



GuideMyGrades

— YOUR GOALS. OUR EXPERTISE. —

EDEXCEL INTERNATIONAL GCSE COMPUTER SCIENCE: QUESTIONS BY TOPIC

Programming (Topic 2)

International GCSE Computer Science Topic 2

24 questions · 378 marks · Mark schemes included

Pearson Edexcel International GCSE (9–1) Computer Science 4CP0, Papers 1 & 2, June 2019 - June 2022

 guidemygrades.com

 @Guidemygrades

 @Guidemygrades

 @Guidemygrades

 info@guidemygrades.com

 Piccadilly Business Centre, Unit C, Aldow Enterprise Park
Blackett Street, Manchester, M12 6AE

HOW TO USE THIS BOOKLET

Questions run from the lowest mark tariff to the highest. Paper 2 answers use the Python mark scheme (2A).

Click any heading or question row below to jump straight to it.

On every question page, click ANSWER in the top-right corner to jump to its mark scheme.

Diagrams, graphs, tables and data are printed exactly as they appeared in the original papers.

QUESTION	SOURCE	MARKS	PAGE
UP TO 10 MARKS QUESTIONS		3 questions · 26 marks	3
1. Q3	Paper 2 June 2022	6	4
2. Q1	Paper 2 November 2020	10	5
3. Q1	Paper 2 June 2021	10	7
11–14 MARKS QUESTIONS		6 questions · 79 marks	9
4. Q5	Paper 2 November 2020	12	10
5. Q4	Paper 2 November 2020	13	12
6. Q1 <i>also Topic 1</i>	Paper 2 November 2021	13	13
7. Q4 <i>also Topic 3</i>	Paper 2 November 2021	13	16
8. Q3 <i>also Topic 1</i>	Paper 2 June 2019	14	17
9. Q1	Paper 2 June 2022	14	18
15+ MARKS QUESTIONS		15 questions · 273 marks	20
10. Q4	Paper 2 June 2021	15	21
11. Q3	Paper 2 November 2021	15	24
12. Q1	Paper 2 June 2019	16	26
13. Q2 <i>also Topic 1</i>	Paper 2 June 2022	16	29
14. Q2 <i>also Topic 1</i>	Paper 2 November 2020	17	31
15. Q3 <i>also Topic 3</i>	Paper 2 June 2021	17	33
16. Q5 <i>also Topic 1</i>	Paper 2 June 2022	17	36
17. Q2 <i>also Topic 1</i>	Paper 2 June 2021	18	39
18. Q2 <i>also Topic 1</i>	Paper 2 November 2021	19	42
19. Q5	Paper 2 June 2019	20	45
20. Q6	Paper 2 November 2020	20	46
21. Q5	Paper 2 June 2021	20	47
22. Q5	Paper 2 November 2021	20	49
23. Q6	Paper 2 June 2022	20	51
24. Q2 <i>also Topic 1</i>	Paper 2 June 2019	23	53



QUESTIONS

24 questions · 378 marks · lowest tariff first



UP TO 10 MARKS QUESTIONS

3 questions in this booklet



3 Manjit sells copies of a science textbook to schools.

She needs a program to process textbook orders. It must:

- accept the input of the number of textbooks required
- generate the price per textbook
- generate the total cost of the order
- display the price per textbook and the total cost of the order.

The price of one textbook depends on the number ordered:

Quantity	Price per textbook
1–5	£20.00
6–9	£15.00
10 or more	£12.00

Open **Q03** in the code editor.

Write the program.

You must use the structure given in **Q03** to complete the program.

Do not add any further functionality.

Save your code as **Q03FINISHED** with the correct file extension for the programming language.

(Total for Question 3 = 6 marks)



1 Computer programs make use of many programming constructs.

(a) Identify which **one** of these is **not** an arithmetic operator.

(1)

- ☐ A add
- ☐ B divide
- ☐ C equals
- ☐ D modulus

(b) State the values used by the Boolean data type.

(1)

- (c) Open **Q01c** in the code editor.

The program should multiply two integers.

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

Amend the code to correct the errors.

Save your amended code as **Q01cFINISHED** with the correct file extension for the programming language.

(3)

- (d) State what is meant by the term 'variable'.

(1)

- (e) Ayesha has written a program that calculates the sum of a set of numbers.

Open **Q01e** in the code editor.

Answer these questions about the code.

- (i) Identify the line number where the code includes a logic operator.

(1)

- (ii) Identify the line number where a subprogram definition starts.

(1)

- (iii) Identify the name of a numeric variable.

(1)

- (iv) Identify the name of a string variable.

(1)

(Total for Question 1 = 10 marks)



1 Computer programs make use of many programming constructs.

(a) Identify which **one** of these is a logic operator.

(1)

☐ **A** modulus

☐ **B** OR

☐ **C** >=

☐ **D** =

(b) Describe **one** difference between a one-dimensional array and a two-dimensional array.

(2)

.....

.....

.....

.....

(c) Open **Q01c** in the code editor.

The program should calculate how many £20 are in £113 and how much is left over.

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

Amend the code to correct the errors.

Save your amended code as **Q01cFINISHED** with the correct file extension for the programming language.

(3)

(d) Raoul has written a program that counts the number of vowels in a sentence.

Open **Q01d** in the code editor.

Amend the code by adding a suitable comment (you may need to add more than one comment to a line):

(i) at the end of the line where there is a **relational operator** (1)

(ii) at the end of a line where **iteration** starts (1)

(iii) at the end of the line where **selection** starts (1)

(iv) at the end of a line where a **data structure** is initialised. (1)

Save your amended code as **Q01dFINISHED** with the correct file extension for the programming language.

(Total for Question 1 = 10 marks)



11–14 MARKS QUESTIONS

6 questions in this booklet



5 Cerys Jones is the manager of a shop that sells electrical goods.

She is creating a stock control program.

(a) Each item of stock will have a ten-character code.

Each code consists of three parts.

Part	Description
Characters 1 to 3	Uppercase letters
Characters 4 to 9	Numbers between 1 and 9 inclusive
Character 10	'e' or 'o' indicating whether the sum of the numbers is even or odd

WAS123456o, FDG232324e and VAC129222e are valid codes.

All codes need to be validated.

Complete the table to show **two** additional validation tests.

For each test, give **one** example of erroneous data. The example that you give should fail **only** that test.

(4)

Validation test	Erroneous data
Is the last character 'o' or 'e'?	ABC654321v

(b) Cerys will need an algorithm to search the table of stock data.

(i) State **two** advantages of using a linear search algorithm rather than a binary search algorithm.

(2)

1

.....

2

.....



(ii) State **two** disadvantages of using a linear search algorithm rather than a binary search algorithm.

(2)

1

.....

2

.....

(c) Cerys will use functions and procedures in her program.

(i) One similarity between a function and a procedure is that they are both subprograms.

State **two** other similarities.

(2)

1

.....

2

.....

(ii) Explain **one** difference between a function and a procedure.

(2)

.....

.....

.....

.....

(Total for Question 5 = 12 marks)



4 Gianluca is a schoolteacher. He is in charge of a year group of pupils.

(a) He has started to create a program to analyse test results.

Open **Q04a** in the code editor.

Amend the code to complete the 'if statement' used to produce the outputs described in this table.

Condition	Output text
English test score was less than 40; Maths test score was less than 50	Student failed both tests
English test score was less than 40 or Maths test score was less than 50	Student failed one test
English test score was at least 80; Maths test score was at least 85	Student passed both tests with distinction
Other conditions	Student passed both tests

Do not add any further functionality.

Save your code as **Q04aFINISHED** with the correct file extension for the programming language.

(6)

(b) Gianluca has started to create a program to analyse pupil attendance.

He wants to use a menu system with subprograms.

Open **Q04b** in the code editor.

The program must include subprograms to:

- display the **names** of students whose percentage attendance was less than 75%
- count and display the **number** of students whose percentage attendance was 90% or higher.

Complete the program code to implement these requirements using the structure given in **Q04b**.

Do not add any further functionality.

Save your code as **Q04bFINISHED** with the correct file extension for the programming language.

(7)

(Total for Question 4 = 13 marks)



1 Programmers write code to solve problems.

- (a) Give **one** technique that programmers use to make their code easy to read and understand.

(1)

- (b) Programs may contain logic, syntax or runtime errors.

A program has three errors.

These may be logic, syntax or runtime errors.

Complete the table by adding a tick (✓) for **each** error description to show whether the error is a logic, syntax or runtime error.

(3)

Error description	Logic	Syntax	Runtime
Divide by 0			
Use x instead of * to multiply			
Subtract 10 from 2 instead of 2 from 10			

(c) **Figure 1** shows a programming construct in a flowchart.

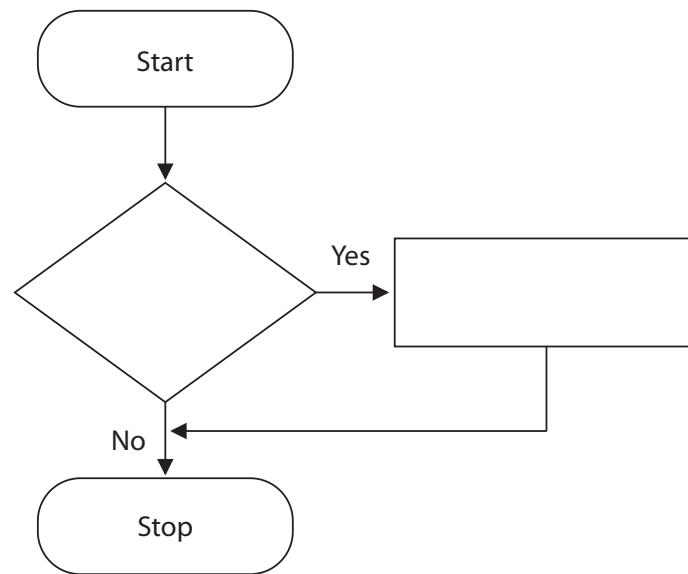


Figure 1

Identify the name of the programming construct.

(1)

- ☐ **A** Array
- ☐ **B** Iteration
- ☐ **C** Operator
- ☐ **D** Selection

(d) Open **Q01d** in the code editor.

The program inputs two numbers, compares them and outputs a suitable message.

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

Amend the code to correct the errors.

Save your amended code as **Q01dFINISHED** with the correct file extension for the programming language.

(3)

- (e) A program is needed to calculate and print a multiplication table for a number that is input by the user.

Figure 2 shows the display when the number 2 is input.

```
Enter the number: 2
1 x 2 is 2
2 x 2 is 4
3 x 2 is 6
4 x 2 is 8
5 x 2 is 10
6 x 2 is 12
7 x 2 is 14
8 x 2 is 16
9 x 2 is 18
10 x 2 is 20
11 x 2 is 22
12 x 2 is 24
```

Figure 2

Open **Q01e** in the code editor.

Write the program.

You **must** use the structure given in the file **Q01e** to complete the program.

Do not add any further functionality.

Save your code as **Q01eFINISHED** with the correct file extension for the programming language.

(5)

(Total for Question 1 = 13 marks)



4 Programs can be used to manipulate data.

(a) A program is needed to convert an 8-bit binary number into denary.

The program must:

- receive the input of a binary pattern and validate that it has exactly eight characters (there is no need to validate that it only contains 0s and 1s)
- request the input again if the binary pattern does not have exactly eight characters
- when a valid binary pattern is input:
 - convert the binary pattern into its denary equivalent
 - print a message that contains the binary pattern and its denary equivalent
- include a comment to explain how the conversion process works.

HINT

Denary placeholder	128	64	32	16	8	4	2	1
Binary number	1	0	1	1	0	1	0	0
Denary number	128	0	32	16	0	4	0	0

$$128 + 32 + 16 + 4 = 180$$

Open **Q04a** in the code editor.

Write a program to implement these requirements.

You must use the structure given in **Q04a** to write the program.

Do not add any further functionality.

Save your code as **Q04aFINISHED** with the correct file extension for the programming language.

(11)

(b) Open **Q04b** in the code editor.

Describe the purpose of the program.

(2)

.....

.....

.....

.....

(Total for Question 4 = 13 marks)

3 Different algorithms can be used to manipulate data.

(a) **Email.txt** contains a list of email addresses.

Open **Q03a** in the code editor.

Write a program to implement these requirements.

The code must:

- check each email address to ensure it contains the '@' symbol.
- write email addresses that do not contain the '@' symbol to an **Error.txt** file.

You must use the structure given in the file **Q03a** to complete the program.

Do not add further functionality.

Save your code as **Q03aFINISHED** with the correct file extension for the programming language.

(6)

(b) Explain **one** drawback of using the merge sort algorithm to sort large data sets.

(2)

.....

.....

.....

.....

(c) Open **Q03c** in the code editor.

Write a program to display the square and cube of a number between 1 and 50 entered by a user.

The code must:

- ask the user to enter a number between 1 and 50 inclusive
- display the number, the square of the number and the cube of the number, with appropriate labels
- stop when a number outside the range 1 to 50 is entered.

Save your code as **Q03cFINISHED** with the correct file extension for the programming language.

(6)

(Total for Question 3 = 14 marks)



1 Programmers write code to solve problems.

(a) Identify the description of a variable in a computer program.

(1)

- ☐ A A value that cannot be used more than once
- ☐ B A value that must be input
- ☐ C A value that is always used in a calculation
- ☐ D A value that can change

(b) Identify the technique that improves the readability of code.

(1)

- ☐ A Using indents on every line
- ☐ B Using descriptive names for variables
- ☐ C Using the correct operators
- ☐ D Using suitable data structures

(c) Complete the table by giving an example value for **each** data type.

The first row has been completed for you.

(2)

Data type	Example value
integer	12
char	
real	

- (d) Describe **one** difference between the data used in boundary testing and the data used in erroneous testing.

