

Please write clearly in	block capitals.		
Centre number		Candidate number	
Surname			
Forename(s)			
Candidate signature			

# GCSE COMPUTER SCIENCE

Paper 1 - Computational thinking and programming skills

Specimen Assessment MaterialsTime allowed: 2 hours

## Materials

- There are no additional materials required for this paper.
- You must **not** use a calculator.



### Instructions

- Use black ink or black ball-point pen. Use pencil only for drawing.
- Answer all questions.
- You must answer the questions in the spaces provided.
- Do all rough work in this book.
- Cross through any work you do not want to be marked.
- Questions that require a coded solution must be answered in VB.Net

### Information

• The total number of marks available for this paper is 90.

### **Advice**

or the multiple-choice questions, completely fill in the lozenge alongside the appropriate answer.
CORRECT METHOD WRONG METHODS © © 🕏 🕏
you want to change your answer you must cross out your original answer as shown. 🔀
you wish to return to an answer previously crossed out, ring the answer you now wish to select as
own.

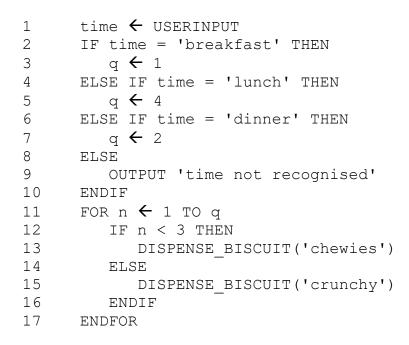
	Answer <b>all</b> questions.		
0 1 . 1	Define the term algorithm.	[2 n	narks]
0 1 . 2	The following are computer science terms (labelled $\mathbf{A} - \mathbf{E}$ ).		
	A assignment B data type C decomposition D efficiency E input  For each of the definitions in the table, write the label of the mocomputer science term. Use a label only once.		narks]
		Label	
	Breaking a problem down into a number of sub-problems		
	The process of setting the value stored in a variable		
	Defines the range of values a variable may take		

0 2	The pseudo	-code in <b>Figure</b> 1	I assigns two	string values to t	wo variables.	
			Figure 1			
		title ← ' level ← '	computer gcse'	science'		
0 2 . 1	Shade one I	lozenge that shov	vs the length	of the contents of	the variable 1	
						[1 mark]
	<b>A</b> 1		0			
	<b>B</b> 2		0			
	<b>C</b> 3		0			
	<b>D</b> 4		0			
0 2 . 2		lozenge that shov able level in Fi		of concatenating t	he variable ti	tle [1 mark]
	A 'com	puter scienc	ce gcse'		0	
	B 'Com	puter Scienc	ce GCSE'		0	
	C 'com	puterscience	egcse'		0	
	D 'com	puter scienc	cegcse'		0	
		Turn over fo	or the next qu	estion		

The algorithm in **Figure 2** has been developed to automate the quantity of dog biscuits to put in a dog bowl at certain times of the day.

• Line numbers are included but are not part of the algorithm.

### Figure 2



Shade **one** lozenge which shows the line number where selection is **first** used in the algorithm shown in **Figure 2**.

[1 mark]

**A** Line number 2

0

**B** Line number 4

0

**C** Line number 9

0

**D** Line number 12

Shade **one** lozenge which shows the line number where iteration is **first** used in the algorithm shown in **Figure 2**.

[1 mark]

**A** Line number 1

0

**B** Line number 8

0

**C** Line number 11

0

**D** Line number 13

0

0 3 . 3	Shade one lozenge which shows how many times the subroutine DISPENSE_BISCUIT would be called if the user input is 'breakfast' in Figure 2.		
	rigule 2.	[1 mark]	
	A 1 subroutine call		
	B 2 subroutine calls		
	C 3 subroutine calls		
	D 4 subroutine calls		
0 3 . 4	Shade <b>one</b> lozenge which shows the data type of the variable time in	ı the	
	algorithm shown in <b>Figure 2</b> .	[1 mark]	
	A Date/Time		
	B String		
	C Integer		
	<b>D</b> Real		
0 3 . 5	State how many times the subroutine <code>DISPENSE_BISCUIT</code> will be with the parameter 'chewies' if the user input is 'lunch' in Figure		
	Turn over for the next question		

A programmer has written a VB.Net program that asks the user to input two integers and then output which of the two integers is the largest. Complete the program below by filling in the gaps using the items in **Figure 3**. You will not need to use all the items in **Figure 3**. Each item in **Figure 3** should only be used once.

