

Programming - GCSE Computer Science Paper 1**1. No.1(1.2_1.5)**

0	1
---	---

2

 The variable `x` is assigned a value using the statement:
$$x \leftarrow \text{LEN}(\text{state})$$
Using **Figure 1**, what is the value of `x`?

Shade one lozenge.

[1 mark]

- | | | | |
|----------|----|---|---|
| A | 1 | <table border="1" style="margin: auto;"><tr><td style="width: 20px; height: 20px; text-align: center; line-height: 20px;">○</td></tr></table> | ○ |
| ○ | | | |
| B | 5 | <table border="1" style="margin: auto;"><tr><td style="width: 20px; height: 20px; text-align: center; line-height: 20px;">○</td></tr></table> | ○ |
| ○ | | | |
| C | 10 | <table border="1" style="margin: auto;"><tr><td style="width: 20px; height: 20px; text-align: center; line-height: 20px;">○</td></tr></table> | ○ |
| ○ | | | |
| D | 12 | <table border="1" style="margin: auto;"><tr><td style="width: 20px; height: 20px; text-align: center; line-height: 20px;">○</td></tr></table> | ○ |
| ○ | | | |

0 1 . 3

What is the result of concatenating the contents of the variables `city` and `landmark` in **Figure 1**?

Shade one lozenge.

[1 mark]

- | | | |
|----------|---|-----------------------|
| A | <code>San Francisco Alcatraz Island</code> | ☐ |
| B | <code>San Francisco,Alcatraz Island</code> | ☐ |
| C | <code>San Francisco, Alcatraz Island</code> | ☐ |
| D | <code>San FranciscoAlcatraz Island</code> | ☐ |

0 1 . 4

The subroutine `SUBSTRING` extracts characters from a given string.

For example, `SUBSTRING(3, 5, 'Computing')` would return `put`

The variable `y` is assigned a value using the statement:

```
y ← SUBSTRING(4, 7, landmark)
```

Using **Figure 1**, what is the value of `y`?

Shade one lozenge.

[1 mark]

- | | | |
|----------|-------------------|-----------------------|
| A | <code>Alca</code> | ☐ |
| B | <code>Atra</code> | ☐ |
| C | <code>land</code> | ☐ |
| D | <code>traz</code> | ☐ |

Figure 1 has been included again below.

Figure 1

```
country ← 'United States of America'  
state ← 'California'  
city ← 'San Francisco'  
landmark ← 'Alcatraz Island'
```

0 1 . 5 The subroutine `POSITION` finds the first position of a character in a string.

For example, `POSITION('Computing', 'p')` would return 3

The variable `z` is assigned a value using the statement:

```
z ← POSITION(landmark, 't')
```

Using **Figure 1**, what value is assigned to `z`?

Shade one lozenge.

[1 mark]

A -1

☐

B 3

☐

C 4

☐

D 5

☐

2. No.3

Explain one advantage of the structured approach to programming.

[2 marks]

3. No.7

A theme park charges £15 per person for a daily ticket. If there are six or more people in a group, the group is given a £5 discount.

Write a C# program to calculate the total charge for a group of people visiting the theme park.

The program must:

- get the user to enter the number of people in a group
- calculate the total charge by:
 - charging £15 per person
 - reducing the total charge by £5 if there are six or more people
- output the total charge.

You **should** use meaningful variable name(s) and C# syntax in your answer.

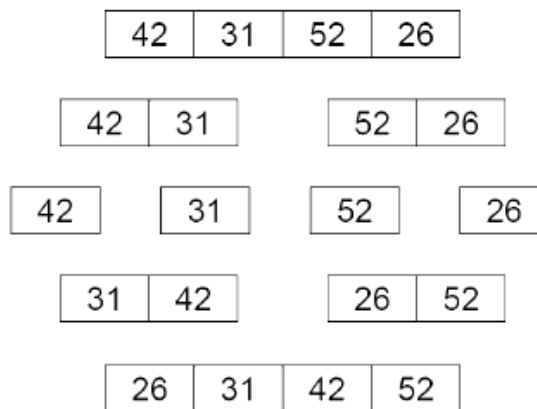
The answer grid below contains vertical lines to help you indent your code.