(2)

- (e) A program should output the value 2

However, there is an error in the code and the actual output is 4

Name this type of error.

(1)

- (f) Open **Q01f** in the code editor.

The program should allow the input of the length of the side of a square and output the area of the square.

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

Amend the code to correct the errors.

Save your amended code as **Q01fFINISHED** with the correct file extension for the programming language.

(3)

- (g) A program is needed that will accept the input of a letter and compare it with a stored letter.

It will check whether the letter comes earlier in the alphabet, later in the alphabet or is the same letter as the stored letter.

It will output the letter and the stored letter with an appropriate message.

Open **Q01g** in the code editor.

Amend the code to complete the if statement used to produce the output.

You must use the structure given in **Q01g** to complete the program.

Do not add any further functionality.

Save your code as **Q01gFINISHED** with the correct file extension for the programming language.

(4)

(Total for Question 1 = 14 marks)



15+ MARKS QUESTIONS

15 questions in this booklet



4 Zak sells snacks at an after-school club.

(a) He wants a program that will hold details of the products he sells.

The program must:

- generate a five-character product code that:
 - starts with the first three letters of the product name
 - ends with a random number between 10 and 30
- display the product code followed by the product name.

Open **Q04a** in the code editor.

Write the program.

You must use the structure given in the file **Q04a** to complete the program.

Do not add any further functionality.

Save your code as **Q04aFINISHED** with the correct file extension for the programming language.

(7)

(b) Zak plans to implement a binary search algorithm to search a table of products.

(i) Explain **one** advantage of a binary search compared to a linear search.

(2)

.....

.....

.....

.....

(ii) **Figure 2** shows an algorithm for a binary search.

```

1  #   Initialise variables
2  SET productSearch TO " "
3  SET startPosition TO 1
4  SET midPosition TO 0
5  SET endPosition TO LENGTH(productList)
6  SET found TO False
7
8  #   Print prompts, take and check input from user
9  SEND "Enter the product code" TO DISPLAY
10 RECEIVE productSearch FROM (STRING) KEYBOARD
11 REPEAT
12     midPosition = (startPosition + endPosition) DIV 2
13     IF productList[midPosition] < productSearch THEN
14         startPosition = midPosition + 1
15     END IF
16     IF productList[midPosition] > productSearch THEN
17         endPosition = midPosition - 1
18     END IF
19 UNTIL productList[midPosition] = productSearch OR startPosition = endPosition

```

Figure 2

This binary search algorithm will be used to search the product list for the product code **str15**.

Complete the table to indicate the order in which the product codes will be examined by the algorithm.

Write the number 1 by the first product code to be examined, 2 by the second code to be examined, and so on.

(4)

Position in list	Product code	Order examined
1	ark11	
2	asp11	
3	bar13	
4	dri15	
5	mil19	
6	rib10	
7	str15	
8	tor16	

(iii) Zak has another list containing the names of five students who attend the after-school club.

Give the maximum number of names that would need to be examined by the binary search algorithm to determine whether a name appears in the list.

(1)

(iv) Give the name of an algorithm that could be used to sort a list.

(1)

(Total for Question 4 = 15 marks)



3 Dates are often used in programs.

- (a) A program accepts the number of a month and outputs the number of days in the month.

Open **Q03a** in the code editor.

Use the code to answer these questions.

- (i) Identify the name of a local variable.

(1)

- (ii) Identify the name of a data structure.

(1)

- (iii) Identify the name of a built-in subprogram.

(1)

- (iv) Identify the name of a user-written function.

(1)

- (v) Identify the name of a parameter.

(1)

- (vi) The program includes validation to ensure the month number is between 1 and 12 inclusive.

Complete the table by adding the type of test data that is being used to test the validation.

The first type of test has been completed for you.

(2)

Type of test data	Test data
Normal	3
	1
	15

(b) Explain **one** reason why parameters are used to pass data into subprograms.

(2)

(c) A leap year is a year that has 366 days rather than 365 days. A leap year usually occurs every four years.

A year that is a leap year:

- can be exactly divided by 400

Or

- can be exactly divided by 4, but **not** exactly divided by 100.

These years are leap years: 2000, 2012 and 2020.

These years are not leap years: 1700, 1900 and 2021.

A program is needed that will accept the input of a year and output whether it is a leap year or not.

Open **Q03c** in the code editor.

Write the program.

You **must** use the structure given in **Q03c**.

Do not add any further functionality.

Save your code as **Q03cFINISHED** with the correct file extension for the programming language.

(6)

(Total for Question 3 = 15 marks)



1 Computer programs use several different programming constructs.

(a) Identify the **most appropriate** data type for the value 4.5

(1)

- ☐ A Boolean
- ☐ B Character
- ☐ C Real
- ☐ D String

(b) Give a definition of a logic error.

(1)

(c) Open **Q01c** in the code editor.

The program should print out a counter and the counter with 7 added to it.

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

Amend the code to correct the errors.

Save your amended code as **Q01cFINISHED** with the correct file extension for the programming language.

(3)

(d) Give **one** reason why computer programmers should use comments in their programs.

(1)

- (e) Describe the difference between a local variable and a global variable in a computer program.

(2)

- (f) Trevor is the manager of a shoe shop.

Sales assistants can earn a bonus based on the numbers of pairs of shoes they sell and the total income for the shop each day.

Open **Q01f** in the code editor.

Amend the code to complete the 'if statement' used to produce the outputs described in the table.

Condition	Output
Shop income is more than £5000 or sales assistant has sold at least 10 pairs of shoes	Bonus is 10% of salary
Shop income is £2000 or more and sales assistant has sold at least 5 pairs of shoes	Bonus is 5% of salary

Do not add any further functionality.

Save your amended code as **Q01fFINISHED** with the correct file extension for the programming language.

(4)



(g) Car registration codes consist of seven characters.

The characters are two upper case letters followed by two numbers and then three upper case letters.

YH62KLV, EY09VPB and CK53PRJ are valid codes.

All codes need to be validated.

Complete the table to show **two** additional validation tests.

For each test give **one** example of erroneous data. The example that you give should fail **only** that test.

(4)

Validation test	Erroneous data
Is it 7 characters?	AB12CD

(Total for Question 1 = 16 marks)

- 2 Raza is writing a program to tell the user whether a number they input is a prime number.

A prime number is a whole number, larger than one, that can only be divided by one and itself with no remainder.

This pseudocode contains the logic required to complete the program.

```
1 FUNCTION checkPrime(pNumber)
2 BEGIN FUNCTION
3     IF pNumber = 1 THEN
4         check = False
5     ELSE
6         check = True
7         FOR count FROM 2 TO pNumber DO
8             IF pNumber MOD count = 0 THEN
9                 check = False
10            END IF
11        END FOR
12    END IF
13 RETURN check
14 END FUNCTION
15
16 SEND "Enter a number: " TO DISPLAY
17 RECEIVE number FROM (INTEGER) KEYBOARD
18 SET result TO checkPrime(number)
19
20 IF result = True THEN
21     SEND (number & " is a prime number") TO DISPLAY
22 ELSE
23     SEND (number & " is not a prime number") TO DISPLAY
24 END IF
```

(a) Answer these questions about the pseudocode.

(i) Identify a line number where **selection** starts.

(1)

(ii) Identify the line number where **iteration** starts.

(1)

(iii) Identify a line number where a **string** and a **number** are printed on the same line.

(1)

(iv) Identify the name of a **local variable**.

(1)

(v) Identify the name of a **parameter**.

(1)

(b) Write a program to implement the logic in the pseudocode.

Open **Q02b** in the code editor.

Write the program.

You must use the structure given in **Q02b** to complete the program.

Do not add any further functionality.

Save your code as **Q02bFINISHED** with the correct file extension for the programming language.

(11)

(Total for Question 2 = 16 marks)



2 Rama wants some computer programs for his daughter, Ayomi.

(a) Rama wants a program to calculate the area of a triangle.

(area of a triangle = $\frac{1}{2}$ base x height)

This pseudocode contains the logic required to create the program.

```
1 # Initialise variables
2
3 SET base TO 50
4 SET heightChk TO TRUE
5
6 # Print prompt and take input from user
7
8 WHILE heightChk DO
9     SEND "Enter the height (between 1 and 100):" TO DISPLAY
10    RECEIVE height FROM (INTEGER) KEYBOARD
11    IF (height >= 1 AND height <= 100)
12        THEN heightChk = FALSE
13    END IF
14 END WHILE
15
16 # Calculate and print out values
17
18 SET area TO 0.5 * base * height
19
20 SEND ("Base of triangle is: " , base) TO DISPLAY
21 SEND ("Height of triangle is: " , height) TO DISPLAY
22 SEND ("Area of triangle is: " , area) TO DISPLAY
```



Write a program to implement the logic in the pseudocode.

Open **Q02a** in the code editor.

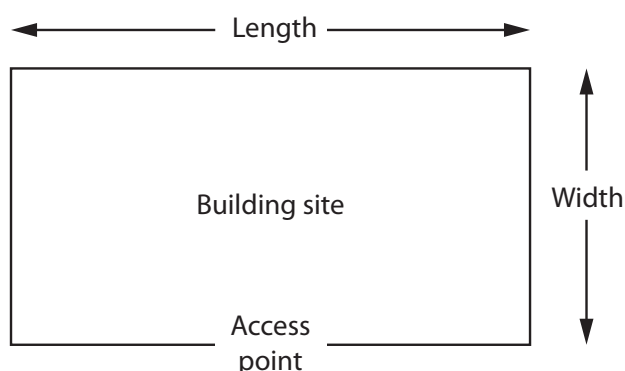
You must use the structure given in **Q02a** to write the program.

Do not add any further functionality.

Save your code as **Q02aFINISHED** with the correct file extension for the programming language.

(10)

- (b) Ayomi wants a program that will calculate the number of fence panels required to surround building sites such as the one shown in the diagram (not to scale).



Each building site is a rectangle.

Each fence panel is 1 metre long.

A gap of 4 metres is needed to access each site. The location of the gap is not significant.

The user will input the length and width of the building site as whole numbers of metres.

The program will output the minimum number of full panels needed.

Q02b provides a structure for the program.

Open **Q02b** in the code editor.

Write the program.

You must use the structure given in **Q02b** to write the program.

Do not add any further functionality.

Save your code as **Q02bFINISHED** with the correct file extension for the programming language.

(7)

(Total for Question 2 = 17 marks)

3 Several encryption algorithms have been developed.

(a) Ahmed has started to create a Caesar cipher encryption program.

Caesar cipher encryption works by giving a number value to a key.

Each letter in a plaintext message is replaced by a new ciphertext letter using the key.

For example, as shown in the table, a key value of +2 would change the plaintext message **encrypt** to the ciphertext message **gpetarv**.

Plaintext	e	n	c	r	y	p	t
Ciphertext	g	p	e	t	a	r	v

Ahmed's program already converts from plaintext to ciphertext.

He wants to extend the program to:

- accept the input of a plaintext message
- accept the input of the key, which must be a positive integer between 1 and 25
- write the generated ciphertext to a text file named **Cipher.txt**
- display the ciphertext.

Open **Q03a** in the code editor.

Complete the program code.

You must use the structure given in the file **Q03a** to complete the program.

Do not add any further functionality.

Save your code as **Q03aFINISHED** with the correct file extension for the programming language.

(8)

(b) Explain **one** reason why the Caesar cipher is a weak encryption algorithm.

(2)

.....

.....

.....

.....

- (c) The Vigenere cipher encrypts plaintext using a series of Caesar ciphers, based on the letters of a keyword and a Vigenere square.

Figure 1 shows a Vigenere square.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
b	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a
c	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b
d	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c
e	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d
f	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e
g	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f
h	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g
i	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h
j	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i
k	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j
l	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k
m	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l
n	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m
o	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n
p	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
q	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
r	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
s	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
t	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
u	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
v	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u
w	w	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
x	x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w
y	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x
z	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y

Figure 1

Complete the table to perform a Vigenere decryption of the ciphertext message **flmktrwhee** using the keyword **thirty**.

(4)

Encrypted letter	f	l	m	k	t	r	w	h	e	e
Keyword letter										
Decrypted letter										



- (d) A user-defined function uses a Vigenere cipher to encrypt plaintext into ciphertext.

This pseudocode shows how the function is called.

SET ciphertext TO ciphertext & getCipherLetter(keywordLetter, plaintextLetter)

- (i) State what would be returned by this function.

(1)

.....

.....

- (ii) Give the name of **one** parameter passed to this function.

(1)

.....

- (iii) State what is meant by a **built-in subprogram**.

(1)

.....

.....

(Total for Question 3 = 17 marks)



5 Julia runs a computer gaming club.

(a) She wants a program to check passwords stored in a file.

The file **passwords.txt** contains the list of passwords.

The program must:

- check each password to ensure that:
 - the first character is an uppercase letter
 - if it is, that it also includes at least one digit (0–9)
- if a password does not meet these requirements:
 - display the password
 - increment the number of incorrect passwords
- display the total number of incorrect passwords after all the passwords have been checked.

Open **Q05a** in the code editor.

Write the program.

You must use the structure given in **Q05a** to write the program.

Do not add any further functionality.

Save your code as **Q05aFINISHED** with the correct file extension for the programming language.

(9)



(b) **Figure 3** shows an array that stores player scores after a game.

3	2	10	8	1	9
---	---	----	---	---	---

Figure 3

(i) Julia uses a bubble sort algorithm to sort the scores.

Complete the table to show how the bubble sort algorithm will sort the scores.

You may not need to use all the rows.

(3)

3	2	10	8	1	9

(ii) Explain **one** reason why a bubble sort is very efficient in terms of memory usage.

(2)

(c) Describe the steps a linear search algorithm takes to find a search item.