[5 marks]

Figure 3

Console.Write	num1	num2	output
Else	<	>	ElseIf
String	Double	Integer	

Dim num1 As Integer
Dim num2 As
Console.Write("Enter a number: ")
<pre>num1 = Console.ReadLine()</pre>
Console.Write("Enter another number: ")
<pre>num2 = Console.ReadLine()</pre>
If num1 > num2 Then
Console.WriteLine(" is bigger.")
ElseIf num1 num2 Then
Console.WriteLine(" is bigger.")
Console.WriteLine("The numbers are equal.")
End If

Write a VB.Net program that allows a taxi company to calculate how much a taxi fare should be.

The program should:

- allow the user to enter the journey distance in kilometres (no validation is required)
- allow the user to enter the number of passengers (no validation is required)
- calculate the taxi fare by
  - o charging £2 for every passenger regardless of the distance
  - charging a further £1.50 for every kilometre regardless of how many passengers there are
- output the final taxi fare.

You **should** use meaningful variable name(s), correct syntax and indentation in your answer.

The answer grid below contains vertical lines to help you indent your code accurately.

		[7 ma	arks]

Do not write outside the box

Write a VB.Net program that inputs a password and checks if it is correct.

Your program should work as follows:

- input a password and store it in a suitable variable
- if the password entered is equal to secret display the message Welcome
- if the password entered is not equal to secret display the message Not welcome.

You **should** use meaningful variable name(s), correct syntax and indentation in your answer.

The answer grid below contains vertical lines to help you indent your code accurately.

[5 marks]

Do not write outside the box

**0 7** The algorithm in **Figure 4** is a sorting algorithm.

- Array indexing starts at 0.
- Line numbers are included but are not part of the algorithm.

# Figure 4

```
1
     arr \leftarrow [4, 1, 6]
2
     swapsMade ← false
3
     WHILE swapsMade = false
4
         swapsMade ← true
5
         i ← 0
6
        WHILE i < 2
7
            IF arr[i+1] < arr[i] THEN</pre>
8
                t \leftarrow arr[i]
9
                arr[i] \leftarrow arr[i+1]
10
                arr[i+1] \leftarrow t
11
                swapsMade \leftarrow false
12
            ENDIF
13
            i ← i + 1
14
         ENDWHILE
15
     ENDWHILE
```

0 7 . 1 State the data type of the variable swapsMade in the algorithm shown in Figure 4.

[1 mark]

0 7 . 2	The identifier swapsMade is used in the algorithm shown in Figure 4.

Explain why this is a better choice than using the identifier  ${\tt s.}$ 

		_

[2 marks]

0 7 . 3		one loz			which of the follo	owing contains	the <b>false</b> state	ement
		J					[1	mark]
	A	The algo	orithm ι	uses a r	named constant.		0	
	В	The algo	orithm (	uses inc	lefinite iteration.		0	
	C	The algo	orithm ι	uses ne	sted iteration.		0	
0 7 . 4		lete the already l			the algorithm sho	own in <b>Figure</b>		es narks]
			arr		swapsMade	i	t	
		[0]	[1]	[2]	Swapsilaac		C	
		4	1	6	false			

Write a VB.Net program that inputs a character and checks to see if it is lowercase or not.

Your program should work as follows:

- gets the user to enter a character and store it in a suitable variable
- determines if the entered character is a lowercase character
- outputs LOWER if the user has entered a lowercase character
- outputs NOT LOWER if the user has entered any other character.

You **should** use meaningful variable name(s), correct syntax and indentation in your answer.

The answer grid below contains vertical lines to help you indent your code accurately.

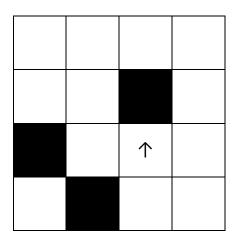
[7 marks]

			14			
0 9	Four separate subroutin  Forward(n) n  TurnLeft() to TurnRight()  ObjectAhead square or returns	noves the urns the r turns the () returr	e robot n robot 90 e robot 90 is true	squares degrees degree if the rol	forward left. s right. oot is fac	
0 9 . 1	Draw the path of the ro executed (the robot stathe arrow).					ollowing program is `facing in the direction of
	Forwar TurnLe Forwar TurnRi Forwar	ft() d(1) ght()				
						[3 marks]
					1	

0 9 . 2 Draw the path of the robot through the grid below if the following program is executed (the robot starts in the square marked by the ↑ facing in the direction of the arrow). If a square is black then it contains an object.

```
WHILE ObjectAhead() = true
  TurnLeft()
    IF ObjectAhead() = true THEN
        TurnRight()
        TurnRight()
        ENDIF
    Forward(1)
ENDWHILE
Forward(1)
```