[6 marks]

4. No.8

Figure 7 shows a merge sort being carried out on a list.

Figure 7



Explain how the merge sort algorithm works.

[4 marks]

5. No.13

A programmer is writing a game. The game uses a 3 x 3 grid containing nine squares.

Figure 14

	A	B	C
1			
2			
3			X

In the game, a square on the grid is referred to by a letter and a number. For example, square C3 in **Figure 14** contains an X.

Figure 15 shows part of a C# program that checks the grid reference entered by a player.

The grid reference is valid if:

- there are exactly two characters
- the first character entered is A, B or C
- the second character entered is 1, 2 or 3.

Figure 15

```
bool check = false;
while (check == false) {
    string square = "";
    while (square.Length != 2) {
        Console.WriteLine("Enter grid reference (eg C2): ");
        square = Console.ReadLine();
        square = square.ToUpper();
    }
}
```

The C# function `ToUpper()` converts letters into uppercase, eg `b1` would be converted to `B1`

Extend the program from **Figure 15** so it completes the other checks needed to make sure a valid grid reference is entered.

Your extended program must:

- use the variable `check`
- repeat the following steps until a valid grid reference is entered:
 - get the user to enter a grid reference
 - output an appropriate message if the grid reference entered is not valid.

You **should** use meaningful variable name(s) and C# syntax in your answer.

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

[6 marks]

```
bool check = false;
```

```
while (check == false) {
```

	string square = "";
--	---------------------

	while (square.Length != 2) {
--	------------------------------

		Console.WriteLine("Enter grid reference (eg C2): ");
--	--	--

		<code>square = Console.ReadLine();</code>
--	--	---

		<code>square = square.ToUpper();</code>
--	--	---

[illegible]

}				
---	--	--	--	--

6. No.14

50 students have voted for the music genre they like best.

Figure 16 shows an **incomplete** algorithm, represented using pseudo-code, designed to output the highest or lowest results of the vote.

The programmer has used a two-dimensional array called `results` to store the genre and the number of votes for each genre.

Parts of the algorithm are missing and have been replaced with the labels **L1** to **L3**.

Figure 16

```

SUBROUTINE showResults(method, numberOfGenres)
    results ← [['Pop', 'Post-Punk', 'Techno', 'Metal',
                'Dance'], ['7', '19', '14', '1', '9']]
    pos ← 0
    high ← -1
    IF method = 'HIGHEST' THEN
        FOR i ← 0 TO numberOfGenres - 1
            Votes ← STRING_TO_INT(results[L1][i])
            IF votes > high THEN
                high ← votes
                pos ← L2
            ENDIF
        ENDFOR
    ELSE
        OUTPUT 'not yet working'
    ENDIF
    IF high ≠ -1 THEN
        OUTPUT results[0][pos], ' with ', results[1][pos]
    ENDIF
ENDSUBROUTINE

OUTPUT 'Show the genre with the HIGHEST or LOWEST number
of votes? '
method ← USERINPUT
showResults(L3, 5)

```

State what should be written in place of the labels **L1** to **L3** in the algorithm in **Figure 16**.

[3 marks]

L1 _____

L2 _____

L3 _____

7. No.15

A group of people have a meal in a restaurant. Instead of one person paying for the whole meal, each person will pay for what they eat.

Write a C# program that asks each person in the group how much they are paying towards the meal and works out when the bill is fully paid. Each person can pay a different amount.

The program should:

- get the user to enter the total amount of the bill
- get a person to enter how much they are paying towards the bill
- subtract the amount entered from the bill:
 - if the amount left to pay is more than 0, output how much is left to pay and repeat until the amount left to pay is 0 or less
 - if the amount left to pay is 0, then output the message `Bill paid`
 - if the amount left to pay is less than 0, then output the message `Tip is` and the difference between the amount left to pay and 0

You **should** use meaningful variable name(s) and C# syntax in your answer.

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

[8 marks]

8. No.16

Question 16 is about a dice game played against a computer.

The aim of the game is to get as close to a score of 21 as you can, without going over 21. If your score goes over 21 then you lose.

The player's score starts at 0.