(3)



2 A book club uses computer applications.

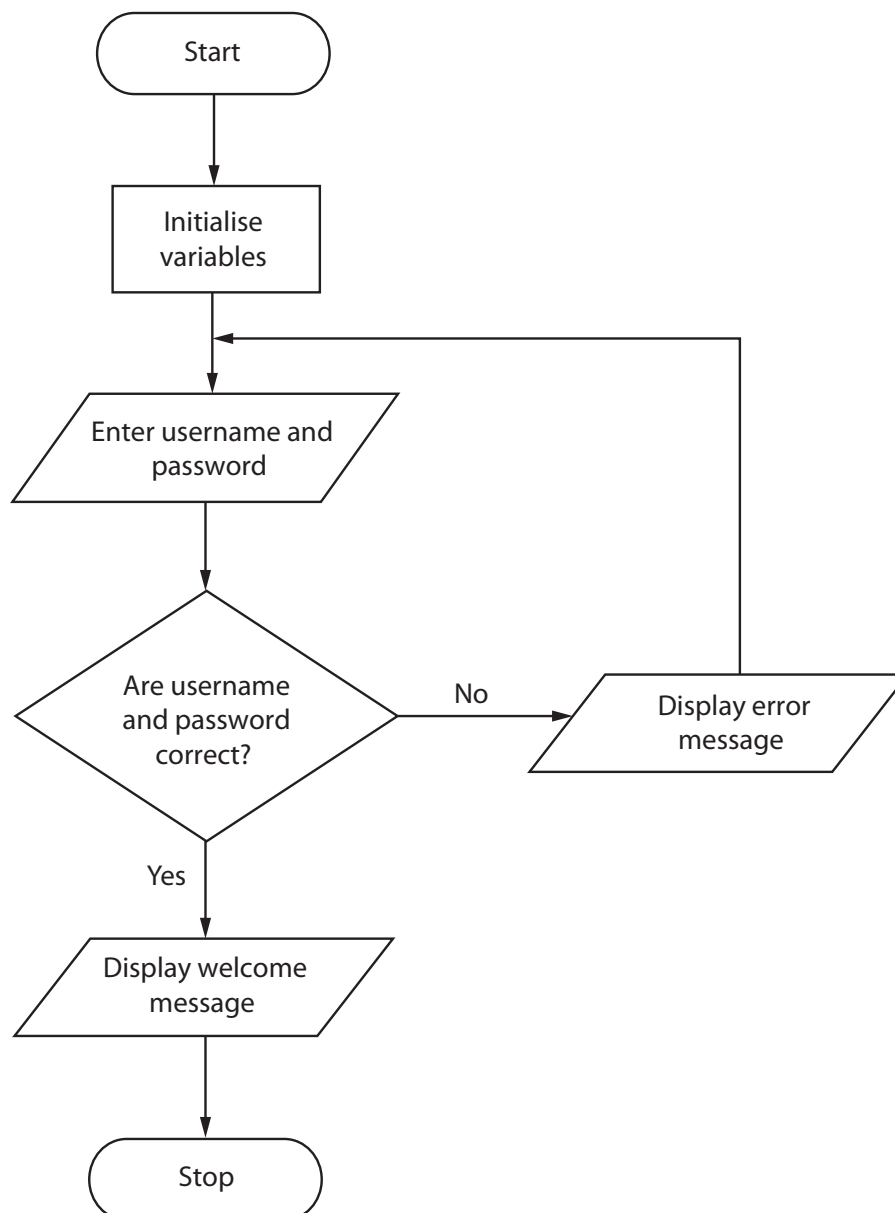
(a) The club wants a program to ensure that logins are valid.

A flowchart for a test version of the program has been designed.

The test version uses:

- a username of bard423
- a password of nX2934?

Here is the flowchart.



Write a program to implement the logic in the flowchart.

Open **Q02a** in the code editor.

You must use the structure given in **Q02a** to write the program.

Do not add any further functionality.

Save your code as **Q02aFINISHED** with the correct file extension for the programming language.

(11)

(b) The club sells books.

It needs a computer program to monitor:

- the number of books sold
- the amount of profit made.

Q02b provides a structure for the program.

Open **Q02b** in the code editor.

Amend the code to complete the If statement used to produce the outputs described in the table.

Condition	Output message
Number of books sold is at least 5 and profit made is at least 10	Sales and profit are good this week
Number of books sold is over 20 and profit made is at least 20	Sales and profit are excellent this week
Number of books sold is under 5 or profit made is under 5	Poor performance this week
All other inputs	Alert manager

Do not add any further functionality.

Save your code as **Q02bFINISHED** with the correct file extension for the programming language.

(4)



(c) Boundary testing is an important part of program development.

The boundary value is the most extreme value that will pass the test.

For example, if the boundary condition is no more than 30 then the boundary test data would be 30.

Complete the test table to show the boundary test data that could be used for each test.

(3)

	Test data	Expected output
booksSold		Poor performances this week
profit		
booksSold		Sales and profit are good this week
profit		
booksSold		Sales and profit are excellent this week
profit		

(Total for Question 2 = 18 marks)

2 Estelle is learning about sorting and searching algorithms.

(a) Estelle's friend has given her a partially completed program to carry out a binary search.

This pseudocode contains the logic required to complete the program.

```
11 RECEIVE item FROM (INTEGER) KEYBOARD
12 WHILE start <= end AND NOT found DO
13     SET middle TO (start + end) DIV 2
14     IF numberList[middle] = item THEN
15         SET found TO TRUE
16     ELSE
17         IF item < numberList[middle] THEN
18             end = middle - 1
19         ELSE
20             start = middle + 1
21         END IF
22     END IF
23     SET count TO count + 1
24 END WHILE
```

Open file **Q02a** in the code editor.

Amend the code to complete the program.

You **must** use the structure given in **Q02a** to write the program.

Do not add any further functionality.

Save your code as **Q02aFINISHED** with the correct file extension for the programming language.

(10)



(b) **Figure 3** shows an array of names.

0	1	2	3	4	5	6	7
Yekatrina	Juan	George	Elija	Christine	Sangeetha	Melanie	Shu-fen

Figure 3

Explain why a binary search algorithm would not find 'Juan' in the array.

(4)

(c) A merge sort algorithm can be used to sort a list into ascending order.

The list 9, 1, 7, 6, 3, 5, 2, 8 needs to be sorted.

Complete the merge sort using the space provided.

(5)

9, 1, 7, 6, 3, 5, 2, 8

1, 2, 3, 5, 6, 7, 8, 9

(Total for Question 2 = 19 marks)



5 Ria is a school librarian.

She wants a program to analyse pupil use of the library.

She wants to encourage reading by awarding gold, silver and bronze medals to the three pupils who have read the most books.

Test data has been included in the code.

Open **Q05** in the code editor.

Write a program to calculate and display:

- the total number and average number of books pupils have read
- the IDs of pupils who have read fewer than ten books
- the details of the gold, silver and bronze medal winners.

Your program should function correctly even if the number of pupils in the file is changed.

Save your code as **Q05FINISHED** with the correct file extension for the programming language.

(20)

You may use this space for planning / design work.



6 Farshia is the regional manager for an insurance company.

She manages a team of sales staff.

She wants a program to analyse the performance of her team over a number of months.

Test data has been included in the code.

Open **Q06** in the code editor.

Write a program to:

- calculate and display the total sales made by each member of the team
- calculate and display the total sales made by the whole team
- display the first name, last name and the total sales made by the two members of the team with the highest total sales. (Ignore the possibility of two or more members of the team having the same total sales.)

Your program should function correctly even if the number of months or number of members of the team is changed.

Save your code as **Q06FINISHED** with the correct file extension for the programming language.

(20)

You may use this space for planning / design work.



5 Bianca has started to write a program.

The program is a guessing game about countries and their capital cities.

Open **Q05** in the code editor.

The program already:

- displays this menu

```
Menu
-----
[1] Add player name
[2] Play guess the capital city
[3] End game
-----
Input your menu choice: |
```

- asks for the user's menu choice.

Write the program code for the menu choices.

Menu choice [1] Add player name

If a player chooses this option, then they must input a player name.

Menu choice [2] Play guess the capital city

If a player chooses this option, then:

- they must answer five questions
- they select a question to answer by choosing a valid question number
- each question can only be selected once (they must not be able to choose a number more than once)
- the question should be displayed
- they must input the name of the capital city
- if they guess correctly their score must be incremented by 1
- if they guess incorrectly the correct answer must be displayed.

Menu choice [3] End game

If the user chooses this option, then the player name and score should be displayed.

Your program should function correctly even if the number of countries and capital cities in the file is increased.

Save your code as **Q05FINISHED** with the correct file extension for the programming language.

(20)



You may use this space for planning / design work.

The content of this page will **not** be assessed.



5 Ann likes to create programs.

Open **Q05** in the code editor.

It contains an array of words.

Ann wants the program to:

- accept the input of a word, or the number 1 to exit. No validation of the input is required
- display all the words in the array that have the same first letter as the word that has been input
- calculate and display the number of words in the array that begin with the same letter as the word that has been input
- display all the words in the array that contain the word that has been input
- calculate and display
 - the number of words in the array that contain the word that has been input
 - the number of letters in the longest word that contains the word that has been input
 - the number of letters in the shortest word that contains the word that has been input
- display
 - the longest word that contains the word that has been input
 - the shortest word that contains the word that has been input
- repeat until the user inputs the number 1 to exit the program.

Figure 4 shows the display when the word **win** is input.

Enter a word or 1 to exit: win

wind
wire

2 word(s) begin with w

wind
borrowing
swing

3 word(s) contain win
The longest word has 9 letters
The shortest word has 4 letters
The longest word is borrowing
The shortest word is wind

Enter a word or 1 to exit:

Figure 4

Write the code for the program.

Your program should function correctly even if the number of words in the array is changed.

Save your code as **Q05FINISHED** with the correct file extension for the programming language.

The next page may be used for planning / design work.

(20)

You may use this space for planning / design work.

The content of this page will **not** be assessed.



6 Carlos wants you to create a **guess the animal** game.

Open **Q06** in the code editor.

The code contains an array of animals.

It also contains a function that randomly selects an animal from the array. This is the secret word the user needs to guess.

Carlos wants the program to:

- generate the number of attempts the user has to guess the secret word. The maximum number of attempts is the length of the secret word +3. For example, the user has 8 attempts to guess when the secret word is **tiger**
- keep track of letters from incorrect attempts that are in the secret word and those that are not. There should be no duplicated letters
- display a message telling the user:
 - the number of letters in the secret word
 - how many attempts they have left
- force the user to input a word that is the same length as the secret word
- check whether the input word matches the secret word:
 - if the words match then a message that includes the secret word and the number of attempts taken to guess it is displayed
 - if the words do not match then:
 - letters from the attempt that appear in the secret word should be added to the correct letters store
 - letters from the attempt that do not appear in the secret word should be added to the wrong letters store
 - the contents of the correct and wrong letter stores are displayed
- allow the user another attempt until they have guessed the word or have run out of attempts
- display a message telling the user the game is over including the random word if the maximum attempts have been taken and the word has not been guessed.



Figure 4 shows the contents of the correct and wrong stores after two attempts to guess the secret word **cow**.

Secret word cow			
First attempt		Second attempt	
input	dog	input	cat
correct store	o	correct store	o c
wrong store	d g	wrong store	d g a t

Figure 4

Your program should include at least two subprograms that you have written yourself.

You must include comments in the code to explain the logic of your solution.

Save your code as **Q06FINISHED** with the correct file extension for the programming language.

You may use this space for planning/design work.

The content of this space will **not** be assessed.

2 Anna has a 5-year-old daughter, Beatrice, and a 14-year-old son, Graham.

(a) Anna wants to program a guessing game for Beatrice to play on the computer.

The program will generate a random number between 1 and 10.

Beatrice has to guess the number.

This pseudocode contains the logic required to create the game.

```

1  #   Initialise variables
2      SET counter TO 1
3      SET answer TO RANDOM(9) + 1    # i.e. a random integer 1 to 10 inclusive
4      SET guess TO 0
5
6  #   Print prompt and take guess from user
7      SEND "Enter a number from 1 to 10: " TO DISPLAY
8      RECEIVE guess FROM (INTEGER) KEYBOARD
9
10 #   Create while loop to check guess
11     WHILE guess <> answer DO
12         SET counter TO counter + 1
13         IF guess > answer THEN
14             SEND (guess & " was too high. Try again.") TO DISPLAY
15         ELSE
16             SEND (guess & " was too low. Try again.") TO DISPLAY
17         END IF
18         SEND "Guess again: " TO DISPLAY
19         RECEIVE guess FROM (INTEGER) KEYBOARD
20     END WHILE
21
22 #   Report the correct answer to the user and display the number of guesses
23     SEND ("You guessed " & guess & " in " & counter & "guesses.") TO DISPLAY

```

Write a program to implement the logic in the pseudocode.

Open **Q02a** in the code editor.

You **must** use the structure give in **Q02a** to write the program.

Do not add any further functionality.

Save your code as **Q02aFINISHED** with the correct file extension for the programming language.

(10)



- (b) Anna has written a program to help Graham with calculations for his science lessons.

Open **Q02b** in the code editor.

Use the code to answer these questions.

- (i) State the value of **distance** when the program finishes.

(1)

- (ii) Give **one** reason why there are brackets around **newSpeed + initialSpeed**

(1)

- (iii) State the data type for the variable **distance**.

Justify your answer.

(2)

Data type

Justification

- (iv) State the value of **distance** when **acceleration** is 2.75

(1)

(c) Anna has written a program that manipulates text.

Open **Q02c** in the code editor.

Use the code to answer these questions.

(i) Identify a line number where the code includes a relational operator.

(1)

(ii) Identify a line number where a variable is set to a numeric value entered by a user.