[3 marks]



Turn over for the next question

3 5 1 4
1 2 3 4

1 2		eveloper is developing a program for wing instructions.	a client. The developer is giv	en the
	pay dogs extra beca owe reall	ny of my friends ask me to walk thei me to do this and the amount I get p is for. If they have more than one do a. I like to walk the dogs in the after ause I often get colds. I need you to d – fortunately for me all of my frien y easy to tell them apart. I charge £ I I always round this up so 47 minute	paid depends on how long I want og then I don't charge the owner noon when the weather is norm to help me keep track of how m ds have different first names s 210 for every 30 minutes of the	alk their er any nally best uch l'm o it is e walk
1 2 . 1		developer needs to remove all of th est. Shade the lozenge next to the i	<u> </u>	client's [1 mark]
	Α	Abstraction	0	
	В	Conversion	0	
	С	Decomposition	0	
	D	Validation	0	
1 2 . 2		developer has decided that the followils from the client's request.  The charge is based on time and the charge is £10 every 30 minutes.	d not how many dogs are walk	
	Stat	e <b>two</b> other relevant details that the		[2 marks]

1 3	The following subroutines different columns.	control the way th	at labelled blocks are placed in
	BLOCK_ON_	_TOP(column)	returns the label of the block on top of the column given as a parameter.
	MOVE(source,	destination)	moves the block on top of the source column to the top of the destination column.
	HE	IGHT(column)	returns the number of blocks in the specified column.
1 3 . 1	This is how the blocks A,	B and C are arran	ged at the start.
	Column 0	Column 1	Column 2
	C B A		
	Draw the final arrangeme	nt of the blocks aft	er the following algorithm has run.
	MOVE (0, 1) MOVE (0, 2) MOVE (0, 2)		
	Column 0	Column 1	Column 2
			[3 marks]

This is how the blocks A,	B and C are arranged at t	he start.
Column 0	Column 1	Column 2
Draw the final arrangement	ent of the blocks after the fe	ollowing algorithm has run.
Column 0	Column 1	Column 2
		[3 marks]
Turn o	ver for the next question	
Turn o	ver for the next question	
	Column 0  C B A  Draw the final arrangement while Height Move(0, 1 ENDWHILE MOVE(1, 2)  Column 0	Draw the final arrangement of the blocks after the few MHILE HEIGHT (0) > 1  MOVE (0, 1)  ENDWHILE  MOVE (1, 2)

1	3 . 3	Develop an algorithm ເ every block from colum		le or a flowchart that will move
			always be at least one	ocks start in column 0. You block in column 0 at the start
		The order of the blocks	s must be preserved.	
				a block from one column to ubroutine in your answer.
		For example, if the sta	rting arrangement of the	e blocks is:
		Column 0	Column 1	Column 2
		BA		
		Then the final arranger	ment should have block	B above block A:
		Column 0	Column 1	Column 2
			BA	
				[4 marks]

A programmer has written the VB.Net program in Figure 5 to add up the 1 4 numbers between one and five.

# Figure 5

Dim total As Integer = 0Dim number As Integer = 0For number = 1 To 5total = total + numberNext Console.WriteLine(total)

The program needs to be changed so that it also multiplies all of the numbers between one and five.

Shade one lozenge next to the program that will do what the programmer wants.

		[1 mark
A	Dim total As Integer = 0 Dim product As Integer = 1 Dim number As Integer = 0 For number = 1 To 5 total = total + number product = total * number  Next Console.WriteLine(total)	0
В	Console.WriteLine(product)  Dim total As Integer = 0  Dim product As Integer = 1  Dim number As Integer = 0  For number = 1 To 5  total = total + number  product = product * number  Next  Console.WriteLine(total)  Console.WriteLine(product)	0
С	<pre>Dim total As Integer = 0 Dim product As Integer = 1 Dim number As Integer = 0 For number = 1 To 5     total = total + number     product = product * total Next Console.WriteLine(total) Console.WriteLine(product)</pre>	0
D	Dim total As Integer = 0 Dim product As Integer = 1 Dim number As Integer = 0 For number = 1 To 5     total = total + number     product = (total + product) * number  Next Console.WriteLine(total) Console.WriteLine(product)	0

A program has been written in VB.Net to display all the odd integers between 1 and the largest odd number smaller than an integer entered by the user. The program is shown in **Figure 6**.