For each turn:

- two dice (each numbered from 1 to 6) are rolled
- the total of the two dice rolls is added to the player's score
- the value of each dice and the player's new total score is output
- if the current score is less than 21, the player is asked if they would like to roll the dice again: if the player says yes, they get another turn; otherwise, the game ends.

At the end of the game, the program should work as follows:

- if the final score is 21, output a message to say the player has won
- if the final score is greater than 21, output a message to say the player has lost
- if the final score is less than 21, the program generates a random number between 15 and 21 inclusive:
 - if this random number is greater than the player's final score, output a message to say the player has lost
 - otherwise, output a message to say the player has won.

Figure 17 shows the output of a program that plays this dice game.

Figure 17

```
Roll 1: 1
Roll 2: 4
Current score: 5
Would you like to roll again? yes

Roll 1: 1
Roll 2: 6
Current score: 12
Would you like to roll again? yes

Roll 1: 1
Roll 2: 2
Current score: 15
Would you like to roll again? yes

Roll 1: 6
Roll 2: 1
Current score: 22
You lost!
```

Write a C# program to simulate this game.

The first line has been written for you in the answer grid.

The dice rolls are carried out by the program generating random numbers between 1 and 6. You will need to use the C# function `r.Next(a, b)` which generates a random integer in the range `a` to `b` starting at `a` but finishing one before `b`.

You should use meaningful variable name(s) and C# syntax in your answer.

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

[11 marks]

[illegible]

[illegible]

9. No.1(1.2_1.4)

Figure 1 shows an algorithm, represented using pseudo-code, which assigns a different value to four variables.

Figure 1

```
country ← 'United States of America'  
state ← 'California'  
city ← 'San Francisco'  
landmark ← 'Alcatraz Island'
```

0 1 . 2 The variable `x` is assigned a value using the statement:

```
x ← LEN(state)
```

Using **Figure 1**, what is the value of `x`?

Shade one lozenge.

[1 mark]

- | | | |
|----------|----|-----------------------|
| A | 1 | ☐ |
| B | 5 | ☐ |
| C | 10 | ☐ |
| D | 12 | ☐ |

0 1 . 3

What is the result of concatenating the contents of the variables `city` and `landmark` in **Figure 1**?

Shade one lozenge.

[1 mark]

- | | | |
|----------|---|-----------------------|
| A | <code>San Francisco Alcatraz Island</code> | ☐ |
| B | <code>San Francisco,Alcatraz Island</code> | ☐ |
| C | <code>San Francisco, Alcatraz Island</code> | ☐ |
| D | <code>San FranciscoAlcatraz Island</code> | ☐ |

0 1 . 4

The subroutine `SUBSTRING` extracts characters from a given string.

For example, `SUBSTRING(3, 5, 'Computing')` would return `put`

The variable `y` is assigned a value using the statement:

$$y \leftarrow \text{SUBSTRING}(4, 7, \text{landmark})$$

Using **Figure 1**, what is the value of `y`?

Shade one lozenge.

[1 mark]

- | | | |
|----------|-------------------|-----------------------|
| A | <code>Alca</code> | ☐ |
| B | <code>Atra</code> | ☐ |
| C | <code>land</code> | ☐ |
| D | <code>traz</code> | ☐ |

Figure 1 has been included again below.

Figure 1

```
country ← 'United States of America'  
state ← 'California'  
city ← 'San Francisco'  
landmark ← 'Alcatraz Island'
```

0 1 . 5 The subroutine `POSITION` finds the first position of a character in a string.

For example, `POSITION('Computing', 'p')` would return 3

The variable `z` is assigned a value using the statement:

```
z ← POSITION(landmark, 't')
```

Using **Figure 1**, what value is assigned to `z`?

Shade one lozenge.

[1 mark]

- | | | |
|----------|----|-----------------------|
| A | -1 | ☐ |
| B | 3 | ☐ |
| C | 4 | ☐ |
| D | 5 | ☐ |

10. No.3Explain **one** advantage of the structured approach to programming.**[2 marks]**

11. No.4

0 4

Figure 3 shows a program written in Python that calculates the area of a rectangle or the volume of a box from the user inputs.