(1)

(iii) Identify the name of a variable passed from the main program to the subprogram.

(1)

(iv) Identify the name of a variable that receives the value calculated by the subprogram.

(1)

(v) State the value returned by the subprogram when the input values are **educational, 3, 6**

(2)

(vi) State the purpose of the subprogram.

(2)

(Total for Question 2 = 23 marks)



MARK SCHEMES

official Pearson Edexcel mark schemes



Question	mp	Additional Guidance	Mark
3	Award 1 mark for each of:		
	A1	At least one variable with a meaningful name (1)	
	A2	At least one condition and price correct (ignore order) (1)	
	A3	All conditions and prices correct (1)	
	A4	Calculation for Total cost is correct (1)	
	A5	Price per textbook and total cost displayed (1)	
	A6	Calculation of cost and display of output are not included in the if statement (1)	
Code examples			
Python	<pre>1 # Q03 2 3 # Initialise variables 4 quantity = 0 5 price = 0 6 cost = 0 7 8 # Print prompt and take number of textbooks required 9 quantity = int(input("Please enter the quantity of books required: ")) 10 11 # Generate and display the price per textbook and the total cost 12 if quantity >= 1 and quantity <= 5: 13 price = 20 14 elif quantity < 10: 15 price = 15 16 else: 17 price = 12 18 19 cost = price * quantity 20 print("The price per book is",price) 21 print("The total cost is",cost)</pre>		(6)

Question	mp	Answer	Additional Guidance	Mark
1 (a)	A1	C (equals) (1)		(1)

Question	mp	Answer	Additional Guidance	Mark
1 (b)	B1	Award one mark for any of: - TRUE and FALSE (1) - Yes and No (1) - T and F (1) 0 and 1 (1)	Ignore capitalisation	(1)

Question	mp	Answer	Additional Guidance	Mark
1 (d)	D1	Award one mark for any of: (Symbolic name associated with) a value that may be changed (1) - A container used to store data (1) - A data store whose contents can change whilst a program is executing (1)		(1)

Question	mp	Answer	Additional Guidance	Mark
1 (e) (i)	E1	Python 6 C# 12 Java 8		(1)
1 (e) (ii)	E2	Python 3 C# 7 Java 3		(1)
1 (e) (iii)	E3	Award one mark for any of: - amount (1) - x (1) - total (1) - count (1)	Accept listNumbers or numArray	(1)
1 (e) (iv)	E4	message		(1)

Question	mp	ref	Answer	Additional Guidance	Mark
1 (c)	C1	2.1.5	Delete space in variable name (1)		(3)
	C2	2.1.5	Capitalise O in variable name (1)		
	C3	2.1.5	Correction to print statement (1)		
Code example					
Python	<pre>1 numOne = 25 2 numTwo = 36 3 numThree = numOne * numTwo 4 print (numThree)</pre>				

Question	mp	Answer	Additional Guidance	Mark
1 (a)	A1	1. The only correct answer is B A is not correct because as it is an arithmetic operator C is not correct because as it is a relational operator D is not correct because as it is a relational operator		(1)

Question	mp	Answer	Additional Guidance	Mark
1 (b)	B1 B2	Award up to 2 marks for a linked description such as: 1D represents items as a list (1), 2D as a table / 2D as rows and columns (1) 1D is a row/column (1), 2D is a table / 2D has rows and columns (1) - Each element in 1D is a single value (1), each element in 2D is a 1D array (1) 1D array can only store one type of element (1). 2D array can store multiple elements in it (1)	Accept an interpretation	(2)

Question	mp	Answer	Additional Guidance	Mark
1 (c)	C1	Change num_twenties == to num_twenties = (1)		(3)
	C2	The left over variable named the same in both places (1)		
	C3	Add + before str(left_over) over in final print statement (1)	Accept equivalent e.g. ,	

Question	mp	Answer	Additional Guidance	Mark
1 (d) (i)	D1	Award 1 mark for adding an appropriate comment at the end of the line where there is relational operator: <pre> 10 if vowel == letter: # relational operator and selection </pre>	May be on different line numbers	(1)
1 (d) (ii)	D2	Award one mark for adding an appropriate comment at the end of a line where iteration starts: <pre> 8 for letter in sentence: # iteration starts 9 for vowel in vowels: # iteration starts 13 print("here are the number of vowels in the s") 14 for vowel in vowels: # iteration starts </pre>	The comment(s) added must clearly identify the component	(1)
1 (d) (iii)	D3	Award one mark for adding an appropriate comment at the end of the line where selection starts: <pre> 10 if vowel == letter: # relational operator and selection </pre>		(1)
1 (d) (iv)	D4	Award one mark for adding an appropriate comment at the end of a line where a data structure is initialised: <pre> 3 vowels = ["a","e","i","o","u"] # data structure initialised 4 numVowels = [0,0,0,0,0] # data structure initialised </pre>		(1)



Question	mp	ref	Answer	Additional Guidance	Mark								
5 (a)	A1 A2 A3 A4	2.1.4	Award 1 mark for each appropriate validation test (up to 2) and 1 mark for suitable example of erroneous data that matches the test e.g. <table><tr><td>First 3 characters are upper case letters</td><td>Fff123456o</td></tr><tr><td>Characters 4 to 9 are non-zero numbers</td><td>FFF101111o</td></tr><tr><td>Consists of only 10 characters</td><td>FFF111111ee</td></tr><tr><td>Final character correct for sum of numbers</td><td>FFF112233o</td></tr></table>	First 3 characters are upper case letters	Fff123456o	Characters 4 to 9 are non-zero numbers	FFF101111o	Consists of only 10 characters	FFF111111ee	Final character correct for sum of numbers	FFF112233o		(4)
First 3 characters are upper case letters	Fff123456o												
Characters 4 to 9 are non-zero numbers	FFF101111o												
Consists of only 10 characters	FFF111111ee												
Final character correct for sum of numbers	FFF112233o												

Question	mp	ref	Answer	Additional Guidance	Mark
5 (b) (i)	B1 B2	1.1.9	Award 1 mark each for any of: - Simple implementation (1) - Can be used for sorted or unsorted lists (1) - If the target is at the beginning of the data structure the search will be faster than a binary search	Accept disadvantages of a binary compared with a linear search.	(2)
5 (b) (ii)	B3 B4	1.1.9	Award 1 mark each for any of: - May need to compare with all items in list before search complete (1) - May need longer time for searching a large list (1)	Accept advantages of a binary compared with a linear search.	(2)

Question	mp	ref	Answer	Additional Guidance	Mark
5 (c) (i)	C1 C2	2.6.2	Award 1 mark each up to a maximum of 2 for: • Both can use arguments / parameter passing (1) • Both can use local variables (1) • Both can be called from anywhere within the program (1) • Code can be reused without being rewritten (1) • Both can be library files (1) • The code can be independently tested for both (1)		(2)
5 (c) (ii)	C3 C4	2.6.2	Award two marks for a linked explanation such as: • A function must always return a result (1) whereas a procedure does not (1) • A function interface must have a data type (1) to signify the type of data that will be returned / a procedure does not need this (1) • The result of a function must always be used (either assigned to a variable or as part of a condition) (1) whereas a procedure does not explicitly return a result to be used (1) • A function produces information (1) whereas a procedure performs a task (1)		(2)

Question	mp	ref	Answer	Additional Guidance	Mark
4 (a)	A1	2.5.2	english < 40, maths < 50 (1)		(6)
	A2	2.5.3	Use of and operator in IF (1)		
	A3	2.5.2	english < 40, maths < 50 (1)		
	A4	2.5.3	Use of or operator in 'else if 1' (1)		
	A5	2.5.2	english >= 80 AND maths >= 85 (1)		
	A6	2.5.3	Correct completion of final else with messageIndex = 3 (1)		

Code example

Python

```

1110, 1100, 1000, 900, 800, 700, 600, 500, 400, 300, 200, 100, 0)
if (english < 40 and maths < 50):
    messageIndex = 0
elif (english < 40 or maths < 50):
    messageIndex = 1
elif (english >= 80 and maths >= 85):
    messageIndex = 2
else:
    messageIndex = 3
print(message[messageIndex])

```

Question	mp	ref	Answer	Additional Guidance	Mark
4 (b)	B1	2.2.2	Loop checks each pupil for low attendance		(7)
	B2	2.1.6	Display of name of low attendances		
	B3	2.3.1	Counter for high attenders initialised		
	B4	2.2.2	Loop uses correct comparison (>=)		
	B5	2.6.2	Number of high attenders displayed correctly		
	B6	2.6.3	Subprogram for option 1 called correctly		
	B7	2.6.3	Subprogram for option 2 called correctly		

Code example

Python

```

3 # write subprogram for menu option 1 here
4 def lowAttendance():
5     for x in range(len(pupilAttendance)):
6         if pupilAttendance[x-1][2] < 75:
7             print (pupilAttendance[x-1][0], pupilAttendance[x-1][1])
8     return
9
10 # write subprogram for menu option 2 here
11
12 def highAttendance():
13     count = 0
14     for x in range(len(pupilAttendance)):
15         if pupilAttendance[x-1][2] >= 90:
16             count += 1
17     return count
18

```

Faroukh Salah
Amara Grzinski
Taz Grimstow
Sadia Bhatti
Fernado Askabat
Siyao Wang

```

45 if option == 1:
46     # complete if statement
47     lowAttendance()
48
49 elif option == 2:
50     # complete else if statement
51     highAtt = highAttendance()
52     print ("There were ", highAtt, " high attenders")
53

```

There were 7 high attenders



Question	mp	Answer	Additional Guidance	Mark
1 (a)	A1	Award 1 mark for any of: - Comments (1) - Indentation (1) - Meaningful variable/constant/subprogram names (1) - White space (1)		(1)

Question	mp	Answer	Additional Guidance	Mark																
1 (b)	B1	Award 1 mark for each column: <table><thead><tr><th>Error description</th><th>Logic</th><th>Syntax</th><th>Runtime</th></tr></thead><tbody><tr><td>Divide by 0</td><td></td><td></td><td>✓</td></tr><tr><td>Use x instead of * multiply</td><td></td><td>✓</td><td></td></tr><tr><td>Subtract 10 from 2 instead of 2 from 10</td><td>✓</td><td></td><td></td></tr></tbody></table>	Error description	Logic	Syntax	Runtime	Divide by 0			✓	Use x instead of * multiply		✓		Subtract 10 from 2 instead of 2 from 10	✓				(3)
	Error description		Logic	Syntax	Runtime															
	Divide by 0				✓															
	Use x instead of * multiply			✓																
	Subtract 10 from 2 instead of 2 from 10		✓																	
B2																				
B3																				

Question	mp	Answer	Additional Guidance	Mark
1 (c)	C1	The only correct answer is D <i>A is not correct because as it is a data structure</i> <i>B is not correct because as there is no arrow looping back to the condition</i> <i>C is not correct because as the flowchart does not include an operator</i>		(1)

Question	mp	Answer	Additional Guidance	Mark
1 (d)	D1 D2 D3	Award 1 mark for each of: D1 = changed to == (1) D2 Condition changed to less than / final two print statements swapped (1) D3 else changed to else (1)		(3)

Code examples

C#	<pre> if (number1 == number2) { Console.WriteLine("The numbers are equal"); } else if (number1 < number2) { Console.WriteLine("The highest number is " + number2 + " and the lowest number is " + number1); } else { Console.WriteLine("The lowest number is " + number2 + " and the highest number is " + number1); } </pre>	
Java	<pre> if (number1 == number2) { System.out.print("Numbers are equal"); } else if (number1 < number2) { System.out.print("The highest number is " + number2 + " and the lowest number is " + number1); } else { System.out.print("The lowest number is " + number2 + " and the highest number is " + number1); } </pre>	
Python	<pre> if number1 == number2: print("Numbers are equal") elif number1 < number2: print("The highest number is", number2, "and the lowest number is", number1) else: print("The lowest number is", number2, "and the highest number is", number1) </pre>	



Question	mp	Answer	Additional Guidance	Mark
1 (e)		Award 3 marks for:		(5)
	E1	Prompt for a number to be input (1)		
	E2	Input stored in an integer variable (1)		
	E3	Loop used (1)		
	E4	Loop iterates only for the values 1 – 12 (1)		
	E5	Display is correct when program executes (1)		
Code examples				
C#		<pre>// Print prompt and get number from user Console.Write("Enter the number: "); int num = Convert.ToInt32(Console.ReadLine()); // Create loop to display the table for (int count = 1; count <= 12; count++) { Console.WriteLine(count + " x " + num + " is " + count * num); } Console.ReadLine();</pre>		
Java		<pre>// Print prompt and get number from user Scanner input = new Scanner(System.in); System.out.print("Enter the number "); int num = input.nextInt(); // Create loop to display the table for (int count = 1; count <=12; count++) { System.out.println(count + " x " + num + " is " + count * num); }</pre>		
Python		<pre># Print prompt and get number from user num = int(input("Enter the number: ")) # Create loop to display the table for count in range(1, 13): print(count, "x" , num , "is" , count * num)</pre>		



Question	mp	Answer	Additional Guidance	Mark
4 (b)	B1 B2	Award up to 2 marks for a linked description such as: To check to see if a word is the same when it is reversed / is a palindrome (1) and output an appropriate message / repaper is the same when reversed (1)		(2)

Question	mp	Answer	Additional Guidance	Mark
4 (a)		Award one mark for each of:		
		Validation		
	A1	Suitable input prompt and storing it in binaryPattern (1)		
	A2	Length of binary pattern validated as 8 (1)		
	A3	Loop used to check the length of binary pattern (1)		
		Binary to denary		
	A4	Loop through each digit in the binary pattern (1)		
	A5	Correct conversion of at least one binary digit to relevant placeholder (1)		
	A6	Attempt at running total using denaryNumber (1)		
	A7	Correct running total using denaryNumber (1)		
	A8	Print statement includes the binary pattern and the denary number (1)		
	A9	Print statement outside of conversion loop (1)		
	A10	Comment explains how conversion works (1)		
	A11	Program executes correctly for any binary pattern that is exactly 8 characters long only (1)		(11)