# Figure 6

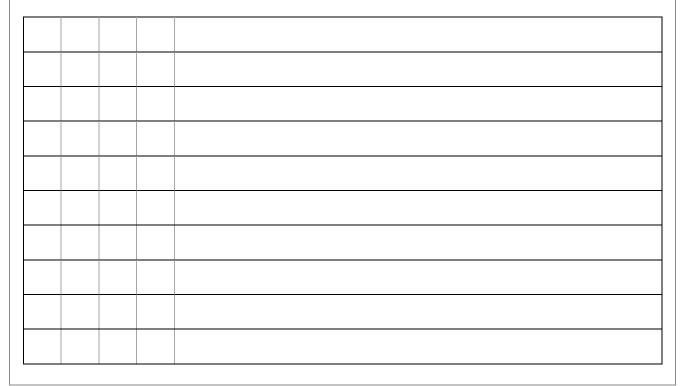
```
Dim odd As Integer = 1
Dim number As Integer
Console.Write("Enter an integer: ")
number = Console.ReadLine()
While odd <> number
   Console.WriteLine(odd)
   odd = odd + 2
End While
Console.WriteLine("Finished!")
```

The program works correctly if the integer entered by the user is an odd, positive integer. For example, if 7 is entered the program correctly displays the values 1, 3 and 5

The program does not work correctly if an odd integer less than 1 is entered by the user. For example, when -7 is entered the program should display the values 1, -1, -3 and -5 but it doesn't do this.

Using VB.Net only, change the program code inside the while loop so that it will work correctly for any odd integer entered by the user.

[4 marks]



Do not write outside the box

# Figure 7 shows part of a program written in VB.Net.

# Figure 7

```
Dim validChoice As Boolean
Dim choice As Integer
validChoice = False
While validChoice = False
    Console.Write("Enter your choice [1 - 10] ")
    choice = Console.ReadLine()
    If choice >= 1 And choice <= 10 Then
        validChoice = True
    Else
        Console.WriteLine("Invalid choice")
    End If
End While
Console.WriteLine("Valid choice")</pre>
```

Complete the following test plan for the code shown in **Figure 7**.

Test type	Test data	Expected result	
Normal data	5	Valid choice message displayed	
Invalid data			
Boundary data			

[2 marks]

Figure 8 shows a VB.Net program that is being developed.

It is supposed to calculate and display the highest common factor of two numbers entered by the user.

The highest common factor of two numbers is the largest number that both numbers can be divided by without leaving a remainder.

### Examples:

- the highest common factor of the numbers 6 and 9 is 3
- the highest common factor of 2 and 5 is 1

Line numbers are shown but are not part of the program code.

# Figure 8

```
1
  Dim num1 As Integer = Console.ReadLine()
  Dim num2 As Integer = Console.ReadLine()
  Dim hcf As Integer = 1
4
  Dim count As Integer = 1
5
  While count < num1
6
     If num1 Mod count = 0 And num2 Mod count = 0 Then
7
       hcf = count
8
     End If
9
     count = count + 1
10 End While
11 Console.WriteLine(hcf)
```

The program in **Figure 8** works correctly sometimes but not always. When the user enters the numbers 4 and 6 it correctly outputs 2, but when the user enters the numbers 4 and 4 it should output 4 but it does not.

1 7 . 1 State the output from the program in **Figure 8** when the user enters the numbers 4 and 4

[1 mark]

1 7 2 State the line number from the program in **Figure 8** which contains the error that stops the program from sometimes working correctly.

[1 mark]

Describe how the line of code identified in your answer to 17.2 should be changed so that the program in **Figure 8** will work correctly.

[1 mark]

Write a VB.Net program that calculates an estimate of the braking distance in metres for a new model of go-kart that is travelling between 10 and 50 kilometres per hour (kph).

Your program should:

- keep asking the user to enter a speed for the go-kart until they enter a speed that is between 10 and 50 (inclusive)
- calculate the braking distance in metres by dividing the speed by 5
- ask the user if the ground is wet (expect the user to enter yes if it is)
- if the ground is wet, multiply the braking distance by 1.5
- output the final calculated braking distance.

You **should** use meaningful variable name(s), correct syntax and indentation in your answer.

The answer grid below contains vertical lines to help you indent your code accurately.

[8 marks]

## **END OF QUESTIONS**

### Copyright information

For confidentiality purposes, all acknowledgements of third-party copyright material are published in a separate booklet. This booklet is published after each live examination series and is available for free download from <a href="https://www.aqa.org.uk">www.aqa.org.uk</a>.

Permission to reproduce all copyright material has been applied for. In some cases, efforts to contact copyright-holders may have been unsuccessful and AQA will be happy to rectify any omissions of acknowledgements. If you have any queries please contact the Copyright Team.

Copyright © 2019 AQA and its licensors. All rights reserved.