Figure 3

```
def calculate(width, length, height):
    if height == -1:
        return width * length
    else:
        return width * length * height

numOne = int(input("Enter width: "))
numTwo = int(input("Enter length: "))
numThree = int(input("Enter height, -1 to ignore: "))

answer = calculate(numOne, numTwo, numThree)

if numThree == -1:
    print(f"Area {answer}")
else:
    print(f"Volume {answer}")
```

0 4 . 1

Complete the trace table using the program in **Figure 3**.

[3 marks]

numOne	numTwo	numThree	Final output
5	6	-1	
10	4	0	
3	5	10	

0 4 . 2

Describe **one** way that the program in **Figure 3** could be made more robust.

[1 mark]

12. No.7

A theme park charges £15 per person for a daily ticket. If there are six or more people in a group, the group is given a £5 discount.

Write a Python program to calculate the total charge for a group of people visiting the theme park.

The program must:

- get the user to enter the number of people in a group
- calculate the total charge by:
 - charging £15 per person
 - reducing the total charge by £5 if there are six or more people
- output the total charge.

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

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

[6 marks]

13. No.12

1 2 . 1

Figure 11 shows a binary search algorithm that has been programmed in Python.

Figure 11

```

animals = ["cat", "dog", "hippo", "llama", "ox",
"rat", "tiger", "wolf"]
animalToFind = input("What animal would you like to
find? ")
validAnimal = False
start = 0
finish = len(animals) - 1
while validAnimal == False and start <= finish:
    mid = (start + finish) // 2
    if animals[mid] == animalToFind:
        validAnimal = True
    elif animalToFind > animals[mid]:
        start = mid + 1
    else:
        finish = mid - 1
print(validAnimal)

```

Complete the trace table for the program in Figure 11 if the user input is `wolf`

Part of the table has already been filled in.

You may not need to use all the rows in the table.

[4 marks]

animalToFind	validAnimal	start	finish	mid
wolf	False	0	7	3

1 2 . 2 Figure 12 shows a line of Python code that creates a list of fruit names.

Figure 12

```
fruits = ["banana", "apple", "orange", "pear",
          "grape", "pineapple"]
```

Extend the program in **Figure 12**. Your answer must be written in Python.

The program should get the user to enter a word and perform a **linear** search on the list `fruits` to find if the word is in the list or not.

The program should:

- ask the user what word they would like to find
- output the message `True` if the word is found
- output the message `False` if the word is not found.

You must write your own linear search routine and **not** use any built-in search function available in Python.

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

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

[7 marks]

<pre>fruits = ["banana", "apple", "orange", "pear", "grape", "pineapple"]</pre>				

1 **2** **3** State why a binary search cannot be used on the list fruits

[1 mark]

1 2 . 4

Figure 13 shows an algorithm, represented using pseudo-code, that should display currency names in reverse alphabetical order, starting with yen.

There are errors in the logic of the algorithm.

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

Figure 13

```

1  SUBROUTINE diffCurrencies(currencies)
      currencies ← ['baht', 'dollar', 'euro',
2      'koruna', 'lira', 'rand',
      'rupee', 'yen']
3      RETURN currencies[x]
4  ENDSUBROUTINE
5
6  FOR i ← 8 TO 0 STEP 1
7      OUTPUT(diffCurrencies(i))
8  ENDFOR

```

Rewrite line 1 and line 6 from **Figure 13** to make the algorithm work as intended.

[3 marks]

Line 1 _____

Line 6 _____

14. No.13

A programmer is writing a game. The game uses a 3 x 3 grid containing nine squares.

Figure 14

	A	B	C
1			
2			
3			X

In the game, a square on the grid is referred to by a letter and a number. For example, square C3 in **Figure 14** contains an X.

Figure 15 shows part of a Python program that checks the grid reference entered by a player.

The grid reference is valid if:

- there are exactly two characters
- the first character entered is A, B or C
- the second character entered is 1, 2 or 3.

Figure 15

```

check = False
while check == False:
    square = ""
    while len(square) != 2:
        square = input("Enter grid reference (eg C2): ")
        square = square.upper()

```

The Python function `upper ()` converts letters into uppercase, eg `b1` would be converted to `B1`

Extend the program from **Figure 15** so it completes the other checks needed to make sure a valid grid reference is entered.