Code examples

C#	<pre> while (binaryPattern.Length != 8) { Console.WriteLine("Enter an 8 digit binary number to convert to denary "); binaryPattern = Console.ReadLine(); } foreach (char digit in binaryPattern) { if (digit == '1') { denaryNumber += denaryPlaceholders[count] * 1; } count++; } Console.WriteLine(binaryPattern + " converted to denary is " + denaryNumber); Console.ReadLine(); </pre>	
Java	<pre> while (binaryPattern.length() != 8) { Scanner input = new Scanner(System.in); System.out.print("Enter an 8 digit binary number to convert to denary "); binaryPattern = input.next(); } for (char digit : binaryPattern.toCharArray()) { if (digit == '1') { denaryNumber += denaryPlaceholders[count] * 1; } count ++; } System.out.print(binaryPattern + " converted to denary is " + denaryNumber); </pre>	

Python	<pre> while len(binaryPattern) != 8: binaryPattern = input("Enter an 8 digit binary number to convert to denary: ") for digit in binaryPattern: if digit == "1": denaryNumber = denaryNumber + denaryPlaceholders[count] * 1 count = count + 1 print(binaryPattern,"converted to denary is", denaryNumber) </pre>	
--------	---	--



Question	Answer	Additional Guidance	Mark
Q3(b)	More memory/storage required (1) Because not sorted in place / not efficient / many steps / division of data set / is recursive (1)		(2)

Question	Answer	Additional Guidance	Mark
Q3(a)	Award 1 mark each for: 1. File opened for input (1) 2. File opened for output (1) 3. Loop for processing data (1) 4. Check for '@' symbol (1) 5. Write invalid address to output file (1) 6. Close files (1)	Amended code should be saved as Q03aFINISHED. Do not penalise candidates who attempt more than the stated requirements.	(6)

Question	Answer	Additional Guidance	Mark
Q3(c)	Award 1 mark for each section of code that: 1. Asks for input of a number and stores it in a variable with a meaningful name (1) 2. Checks number is between 1 and 50 (1) 3. Calculates square and cube of number (1) 4. Displays suitable labels e.g. 'Number', 'Square' and 'Cube' and relevant values (1) For the whole code: 5. Compiles and runs without logical or syntax errors (1) 6. Efficient in terms of computation, storage and selection of programming constructs (1)	Amended code should be saved as Q03cFINISHED. Allow follow through for mark point 4 when square and cube are incorrectly calculated. Credit alternative coding which produces same results. Accept use of exponent operators: C# e.g. Math.Pow (2, 3) = 8 Java e.g. Math.pow (2, 3) = 8 Python e.g. 2**3 = 8.	(6)

3(a)	<pre> 1 Q03aFINISHED 2 3 # Open file and input data 4 with open('Email.txt','r') as inputFile: 5 emailList = inputFile.readlines() 6 7 # Open file for output 8 outputFile = open('Error.txt', 'a') 9 10 # Find errors and write to error log 11 for address in emailList: 12 if not "@" in address: 13 outputFile.write(address) 14 15 # Close files 16 outputFile.close() 17 inputFile.close() 18 </pre>	
------	--	--

3(c)	<pre> 1 # Q03c 2 3 # Write your code below this line 4 myValue = 1 5 6 while (myValue >= 1 and myValue <= 50): 7 myValue = int(input("Enter a value from 1 to 50: ")) 8 if (myValue >= 1 and myValue <= 50): 9 print("Value: ", myValue, "Square: ",myValue*myValue, "Cube: ", myValue*myValue* 10 myValue) 11 print("Value: ", myValue, "Square: ", myValue**2, "Cube: ", myValue**3) </pre> Note: Line 11 is an alternative response to line 9.	
------	---	--

Question	mp	Answer	Additional Guidance	Mark
1 (a)	A1	The only correct answer is D A is not correct because the value can be used more than once B is not correct because the value does not always have to be input C is not correct because the value does not always have to be used in a calculation		(1)

Question	mp	Answer	Additional Guidance	Mark
1 (b)	B1	The only correct answer is B A is not correct because this would not aid readability of the code C is not correct because this would not aid readability of the code D is not correct because this would not aid readability of the code		(1)

Question	mp	Answer	Additional Guidance	Mark						
1 (c)	C1	Award 1 mark for each of:	Accept quotes around the character							
	C2									
		<table><tr><th>Data type</th><th>Example value</th></tr><tr><td>char</td><td>An example of any single letter, number or symbol</td></tr><tr><td>real</td><td>An example of a number with a decimal point</td></tr></table>	Data type	Example value	char	An example of any single letter, number or symbol	real	An example of a number with a decimal point	Real number does not need a number after the decimal point to be awarded the mark	(2)
Data type	Example value									
char	An example of any single letter, number or symbol									
real	An example of a number with a decimal point									

Question	mp	Answer	Additional Guidance	Mark
1 (d)	D1	Award up to 2 marks for a description such as: Boundary testing uses the most extreme valid data / minimum and maximum valid values (1) whereas erroneous testing uses invalid data (1)		(2)

Question	mp	Answer	Additional Guidance	Mark
1 (e)	E1	Logic/logical		(1)

Question	mp	Answer	Additional Guidance	Mark
1 (f)		Award 1 mark for each of: F1 Bracket added and no other amendments made to the line (1) F2 == becomes = (1) F3 " moved to after is and before , OR , area changed to +str(area) (1) Must still include the print statement		(3)

Code examples

```

Python
3 length = float(input("Enter the first number "))
4
5 area = length * length
6
7 print("The area of the square is ", area)

```

Question	mp	Answer	Additional Guidance	Mark
1 (g)		Award 1 mark for each of: G1 At least 1 condition correct (1) G2 At least 1 > or < condition and message match (1) G3 At least 1 output includes letter and stored letter with an appropriate message (1) G4 All conditions and outputs correct (1)		(4)

Code examples

```

Python
# Amend the code by completing the if statement
if letter > storedLetter:
    print(letter + " is later in the alphabet than " + storedLetter)
elif letter == storedLetter:
    print(letter + " is the same as " + storedLetter)
else:
    print(letter + " is earlier in the alphabet than " + storedLetter)

```



Question	mp	Answer	Additional Guidance	Mark																											
4 (b)(i)	B1	Award up to 2 marks for a linked explanation: • binary search can be quicker / is more effective than a linear search / is more effective with larger lists (1) as it does not have to examine each item in the list (1) • binary search halves the list each time (1) so it can be faster to find an item (1) • binary search requires fewer comparisons than a linear search to establish an item is not in the list (1) because the linear list would need to compare each item before establishing this (1)		(2)																											
4 (b)(ii)		Correct answer <table><tr><th>Position in list</th><th>Product code</th><th>Order examined</th></tr><tr><td>1</td><td>ark11</td><td></td></tr><tr><td>2</td><td>asp11</td><td></td></tr><tr><td>3</td><td>bar13</td><td></td></tr><tr><td>4</td><td>dri15</td><td>1</td></tr><tr><td>5</td><td>mil19</td><td></td></tr><tr><td>6</td><td>rib10</td><td>2</td></tr><tr><td>7</td><td>str15</td><td>3</td></tr><tr><td>8</td><td>tor16</td><td></td></tr></table> Award one mark for each correct value in order column	Position in list	Product code	Order examined	1	ark11		2	asp11		3	bar13		4	dri15	1	5	mil19		6	rib10	2	7	str15	3	8	tor16			(4)
Position in list	Product code	Order examined																													
1	ark11																														
2	asp11																														
3	bar13																														
4	dri15	1																													
5	mil19																														
6	rib10	2																													
7	str15	3																													
8	tor16																														
	B1	Start of search correct	Accept 5 and 7 for B1 and B2 (2 marks)																												
	B2	Second search item correct	Follow through if start of search incorrect																												
	B3	Third search item correct	Follow through if start of search incorrect																												
	B4	All correct																													
4 (b)(iii)	B5	Award 1 mark for: 3 or $\log_2 n + 1$		(1)																											
4 (b)(iv)	B6	Award 1 mark for any of: • bubble sort • merge sort	Accept any known sorting algorithm	(1)																											

Question	mp	Answer	Additional Guidance	Mark
4 (a)	A1	At least 1 variable has a meaningful name	Ignore spelling mistakes in input message	(7)
	A2	Product name requested using a suitable input message		
	A3	Random number generated that would be at least 10 or no higher than 30		
	A4	Random number generated that would be in the correct range 10 to 30		
	A5	First 3 letters of product name generated		
	A6	First 3 letters of product name and random number concatenated to generate productCode		
	A7	productCode and productName output in the same print statement		

Code examples

Python

```

# Get input
productName = input("Enter the product name: ")

randomNum = 0

# Generate a random number between 10 and 30 inclusive
randomNum = random.randint(10,30)

# Generate the product code = first three letters of product name + random number
productCode = productName[0:3] + str(randomNum)

# Display the product code and the product name
print(productCode + " " + productName)

```

Question	mp	Answer	Additional Guidance	Mark						
3 (a) (i)	A1	C# and Python monthNumber/monthsAndDays/count/found (1) Java monthNumber/monthsAndDays/count/found/input		(1)						
3 (a) (ii)	A2	monthsAndDays (1)		(1)						
3 (a) (iii)	A3	C# - WriteLine, ToString, ToInt32, ReadLine Java - Print, ParseInt, nextLine, ToString, Equals Python - print/len/int/input (1)		(1)						
3 (a) (iv)	A4	Display/showMonthNameAndDays (1)		(1)						
3 (a) (v)	A5	pMonth (1)		(1)						
3 (a) (vi)	A6 A7	Award 1 mark for each of: <table><tr><th>Type of test data</th><th>Test data</th></tr><tr><td>Boundary (1)</td><td>1</td></tr><tr><td>Erroneous (1)</td><td>15</td></tr></table>	Type of test data	Test data	Boundary (1)	1	Erroneous (1)	15		(2)
Type of test data	Test data									
Boundary (1)	1									
Erroneous (1)	15									

Question	mp	Answer	Additional Guidance	Mark
3 (b)	B1 B2	Award up to 2 marks for a linked explanation such as: - A library program would not know the names of the calling program's variables in advance (1) meaning it would not be a reusable solution (1) (Using parameters) enables reusable solutions (1) using different data values/arguments (1) - Using parameters uses less memory / uses memory more efficiently (1) because it avoids the use of global variables / because the memory is freed after the subprogram is finished (1)		(2)

Question	mp	Answer	Additional Guidance	Mark
3 (c)	C1 C2 C3 C4 C5 C6	Award 1 mark for each of: - Condition to check whether the year exactly divides by 4 or equivalent condition (1) - Condition to check whether year exactly divides by 400 or equivalent condition (1) - Condition to check whether year does not exactly divide by 100 or equivalent condition (1) - Correct message for at least one of the conditions (1) - Program executes and produces the correct output (1) - Efficient solution (1)		(6)

Code examples

C#	<pre> if (year % 400 == 0) { Console.WriteLine(year + " is a leap year"); } else if (year % 4 == 0 && year % 100 != 0) { Console.WriteLine(year + " is a leap year"); } else { Console.WriteLine(year + " is not a leap year"); } // End of main program </pre>	
Java	<pre> if (year % 400 == 0) { System.out.print(year + " is a leap year"); } else if (year % 4 == 0 && year % 100 != 0) { System.out.print(year + " is a leap year"); } else { System.out.print(year + " is not a leap year"); } </pre>	
Python	<pre> if year % 400 == 0: print(year,"is a leap year") elif year % 4 == 0 and year % 100 !=0: print(year,"is a leap year") else: print(year,"is not a leap year") </pre>	



Question	Answer	Additional Guidance	Mark
1(a)	The only correct answer is C A is not correct because Boolean only has values True and False B is not correct because it only allows for one character D is not correct because it is used for mixed text and numeric data		(1)
1(b)	(an error that) produces incorrect /unintended output (1)	Accept an example of an explained logic error	(1)
1(d)	Any one from: 1. explain how the program works (1) 2. improve readability of the program (1) 3. make code more understandable (1)		(1)
1(e)	1. Global variables created in the main program / can be accessed from any part of the program / are in scope throughout the program (1) 2. Local variables are declared/created inside a function/blocked code (indented, brackets) and are only accessible there / are only in scope within the function in which they're declared (1)		(2)

Question	Answer	Additional Guidance	Mark								
1(g)	For up to two tests Award 1 mark for an appropriate validation test and 1 mark for an item of test data that would fail only the given test <table><tr><th>Validation test</th><th>Erroneous data</th></tr><tr><td>Are the first two characters letters/upper case letters?</td><td>1234CCC</td></tr><tr><td>Are the third- and fourth- characters numerical digits?</td><td>CCCCBBB</td></tr><tr><td>Are the last three characters letters/upper case letters?</td><td>CC12345</td></tr></table>	Validation test	Erroneous data	Are the first two characters letters/upper case letters?	1234CCC	Are the third- and fourth- characters numerical digits?	CCCCBBB	Are the last three characters letters/upper case letters?	CC12345	Each test should target a different feature of one of the requirements • Exactly 7 characters (in the stem) • First two characters are letters / upper case letters • Characters 3 & 4 are numeric • Last three characters are letters / upper case letters	(4)
Validation test	Erroneous data										
Are the first two characters letters/upper case letters?	1234CCC										
Are the third- and fourth- characters numerical digits?	CCCCBBB										
Are the last three characters letters/upper case letters?	CC12345										

