEL – October/November 2011</b>	<b>9691</b>	<b>21</b>

- (g) (i) -valid/normal data  
-extreme/boundary data [2]
- (ii) 6 different types of test data sets + 6 sensible reasons  
e.g.  
9999, 1000, 10, 20 all boundary values  
4000 valid data  
  
Reason must relate to the scenario  
  
*Value + correct reason = 1 mark* [6]
- (h) (i) (PrintAccounts = 'y') OR (PrintAccounts = 'n')  
*1 mark for 2 separate correct conditions*  
*1 mark for OR*  
  
Alternative answer:  
PrintAccounts IN ['y','n']  
*2 marks (1 mark for IN, 1 mark for ['y','n'])* [2]
- (ii) (StockID >= 1000) AND (StockID <= 9999)  
*Alternative: (StockID > 999) AND (StockID < 10000)*  
  
*1 mark for 2 separate correct conditions*  
*1 mark for AND*  
  
*Alternative answer:*  
NOT ((StockID < 1000) OR (StockID >9999)) [2]
- (iii) -data outside expected range/invalid data could be entered  
-the program would not find an associated record/data [2]

3 (a)

MyWord	Counter	LENGTH (MyWord)	Temp[1]	Temp[2]	Temp[3]	Surprise
rac						
	1					
		3				
			c			
	2					
				a		
	3					
					r	
						car

1 mark for correct length (3)  
 1 mark for correct Temp[1] (c)  
 1 mark for correct Temp[2] and Temp[3] (a and r)  
 1 mark for correct surprise (car)

Max [3]

(b) -reverses the string

[1]

(c) -assigns return value to Surprise  
 -that value is returned to the function call  
 -name of function used as a variable

Max [2]

(d) -is a subroutine // can be called more than once // can be called from different locations  
 -given a name/identifier  
 -may take parameter values from the program  
 -may return parameter values to the program

Max [3]

(e) (i) -indentation  
 -meaningful/sensible variable names

[2]

(ii) -moves character from its place in MyWord  
 -to the opposite place in Temp  
 -more explanation about Counter – 1

Max [2]

<b>Page 8</b>	<b>Mark Scheme: Teachers' version</b>	<b>Syllabus</b>	<b>Paper</b>
	<b>GCE AS/A LEVEL – October/November 2011</b>	<b>9691</b>	<b>21</b>

- (f) -characters compared in turn  
-from the left hand side/start of each word  
-the first higher code value determines the largest word  
-if 2 words are the same when one ends  
-the other is the larger alphabetically

Max [3]