Your extended program must:

- use the variable `check`
- repeat the following steps until a valid grid reference is entered:
 - get the user to enter a grid reference
 - output an appropriate message if the grid reference entered is not valid.

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

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

[6 marks]

```
check = False
```

```
while check == False:
```

	square = ""
--	-------------

while len(square) != 2:

	<pre>square = input("Enter grid reference (eg C2): ")</pre>
--	---

	<code>square = square.upper()</code>
--	--------------------------------------

[illegible]

15. No.14

50 students have voted for the music genre they like best.

Figure 16 shows an incomplete algorithm, represented using pseudo-code, designed to output the highest or lowest results of the vote.

The programmer has used a two-dimensional array called `results` to store the genre and the number of votes for each genre.

Parts of the algorithm are missing and have been replaced with the labels **L1** to **L3**.

Figure 16

```

SUBROUTINE showResults(method, numberOfGenres)
    results ← [['Pop', 'Post-Punk', 'Techno', 'Metal',
                'Dance'], ['7', '19', '14', '1', '9']]
    pos ← 0
    high ← -1
    IF method = 'HIGHEST' THEN
        FOR i ← 0 TO numberOfGenres - 1
            votes ← STRING_TO_INT(results[L1][i])
            IF votes > high THEN
                high ← votes
                pos ← L2
            ENDIF
        ENDFOR
    ELSE
        OUTPUT 'not yet working'
    ENDIF
    IF high ≠ -1 THEN
        OUTPUT results[0][pos], ' with ', results[1][pos]
    ENDIF
ENDSUBROUTINE

OUTPUT 'Show the genre with the HIGHEST or LOWEST number
of votes? '
method ← USERINPUT
showResults(L3, 5)

```

State what should be written in place of the labels **L1** to **L3** in the algorithm in **Figure 16**.

[3 marks]

L1 _____

L2 _____

L3 _____

16. No.15

A group of people have a meal in a restaurant. Instead of one person paying for the whole meal, each person will pay for what they eat.

Write a Python program that asks each person in the group how much they are paying towards the meal and works out when the bill is fully paid. Each person can pay a different amount.

The program should:

- get the user to enter the total amount of the bill
- get a person to enter how much they are paying towards the bill
- subtract the amount entered from the bill:
 - if the amount left to pay is more than 0, output how much is left to pay and repeat until the amount left to pay is 0 or less
 - if the amount left to pay is 0, then output the message `Bill paid`
 - if the amount left to pay is less than 0, then output the message `Tip is` and the difference between the amount left to pay and 0

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

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

[8 marks]

17. No.16

Question 16 is about a dice game played against a computer.

The aim of the game is to get as close to a score of 21 as you can, without going over 21. If your score goes over 21 then you lose.

The player's score starts at 0.

For each turn:

- two dice (each numbered from 1 to 6) are rolled
- the total of the two dice rolls is added to the player's score
- the value of each dice and the player's new total is output
- if the current score is less than 21, the player is asked if they would like to roll the dice again: if the player says yes, they get another turn; otherwise, the game ends.

At the end of the game, the program should work as follows:

- if the final score is 21, output a message to say the player has won
- if the final score is greater than 21, output a message to say the player has lost
- if the final score is less than 21, the program generates a random number between 15 and 21 inclusive:
 - if this random number is greater than the player's final score, output a message to say the player has lost
 - otherwise, output a message to say the player has won.

Figure 17 shows the output of a program that plays this dice game.

Figure 17

```
Roll 1: 1
Roll 2: 4
Current score: 5
Would you like to roll again? yes

Roll 1: 1
Roll 2: 6
Current score: 12
Would you like to roll again? yes

Roll 1: 1
Roll 2: 2
Current score: 15
Would you like to roll again? yes

Roll 1: 6
Roll 2: 1
Current score: 22
You lost!
```

Write a Python program to simulate this game.

The first line has been written for you in the answer grid.