Question	Answer	Additional Guidance	Mark
1(c)	C# 1. Semicolon at the end of line 9 [maxValue=5;] (1) 2. Consistent capitalisation of 'count' on lines 10, 11, 13 and 14 (1) 3. constantValue needs to be defined and given the value 7 / replaced by value 7 (1) Java 1. Semicolon at the end of line 8 [maxValue = 5;] (1) 2. Consistent capitalisation of 'count' on lines 7, 9, 11 and 12 (1) 3. constantValue needs to be defined and given the value 7 / replaced by 7 (1) Python 1. Colon needed at end of line 7 [count, maxValue:;] (1) 2. Consistent capitalisation of 'count' on line 3, 7, 8 and 9 (1) 3. constantValue needs to be defined and given the value 7 / replaced by 7 (1)	Candidates are required to open the file Q01c in the code editor. Amended code should be saved as Q01cFINISHED. Line numbers may have been changed by corrections to code (e.g. insertion of constantValue definition)	(3)

Question	Answer	Additional Guidance	Mark						
1(f)	Award 1 mark for each correct pair of relational operators and 1 mark for correct Boolean operator for each condition. <table><tr><th>Condition</th><th>Coding</th></tr><tr><td>Shop income more than £5000 or assistant sold at least 10 pairs</td><td>shopIncome > 5000; assistantSales >= 10 (1) correct OR operator (1)</td></tr><tr><td>Shop income at least £2000 and assistant sold at least 5 pairs</td><td>shopIncome >= 2000; assistantSales >= 5 (1) correct AND operator (1)</td></tr></table> Accept >9, >1999 and >4 for respective >= comparisons	Condition	Coding	Shop income more than £5000 or assistant sold at least 10 pairs	shopIncome > 5000; assistantSales >= 10 (1) correct OR operator (1)	Shop income at least £2000 and assistant sold at least 5 pairs	shopIncome >= 2000; assistantSales >= 5 (1) correct AND operator (1)	Candidates are required to open the file Q01f in the code editor. Amended code should be saved as Q01fFINISHED. Do not penalise candidates who attempt more than the stated requirements. Do not award Boolean operator mark for single or & as these are not correct.	(4)
Condition	Coding								
Shop income more than £5000 or assistant sold at least 10 pairs	shopIncome > 5000; assistantSales >= 10 (1) correct OR operator (1)								
Shop income at least £2000 and assistant sold at least 5 pairs	shopIncome >= 2000; assistantSales >= 5 (1) correct AND operator (1)								

1(c)

```
1  # Q01c
2
3  count = 0
4
5  maxValue = 5
6  constantValue = 7
7
8  while (count < maxValue):
9      print(count, count + constantValue)
10     count = count + 1
11
```

1(f)

```
11  if (shopIncome > 5000) or (assistantSales >= 10):
12      print ("Assistant bonus = ", assistantSalary * 0.1)
13
14  elif (shopIncome >= 2000) and (assistantSales >= 5):
15      print("Assistant bonus = ", assistantSalary * 0.05)
```

Question	mp		Additional Guidance	Mark
2 (a)(i)	A1	3/8/20		(1)
2 (a)(ii)	A2	7		(1)
2 (a) (iii)	A3	21/23		(1)
2 (a)(iv)	A4	check/count/pNumber		(1)
2 (a)(v)	A5	pNumber		(1)

Question	mp		Additional Guidance	Mark
2 (b)	Award 1 mark for each of: Evidence should be found in the Function B1 Function interface is correct (1) B2 Condition correct (for the number 1) (1) B3 Loop initialised correctly (from 2 to the input number) (1) B4 If condition correct (modulus) (1) B5 Check returned (1) Evidence should be found in the Main program B6 Input of number stored as an integer variable (1) B7 Input includes a space/colon / new line before the input value B8 Function called correctly (1) B9 Display includes number and message for either a prime number OR not (1) Overall B10 Compiling without syntax errors (1) B11 Executing and producing the correct output (1)		Logic of algorithm must be followed as set out. Alternatives must address each point. Do not penalise candidates who attempt more than the stated requirements. Do not penalise spelling mistakes in the input message.	(11)

Code examples

Python

```

# Write the function here
def checkPrime(pNumber):
    if pNumber == 1:
        check = False
    else:
        check = True
        for count in range(2,pNumber):
            if pNumber % count == 0 :
                check = False
        return check

# Write the main program here
number = int(input("Enter a number: "))
result = checkPrime(number)
if result == True:
    print(number,"is a prime number")
else:
    print(number,"is not a prime number")

```



Question	mp	ref	Answer	Additional Guidance	Mark
2 (a)	A1	2.3.4	Set variable base to 50 or heightChk to true	Logic of algorithm must be followed as set out. Alternatives must address each point. Do not penalise candidates who attempt more than the stated requirements. Don't penalise spelling mistakes and alternative wording of the output.	(10)
	A2	2.2.2	Create WHILE DO loop		
	A3	2.4.1	Request input of height		
	A4	2.2.2	if statement checks if height ≥ 1 or ≤ 100		
	A5	2.2.2	if statement checks if height is ≥ 1 and ≤ 100		
	A6	2.2.2	heightCheck set to false if condition is met		
	A7	2.5.1	Set area using given formula ($0.5 \times \text{base} \times \text{height}$)		
	A8	2.4.1	Display one of base, height and area with appropriate label or displays two of base, height and area without labels		
	A9	2.4.1	Display all of base, height and area with appropriate labels		
	A10	1.1.6	Executing and producing correct output		

Code example

Python

```

1  # Q02aFINISHED
2
3  # Initialise variables
4
5  base = 50
6
7  # Print prompt and take input from user
8
9  heightChk = TRUE
10 while heightChk:
11     height = int(input("Enter the height (between 1 and 100): "))
12     if (height >= 1 and height <= 100):
13         heightChk = FALSE
14
15 # Calculate area and print out values
16
17 area = 0.5 * base * height
18
19 print ("Base length of triangle is : ", base)
20 print ("Height of triangle is : ", height)
21 print ("Area of triangle is : ", area)
22

```

Question	mp	ref	Answer	Additional Guidance	Mark
2 (b)	B1	2.2.1	Meaningful variable names used (1)	The code example shown is one way of responding to the task. Other methods should be credited accordingly.	(7)
	B2	2.4.1	Meaningful prompts for input (1)		
	B3	2.4.1	Input of length and width as whole numbers (1)		
	B4	2.5.1	Perimeter calculated ($= 2 \times \text{length} + 2 \times \text{width}$) (1)		
	B5	2.5.1	Gap deducted from perimeter (1)		
	B6	2.4.1	Meaningful output (1)		
	B7	2.1.1	Code is fit for purpose. Must have mark points 1-6 and be easy to read (1)		

Code example

Python

```

1  # Q02bFINISHED
2
3  # Request input
4
5  length = int(input("Enter the length: "))
6  width = int(input("Enter the width: "))
7
8  # Calculate number of panels needed
9
10 panels = 2 * length + 2 * width - 4
11
12 # Print out number of panels needed
13
14 print ("Number of panels needed: " , panels)

```

Question	mp	Answer	Additional Guidance	Mark
3 (b)	B1	Award up to 2 marks for a linked explanation such as: - The number of keys are limited/only one shift used (1) making it easy to use brute force to decrypt (1) - It can be easy to find commonly used letters (e.g. E) (1) and guess the key (1)	Accept alternative similar wording.	(2)

Question	mp	Answer	Additional Guidance	Mark																																	
3 (c)		Award 1 mark each up to a maximum of 4 for: <table><tr><td>Encrypted letter</td><td>f</td><td>l</td><td>m</td><td>k</td><td>t</td><td>r</td><td>w</td><td>h</td><td>e</td><td>e</td></tr><tr><td>Keyword letter</td><td>t</td><td>h</td><td>i</td><td>r</td><td>t</td><td>y</td><td>t</td><td>h</td><td>i</td><td>r</td></tr><tr><td>Decrypted letter</td><td>m</td><td>e</td><td>e</td><td>t</td><td>a</td><td>t</td><td>d</td><td>a</td><td>w</td><td>n</td></tr></table>	Encrypted letter	f	l	m	k	t	r	w	h	e	e	Keyword letter	t	h	i	r	t	y	t	h	i	r	Decrypted letter	m	e	e	t	a	t	d	a	w	n		(4)
	Encrypted letter	f	l	m	k	t	r	w	h	e	e																										
	Keyword letter	t	h	i	r	t	y	t	h	i	r																										
	Decrypted letter	m	e	e	t	a	t	d	a	w	n																										
	C1	Award 1 mark from: - flmktrwhee mapped to thirtythir (1) - Flmktr mapped to thirty (1)																																			
	C2	At least one letter decrypted correctly (1)																																			
C3	At least one word decrypted correctly (1)																																				
C4	Decrypted message 'meet at dawn' (1)																																				

Question	mp	Answer	Additional Guidance	Mark
3 (d)(i)	D1	Award 1 mark for: cipherLetter / a single encrypted letter (1)	Do not accept word/message/text	(1)
3 (d)(ii)	D2	Award 1 mark for any of: - keywordLetter - plaintextLetter	Ignore case	(1)
3 (d)(iii)	D2	Award 1 mark for any of: - subprogram that is already defined - subprogram that is already written - subprogram that is already compiled - subprogram that can be called without having to write code for it	Must be clear they are referring to subprogram provided by the language itself	(1)

Question	mp	Answer	Additional Guidance	Mark
3 (a)	A1	Get plaintext and store in plaintext variable	Accept alternative wording	(1)
	A2	Get key and store in key variable	Line numbers may be different compared to the examples shown	(1)
	A3	Validate key (must repeat until a valid key has been input)		(1)
	A4	Open file to write/append		(1)
	A5	Write cipher text	When testing the completed code use lowercase for the input	(1)
	A6	Close file		(1)
	A7	Displays ciphertext		(1)
	A8	Executing and producing correct output to file and screen (must have A3)		(1)

Code examples

Python

```

15
16 # Add your code to get the plaintext and convert it to lowercase
17 plaintext = input("Enter the plaintext using lowercase letters: ")
18 plaintext = plaintext.lower()

```

```

# Add your code to get the key and make sure the key is between 1 and 25
while key < 1 or key > 25:
    key = int(input("Enter the key - a number between 1 and 25 "))

```

```

39 # Add your code to write the ciphertext to a text file
40 cipherFile = open("Cipher.txt", "w")
41 cipherFile.write(ciphertext)
42 cipherFile.close()

```

```

40
41 # Add your code to display the ciphertext
42 print("The ciphertext message is:", ciphertext)
43

```



Question	mp	Answer	Additional Guidance	Mark																								
5 (b)(i)	B1 B2 B3	Award one mark for each: • Pass 1 correct in first row of the response (1) • Pass 2 OR 3 correct in any row of the response (1) • All passes correct and no extra passes (1) <table border="1"><tr><td>2</td><td>3</td><td>8</td><td>1</td><td>9</td><td>10</td></tr><tr><td>2</td><td>3</td><td>1</td><td>8</td><td>9</td><td>10</td></tr><tr><td>2</td><td>1</td><td>3</td><td>8</td><td>9</td><td>10</td></tr><tr><td>1</td><td>2</td><td>3</td><td>8</td><td>9</td><td>10</td></tr></table>	2	3	8	1	9	10	2	3	1	8	9	10	2	1	3	8	9	10	1	2	3	8	9	10	If the final row (1, 2, 3, 8, 9, 10) is repeated treat as the same row.	(3)
2	3	8	1	9	10																							
2	3	1	8	9	10																							
2	1	3	8	9	10																							
1	2	3	8	9	10																							
5 (b)(ii)	B4 B5	Award up to 2 marks for a linked explanation such as: • The sorting is done in the same place as the original data (1), which means only one additional variable is needed to store the value being swapped (when the values are out of order) (1) • All sort and swapping operations are done on the original data (1) which means the amount of memory needed is constant (1) • The list does not need to be split (1), which would use more memory (1)		(2)																								

Question	mp	Answer	Additional Guidance	Mark
5 (c)	C1 C2 C3	Award up to 3 marks for a linked description that includes: - Start at the beginning of the list (1) - Compare each value in the list with the search item (1) - Repeat until a match is found / the end of the list is reached (1)		(3)

Question	mp	Answer	Additional Guidance	Mark
5 (a)	A1 A2 A3 A4 A5 A6 A7 A8 A9	Award 1 mark for each of: - Variable to store a number of incorrect passwords (1) - Loop used (1) - Each password checked to see if the first character is an uppercase letter (1) - Each password checked to see if it contains a digit (1) - Only check for digits if first character is uppercase OR Only checks if first character is uppercase if at least one digit is present (1) - Number of incorrect passwords incremented correctly (1) (allow follow through) - All incorrect passwords displayed (1) (allow follow through but must try to display those without uppercase and those without digits) - Text file closed (1) - Number of incorrect passwords displayed (1) (allow follow through)		(9)

Code examples

Python

```
# Add your code here
incorrectPasswords = 0

for password in passwordFile:
    valid = False

    if password[0] in alphabet:
        index = 1
        while (valid == False) and (index in range(1, len(password))):
            if password[index] in digit:
                valid = True
            else:
                index = index + 1

    if not valid:
        incorrectPasswords = incorrectPasswords + 1
        print(password)

print(incorrectPasswords, "passwords are incorrect.")

passwordFile.close()
```



Question	mp	Answer	Additional Guidance	Mark
2 (c)	Award 1 mark for each set of test data.			(3)
	C1			
	C2	booksSold		
	C3	profit		
		Test data		
		Expected results		
		booksSold	Either of	Poor performances this week
		profit	- booksSold = 4 - profit = 4	
		booksSold	5	Sales and profit are good this week
		profit	10	
		booksSold	21	Sales and profit are excellent this week
		profit	20	

Question	mp	Answer	Additional Guidance	Mark
2 (a)	Award one mark for each of:		Logic of algorithm must be followed as set out.	(11)
	A1	At least one variable with a suitable variable name	Alternatives must address each point.	
	A2	username = bard423		
	A3	password = nX2934?	Do not penalise candidates who attempt more than the stated requirements.	
	A4	Loop used		
	A5	Username or password entered	Do not penalise spelling mistakes in the assignment of username and/or password	
	A6	Username or password stored in variable(s)		
	A7	At least one suitable input message	Do not penalise spelling mistakes and alternative wording of the output.	
	A8	Checks username and password		
	A9	Appropriate error message(s) displays		
	A10	Welcome message displayed		
	A11	Executing and producing correct output (must have A4)		

Code example

Python

```
# Initialise variables
username = "bard423"
password = "nX2934?"
count = 0
inputUsername = ""
inputPassword = ""

# Print prompts, take and check input from user
while inputUsername != username or inputPassword != password:
    if count > 0:
        print("There is a problem with the login details. Try again")
    count = count + 1
    inputUsername = input("Enter your username ")
    inputPassword = input("Enter your password ")
print("Welcome")
```

Question	mp	Answer	Additional Guidance	Mark
2 (b)	Award 1 mark for each correct condition.		Accept alternatives e.g. Line 11 booksSold <= 4 etc.	(4)
		Condition		
	B1	Number of books sold is under 5 or profit made is under 5		
	B2	Number of books sold is over 20; profit made is at least 20		
	B3	Number of books sold is at least 5; profit made is at least 10		
	B4	All other inputs	Alert manager	

Code example

Python

```
11 if booksSold < 5 or profit < 5:
12     print("Poor performance this week")
13 elif booksSold > 20 and profit >= 20:
14     print("Sales and profit are excellent this week")
15 elif booksSold >= 5 and profit >= 10:
16     print("Sales and profit are good this week")
17 else:
18     print("Alert manager")
```



Question	mp	Answer	Additional Guidance	Mark
2 (b)	B1 B2 B3 B4	Award up to 4 marks for a linked explanation such as: - The list is not sorted (1) Juan should come after Elija if the list was sorted / Elija is compared with Juan (1), so the bottom half of the list would be discarded (1) after the first pass through the loop (1) - Binary search works on a sorted list (1) because it uses divide and conquer (1) half the list discarded each pass through (1) based on comparison of search item with middle item (1)		(4)
Question	mp	Answer	Additional Guidance	Mark
2 (c)	C1 C2 C3 C4 C5	Award 1 mark for each of: Split into two sets of four (1) Split each set into two pairs (1) Split each pair into single elements (1) Merge elements into sorted pairs (1) Merge pairs into sorted sets (1) 9 1 7 6 3 5 2 8 9, 1, 7, 6 3, 5, 2, 8 9, 1 7, 6 3, 5 2, 8 9 1 7 6 3 5 2 8 1, 9 6, 7 3, 5 2, 8 1 6, 7, 9 2, 3, 5, 8 1 2 3 5 6 7 8 9	Allow step at C3 to be implied	(5)

Question	mp	Answer	Additional Guidance	Mark
2 (a)	A1 A2 A3 A4 A5 A6 A7 A8 A9 A10	Award one mark for each of: Input as an integer (1) Loop with correct conditions start <= end and not found (1) Use of correct symbol for integer division (in conversion of DIV) or type coercion (1) Check to see if number in middle matches item (1) Set found to true (1) Check to see if item is less than number in middle (1) Set end to middle - 1 (1) else - set start to middle + 1 (1) count incremented by 1 (1) Program execution is fully correct (1)	Logic of algorithm must be followed as set out. Alternatives must address each point. Do not penalise candidates who attempt more than the stated requirements. Do not penalise spelling mistakes in the input message.	(10)

Code examples

C#	<pre> Console.WriteLine("What is the number to find? "); int item = Convert.ToInt32(Console.ReadLine()); while (start <= end && !found) { middle = (start + end) / 2; if (numberList[middle] == item) { found = true; } else { if (item < numberList[middle]) { end = middle - 1; } else { start = middle + 1; } } count++; } </pre>
----	---

Java	<pre>Scanner input = new Scanner(System.in); System.out.print("What is the number to find? "); int item = Integer.parseInt(input.nextLine()); while (start <= end && !found) { middle = (start + end) / 2; if (numberList[middle] == item) { found = true; } else { if (item < numberList[middle]) { end = middle - 1; } else { start = middle + 1; } } count ++; }</pre>	
Python	<pre>item = int(input("What is the number to find? ")) while start <= end and not found: middle = (start + end) // 2 if numberList[middle] == item: found = True else: if item < numberList[middle]: end = middle - 1 else: start = middle + 1 count = count + 1</pre>	

Question	Designated marks:	Additional Guidance	Mark
Q5	Designated marks: 1. Initialise variables for at least two of total number of books, count of pupils and average 2. Initialise variables for gold, silver and bronze / sort array in order of books read 3. Create loop for identifying (and printing) those borrowing fewer than 10 books 4. Print out appropriate details (must include pupil ID) 5. Print out total of books borrowed 6. Print out average number of books borrowed 7. Identification of gold medal winner 8. Identification of silver medal winner 9. Identification of bronze medal winner 10. Details of at least one of gold, silver and bronze medal winner printed (minimum last name or first name) 11. Details of all three medal winners (first name and last name)	Award marks independently Award marks for reasonable attempt to meet requirement.	(11)

Question 5 Overview marks:				
Mark Band 0	Mark Band 1	Mark Band 2	Mark Band 3	Marks
0	1 – 3	4 – 6	7 – 9	
No rewardable content	Little attempt to decompose the problem into component parts Some parts of the logic are clear and appropriate to the problem Some appropriate use and manipulation of data types, variables, data structures and program constructs Parts of the code are clear and readable Finished program will not be flexible enough with other data sets or input The program meets some of the given requirements	Some attempt to decompose the problem into component parts Most parts of the logic are clear and mostly appropriate to the problem The use and manipulation of data types, variables and data structures and program constructs is mostly appropriate Code is mostly clear and readable Finished program will function with some but not all other data sets or input The program meets most of the given requirements	The problem has been decomposed into component parts The logic is clear and appropriate to the problem The use and manipulation of data types, variables and data structures and program constructs is appropriate Code is clear and readable Finished program could be used with other data sets or input The program fully meets the given requirements	(9)

Python Example code

5	<pre> 26 # Write your code below this line 27 28 total = 0 29 gold = 0 30 silver = 0 31 bronze = 0 32 goldID = 0 33 silverID = 0 34 bronzeID = 0 35 36 print ("Pupil ID of pupils who have read less than 10 books ") 37 38 for row in range(len(libraryRecord)): 39 books = int(libraryRecord[row][3]) 40 if (books > gold): 41 bronze = silver 42 silver = gold 43 gold = books 44 goldID = row 45 elif (books > silver): 46 bronze = silver 47 silver = books 48 silverID = row 49 elif (books > bronze): 50 bronze = books 51 bronzeID = row 52 53 total = total + books 54 55 if (books < 10): 56 print ("\t", libraryRecord[row][0]) 57 58 print ("\n") 59 print ("Total number of books read by pupils: ", total) 60 61 print ("\n") 62 print ("Average number of books read by pupils: ", total / len(libraryRecord[0])) 63 64 print ("\n") 65 print ("Gold medal winner is: \t\t", libraryRecord[goldID][1], " ", libraryRecord[goldID][2]) 66 print ("Silver medal winner is: \t", libraryRecord[silverID][1], " ", libraryRecord[silverID][2]) 67 print ("Bronze medal winner is: \t", libraryRecord[bronzeID][1], " ", libraryRecord[bronzeID][2]) </pre>	
---	---	--



For Q6, the first 11 marks are for coding that matches requirements of task. The remaining 9 marks should be allocated on a best fit.

Question	mp	ref	Answer	Additional Guidance	Mark
6	A1	2.2.1	Initialise variable for total sales by member of staff (1)	Sales for each member of staff	(11)
	A2	2.5.1	Add succeeding months' sales to total sales (1)	George Taylor 28466 Fehlix Chayne 35250	
	A3	2.1.6	Display total sales by member of staff (1)	Sumren Bergen 30186 Samira Beckle 37430	
	A4	2.2.2	Loop repeats for each member of staff (1)	Nellie Bogart 20448 Cheryl Grouth 34383	
	A5	2.2.1	Running team total sales initialised (1)	Danuta Graunt 26421 Jaiden Deckle 21507	
	A6	2.2.2	Add succeeding individual staff sales to team total (1)	Jimran Caliks 32415 Deynar Derran 28590	
	A7	2.1.6	Display final running total (1)	=====	
	A8	2.1.6	Staff member with highest individual sales identified (1)	Total staff sales: 295096	
	A9	2.1.6	Staff member with second highest individual sales identified (1)	Highest sales by: Samira Beckle with 37430	
	A10	2.1.6	Information about staff with highest sales displayed (1)	Second highest sales by: Fehlix Chayne with 35250	
	A11	2.1.6	Information about staff with second highest sales displayed (1)	>>>	

Band 1 (1-3 marks)	Band 2 (4-6 marks)	Band 3 (7-9 marks)	Mark
Little attempt to decompose the problem into component parts	Some attempt to decompose the problem into component parts	The problem has been decomposed into component parts	(9)
Some parts of the logic are clear and appropriate to the problem	Most parts of the logic are clear and mostly appropriate to the problem	The logic is clear and appropriate to the problem	
Some appropriate use and manipulation of data types, variables, data structures and program constructs	The use and manipulation of data types, variables and data structures and program constructs is mostly appropriate	The use and manipulation of data types, variables and data structures and program constructs is appropriate	
Parts of the code are clear and readable	Code is mostly clear and readable	Code is clear and readable	
Finished program will not be flexible enough with other data sets or input	Finished program will function with some but not all other data sets or input	Finished program could be used with other data sets or input	
The program meets some of the given requirements	The program meets most of the given requirements	The program fully meets the given requirements	

Code example

Python

```

32 allStaffTotal = 0
33 highestSales = 0
34 secondSales = 0
35 high = 0
36 second = 0
37
38 print ("Sales for each member of staff","\n\n")
39 for staff in staffSales:
40     staffTotal = 0
41     sales = 3
42     while (sales < 9):
43         staffTotal = staffTotal + staff[sales]
44         sales = sales + 1
45     print (staff[1],staff[2],staffTotal)
46     if (staffTotal > secondSales):
47         if (staffTotal > highestSales):
48             highestSales = staffTotal
49             high = staff
50         else:
51             secondSales = staffTotal
52             second = staff
53     allStaffTotal = allStaffTotal + staffTotal
54 print ("=====\n\nTotal sales for all staff: ", allStaffTotal)
55
56 print ("\n\nHighest sales by: ",high[1],high[2], " with ", highestSales)
57 print ("\n\nSecond highest sales by: ",second[1],second[2], " with ", secondSales)
58

```



For Q5, the first 11 marks are for coding that matches requirements of task. The remaining 9 marks should be allocated on a best fit.

Question	mp	Answer	Additional Guidance	Mark
5	addPlayerName()			
	A1	Suitable prompt for player name and assigned to suitable variable		
	guessCapital()			
	A2	Ensure question can only be used once	Do not award if more than one question variable e.g. question1, question2 etc.	
	A3	Question includes suitable message and country name		
	A4	Check made to see if guess is correct		
	A5	If guess correct score incremented		
	A6	If guess is incorrect suitable message displayed		
	A7	If guess incorrect capital concatenated with message		
	A8	Repeated for a minimum of five questions	Do not award if questions are asked manually e.g. question1, question2, repeated code for each question etc. Do not award if 5 unique questions are not asked while the program is running	
	Main Program			
	A9	Player name or score displayed	Do not award if the return value from at least one function is not used	
A10	At least one menuChoice calls correct subprogram			
A11	Main program calls the two sub-programs correctly			

(11)

Band 1 (1-3 marks)	Band 2 (4-6 marks)	Band 3 (7-9 marks)	Mark
Little attempt to decompose into component parts	Some attempt to decompose into component parts	The problem has been decomposed into component parts	(9)
Some parts of the logic are clear and appropriate to the problem	Most parts of the logic are clear and mostly appropriate to the problem	The logic is clear and appropriate to the problem	
Some appropriate use and manipulation of data types, variables, data structures and program constructs	The use and manipulation of data types, variables and data structures and program constructs is mostly appropriate	The use and manipulation of data types, variables and data structures and program constructs is appropriate	
Parts of the code are clear and readable	Code is mostly clear and readable	Code is clear and readable	
Finished program will not be flexible enough with other data sets or input	Finished program will function with some but not all other data sets or input	Finished program could be used with other data sets or input	
The program meets some of the given requirements	The program meets most of the given requirements	The program fully meets the given requirements	



Code examples	
Python	Add player name function <pre> player = "" while(player == ""): player = input("Enter your player name ") return player #----- </pre> Main program <pre> menuChoice = 0 score = 0 playerName = "" while menuChoice != 3: displayMenu() menuChoice = getMenuChoice() # Add your code to: # call the relevant subprogram if the menu choice is 1 or 2 # display the player name and the score if the menu choice is 3 if menuChoice == 1: playerName = addPlayerName() elif menuChoice == 2: score = guessCapital() else: print("Well done " + playerName + ". The score is " + str(score)) </pre>

	Guess capital city function <pre> # Add your code here questionCount = 1 questionScore = 0 # Ask 5 questions while questionCount <= 5: questionChoice = -1 questionNumbers = "" # Build a string containing the question numbers available for question in questions: if question != 0: questionNumbers += str(question) + " " # Ensure valid question number is chosen while str(questionChoice) not in questionNumbers: questionChoice = int(input("Pick a number from " + questionNumbers)) # Get the country and its capital country = countries[questionChoice - 1] capital = capitals[questionChoice - 1] # Display the country and get the guess guess = input("What is the capital of " + country + "? ").lower() # If the guess is correct display message and increment score if guess == capital.lower(): print("Well done, you guessed correctly") questionScore += 1 else: # Otherwise display the country name and correct capital print("You did not guess correctly. The capital of " + country + " is " + capital) # Increment the number of questions asked questionCount = questionCount + 1 # Set the question number to 0 so that it cannot be guessed again questions[questionChoice - 1] = 0; questions[questionChoice - 1] = 0; # return the score to the main menu return questionScore </pre>
--	--

For Q5, the first 11 marks are for coding that matches requirements of task. The remaining 9 marks should be allocated on a best fit

Question	mp	Answer	Additional Guidance	Mark
5	A1	At least three variables initialised appropriately		(1)
	A2	Word input		(1)
	A3	Program repeats until 1 is input		(1)
	A4	Loop through each word in the array		(1)
		Words that begin with the same letter		
	A5	Check to see if the first letter of the input word matches the first letter of at least one word in the array		(1)
	A6	The number of words that begin with the same letter displayed		(1)
		Words that contain the input word		
	A7	Check to see if the word contains the input word		(1)
	A8	Number of letters in the word calculated		(1)
	A9	At least one word from the array that contains the input word displayed		(1)
	A10	The number of words that contain the input word displayed		(1)
	A11	Identification of the longest word or the shortest word		(1)

Band 1 (1-3 marks)	Band 2 (4-6 marks)	Band 3 (7-9 marks)	Mark
Little attempt to decompose into component parts	Some attempt to decompose into component parts	The problem has been decomposed into component parts	(9)
Some parts of the logic are clear and appropriate to the problem	Most parts of the logic are clear and mostly appropriate to the problem	The logic is clear and appropriate to the problem	
Some appropriate use and manipulation of data types, variables, data structures and program constructs	The use and manipulation of data types, variables and data structures and program constructs is mostly appropriate	The use and manipulation of data types, variables and data structures and program constructs is appropriate	
Parts of the code are clear and readable	Code is mostly clear and readable	Code is clear and readable	
Finished program will not be flexible enough with other data sets or input	Finished program will function with some but not all other data sets or input	Finished program could be used with other data sets or input	
The program meets some of the given requirements	The program meets most of the given requirements	The program fully meets the given requirements	

Code examples

C#

```

14 // Add your code here
15 string inputWord = "";
16
17 while (inputWord != "1")
18 {
19     int shortest = 5000;
20     int longest = 0;
21     String shortestWord = "";
22     String longestWord = "";
23
24     Console.WriteLine("Enter a word or 1 to exit ");
25     inputWord = Console.ReadLine();
26
27     if (inputWord != "1")
28     {
29         int count = 0;
30
31         foreach (String word in wordArray)
32         {
33             if (word[0] == inputWord[0])
34             {
35                 Console.WriteLine(word);
36                 count ++;
37             }
38         } // End of Loop
39
40         Console.WriteLine(count + " word(s) beginning with " + inputWord[0]);
41         Console.WriteLine("-----");
42
43         count = 0;
44
45         foreach (String word in wordArray)
46         {
47             if (word.Contains(inputWord))
48             {
49                 count ++;
50                 Console.WriteLine(word);
51
52                 int length = word.Length;
53
54                 if (longest < length)

```



Java

```

13 // Add your code here
14 Scanner input = new Scanner(System.in);
15 String inputWord = "";
16
17 // While there is a word run the program
18 while (!inputWord.equals("1")) {
19
20     int shortest = 5000;
21     int longest = 0;
22     String shortestWord = "";
23     String longestWord = "";
24
25     System.out.println("Enter a word or 1 to exit ");
26     System.out.println("-----");
27     inputWord = input.nextLine();
28     if (!inputWord.equals("1")){
29         // Find words that begin with the same letter as the input word
30         int count = 0;
31
32         // Get each word in the array
33         for (String word: wordArray){
34
35             // If the first character of the word is the same as the first character of
36             // the input word then print the word
37             if (word.charAt(0) == inputWord.charAt(0)){
38                 System.out.println(word);
39                 count ++;
40             }
41         } // End of for Loop
42
43         // Print the total number of words that begin with the same letter
44         System.out.println(count + " word(s) beginning with " + inputWord.charAt(0));
45         System.out.println("-----");
46
47         // Find words that contain the input word
48         count = 0;
49
50         // Get each word in the array
51         for (String word: wordArray){
52             // Check to see if the word contains the inputWord
53             if (word.contains(inputWord)){
54                 count ++;
55                 System.out.println(word);
56
57                 // Set the Length of the word
58                 int length = word.length();
59

```

Python

```

13 # While there is a word run the program
14 while inputWord != "1":
15
16     shortest = 5000
17     longest = 0
18     shortestWord = ""
19     longestWord = ""
20
21     inputWord = input("Enter a word or 1 to exit: ")
22     print("-----")
23
24     if inputWord != "1":
25         # Find words that begin with the same letter as the input word
26         count = 0
27         # Get each word in the array
28         for word in wordArray:
29             # If the first character of the word is the same as the first character of
30             # the input word then print the word
31             if word[0] == inputWord[0]:
32                 print(word)
33                 count = count + 1
34
35         # Print the total number of words that begin with the same letter
36         print("\n", count, "word(s) beginning with", inputWord[0])
37         print("-----")
38
39         # Find words that contain the input word
40         count = 0
41         # Get each word in the array
42         for word in wordArray:
43             # Check to see if the word contains the inputWord
44             if inputWord in word:
45                 count = count + 1
46                 print(word)
47                 # Set the length of the word
48                 length = len(word)
49
50             # If the length is longer than the current longest replace it and the longestWord
51             if longest < length:
52                 longest = length
53                 longestWord = word
54
55             # If the length is shorter than the shortest replace it and the shortestWord
56             if shortest > length:
57                 shortest = length
58                 shortestWord = word

```



Question	mp	Answer	Additional Guidance	Mark
6	A1	At least two appropriate subprograms used		(1)
	A2	Repeat guess until the word has been guessed OR no attempts left		(1)
	A3	Generate and display the number of attempts left	Must see working correctly	(1)
	A4	Request input of the word		(1)
	A5	Validate the input of the word to ensure it is the same length as the random word		(1)
	A6	Check to see if the guess matches the random word		(1)
	A7	If the word has been guessed, display message including the random word and number of attempts		(1)
	A8	Keep track and display letters that are in the word	Must see working	(1)
	A9	Keep track and display letters that are not in the word	Must see working	(1)
	A10	Letters appear only once in each list	Must see working	(1)
	A11	Display lose message including the random word		(1)

Band 1 (1-3 marks)	Band 2 (4-6 marks)	Band 3 (7-9 marks)	Mark
Little attempt to decompose into component parts	Some attempt to decompose into component parts	The problem has been decomposed into component parts	(9)
Some parts of the logic are clear and appropriate to the problem	Most parts of the logic are clear and mostly appropriate to the problem	The logic is clear and appropriate to the problem	
Some appropriate use and manipulation of data types, variables, data structures and program constructs	The use and manipulation of data types, variables and data structures and program constructs is mostly appropriate	The use and manipulation of data types, variables and data structures and program constructs is appropriate	
Parts of the code are clear and readable	Code is mostly clear and readable	Code is clear and readable	
Finished program will not be flexible enough with other data sets or input	Finished program will function with some but not all other data sets or input	Finished program could be used with other data sets or input	
The program meets some of the given requirements	The program meets most of the given requirements	The program fully meets the given requirements	

Code examples

Python

Subprograms

```
# Add your subprograms here
def checkInput(pWordLength):
    # Validate input to make sure it is the same length as the secret word
    check = ""
    while len(check) != pWordLength:
        check = input("Enter your guess. The secret word has " + str(pWordLength) + " letters: ")
    return(check)

def checkLettersInWord(pGuess, pWordToGuess, pLetters, type):
    for letter in pGuess: # Check each Letter in the guess
        if type == 0: # The check to carry out is for correct Letters
            # If the letter is in the secret word and not already in the correctLetters, add it
            if letter in pWordToGuess and letter not in pLetters:
                pLetters = pLetters + letter
        else: # The check to carry out is for wrong Letters
            # If the letter is not in the secret word and not already in the wrongLetters, add
            if letter not in pWordToGuess and letter not in pLetters:
                pLetters = pLetters + letter
    return pLetters

def display(pCorrectLetters, pWrongLetters):
    # Display the correct and wrong Letters
    if len(pCorrectLetters) > 0:
        print("Letter(s) in the secret word:", pCorrectLetters)

    if len(pWrongLetters) > 0:
        print("Letter(s) not in the secret word:", pWrongLetters)
```



Main program

```
# Add your main program code here
wordLength = len(wordToGuess)
maxAttempts = wordLength + 3
numAttempts = 0
correctLetters = ""
wrongLetters = ""
guessed = False

# Loop until the no attempts are left or the secret word has been guessed
while numAttempts < maxAttempts and guessed == False:
    print("You have", maxAttempts - numAttempts, "attempts to guess the secret word.")
    numAttempts = numAttempts + 1

    guess = checkInput(wordLength) # Validate the input

    if guess == wordToGuess: # Check to see if the secret word has been guessed
        guessed = True
        print("Well done. You guessed the secret word was", wordToGuess, "in", numAttempts, "attempts.")

    else: # If it hasn't been guessed check for correct and wrong letters
        correctLetters = checkLettersInWord(guess, wordToGuess, correctLetters, 0)
        wrongLetters = checkLettersInWord(guess, wordToGuess, wrongLetters, 1)
        display(correctLetters, wrongLetters)

# Display game over message if attempts have run out and the secret word has not been guessed
if numAttempts == maxAttempts and guessed == False:
    print("Game over. You did not guess that the secret word was", wordToGuess)
```



Question	Answer	Additional Guidance	Mark
2(b)(i)	24.0 (1)	Accept 24	(1)
2(b)(ii)	One of: By the rules of precedence multiplication is done before addition. (1) Without the brackets the addition would be done to the total of the multiplication. (1) By the rules of the BODMAS/BIDMAS additions happen after multiplication. (1) newSpeed and intialSpeed have to be added together before multiplying (by time). (1)		(1)

Question	Answer	Additional Guidance	Mark						
2(b)(iii)	Real (1) Because real x integer = real (1)	Accept double / float Accept 'because it could be a number with decimal places'	(2)						
2(b)(iv)	27.375 (1)		(1)						
Q2(c)(i)	<table><tr><td>C#:</td><td>16 or 20 (1)</td></tr><tr><td>Java:</td><td>16 or 21 (1)</td></tr><tr><td>Python:</td><td>8 or 10 (1)</td></tr></table>	C#:	16 or 20 (1)	Java:	16 or 21 (1)	Python:	8 or 10 (1)		(1)
C#:	16 or 20 (1)								
Java:	16 or 21 (1)								
Python:	8 or 10 (1)								
Q2(c)(ii)	<table><tr><td>C#:</td><td>29 or 31 (1)</td></tr><tr><td>Java:</td><td>24 or 26 (1)</td></tr><tr><td>Python:</td><td>14 or 15 (1)</td></tr></table>	C#:	29 or 31 (1)	Java:	24 or 26 (1)	Python:	14 or 15 (1)		(1)
C#:	29 or 31 (1)								
Java:	24 or 26 (1)								
Python:	14 or 15 (1)								
Q2(c)(iii)	Any of: 1. Message (1) 2. startPos (Python) / pos (Java or C#) (1) 3. characters (1)		(1)						
Q2(c)(iv)	newMessage (1)	Do not accept outMessage	(1)						
Q2(c)(v)	Response starts with 'c' (1) Response has six letters in sequence from 'educational' (1)	Marks can be awarded from Q2(c)(vi)	(2)						
Q2(c)(vi)	1. Returns substring / crops a string (1) 2. Located between the two input values / with number of characters identified in the last parameter (characters) and starting at the second parameter (startPos) (1)	Award two marks for: It produces a substring of the message starting at position startPos with length characters Award up to 2 marks for a correct interpretation of the code (e.g 'obtain a newMessage' – is worth 1) Accept a correct generic explanation of what a subprogram does for 1 mark	(2)						

Question	Answer	Additional Guidance	Mark
2(a)	Award 1 mark for each of: 1. set counter to 1 and guess to 0 (1) 2. set answer to random value between 1 and 10 (1) 3. request input of and accept an integer value for guess (1) 4. create a while loop with a correct condition (1) 5. increment the counter (1) 6. use if else selection to determine and display appropriate output message for incorrect guesses (1) 7. request input of and accept another integer value for a guess inside the loop (1) 8. display correct guess and count of guesses (1) 9. compiling without syntax errors (1) 10. coding meets all requirements of question (1)	Candidates are required to open the file Q02a in the code editor. Amended code should be saved as Q02aFINISHED. Logic of algorithm must be followed as set out. Alternatives must address each point. Do not penalise candidates who attempt more than the stated requirements.	(10)

2(a)

```
5  # Initialise variables
6  counter = 1
7  answer = randint(1,10)
8  guess = 0
9
10 # Print prompt and take guess from user
11 guess = int(input("Enter a number from 1 to 10: "))
12
13 # Create while loop to check answer
14 while guess!=answer:
15     counter = counter + 1
16     if(guess > answer):
17         print (guess, " was too high. Try again.")
18     else:
19         print (guess, " was too low. Try again.")
20
21     guess = int(input("Guess again: "))
22
23 # Report the correct answer to the user and indicate the number of guesses
24 print ("You guessed", guess, "in", counter, "guesses")
25
```