The dice rolls are carried out by the program generating random numbers between 1 and 6. You will need to use the Python function `random.randrange(a, b)` which generates a random integer in the range `a` to `b` starting at `a` but finishing one before `b`.

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

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

[11 marks]

import random				

18. No.1(1.2_1.5)

Figure 1 shows an algorithm, represented using pseudo-code, which assigns a different value to four variables.

Figure 1

```
country ← 'United States of America'  
state ← 'California'  
city ← 'San Francisco'  
landmark ← 'Alcatraz Island'
```

☐ 0 ☐ 1 ☒ 2 The variable x is assigned a value using the statement:

```
 $x \leftarrow \text{LEN}(\text{state})$ 
```

Using **Figure 1**, what is the value of x ?

Shade one lozenge.

[1 mark]

A 1

☐

B 5

☐

C 10

☐

D 12

☐

0 1 . 3

What is the result of concatenating the contents of the variables `city` and `landmark` in **Figure 1**?

Shade one lozenge.

[1 mark]

- | | | |
|----------|---|-----------------------|
| A | <code>San Francisco Alcatraz Island</code> | ☐ |
| B | <code>San Francisco,Alcatraz Island</code> | ☐ |
| C | <code>San Francisco, Alcatraz Island</code> | ☐ |
| D | <code>San FranciscoAlcatraz Island</code> | ☐ |

0 1 . 4

The subroutine `SUBSTRING` extracts characters from a given string.

For example, `SUBSTRING(3, 5, 'Computing')` would return `put`

The variable `y` is assigned a value using the statement:

$$y \leftarrow \text{SUBSTRING}(4, 7, \text{landmark})$$

Using **Figure 1**, what is the value of `y`?

Shade one lozenge.

[1 mark]

- | | | |
|----------|-------------------|-----------------------|
| A | <code>Alca</code> | ☐ |
| B | <code>Atra</code> | ☐ |
| C | <code>land</code> | ☐ |
| D | <code>traz</code> | ☐ |

Figure 1 has been included again below.

Figure 1

```
country ← 'United States of America'  
state ← 'California'  
city ← 'San Francisco'  
landmark ← 'Alcatraz Island'
```

0 1 . 5 The subroutine `POSITION` finds the first position of a character in a string.

For example, `POSITION('Computing', 'p')` would return 3

The variable `z` is assigned a value using the statement:

```
z ← POSITION(landmark, 't')
```

Using **Figure 1**, what value is assigned to `z`?

Shade one lozenge.

[1 mark]

- | | | |
|----------|----|--------------------------|
| A | -1 | ☐ |
| B | 3 | ☐ |
| C | 4 | ☐ |
| D | 5 | ☐ |

19. No.2(2.3)

Figure 2 shows an algorithm that uses integer division which has been represented using pseudo-code.

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

Figure 2

```

1  again ← True
2  WHILE again = True
3      a ← USERINPUT
4      IF a > 0 THEN
5          counter ← 0
6          WHILE a > 0
7              a ← a DIV 3
8              counter ← counter + 1
9          ENDWHILE
10     ELSE
11         again ← False
12     ENDIF
13     OUTPUT a
14 ENDWHILE

```

Integer division is the number of times one integer divides into another, with the remainder ignored.

For example:

- 14 DIV 5 evaluates to 2
- 25 DIV 3 evaluates to 8

0 2 . 3

In the algorithm in Figure 2, what is the largest possible value of the variable counter when the user input is 36?

Shade one lozenge.

[1 mark]

- | | | |
|----------|---|-----------------------|
| A | 0 | ☐ |
| B | 2 | ☐ |
| C | 4 | ☐ |
| D | 5 | ☐ |

20. No.3Explain **one** advantage of the structured approach to programming.**[2 marks]**

21. No.4**Figure 3** shows a program written in VB.NET that calculates the area of a rectangle or the volume of a box from the user inputs.**Figure 3**

```
Function calculate(width As Integer, length As Integer,
height As Integer) As Integer
    If height = -1 Then
        Return width * length
    Else
        Return width * length * height
    End If
End Function

Sub Main()
    Dim numOne, numTwo, numThree, answer As Integer
    Console.Write("Enter width: ")
    numOne = Console.ReadLine()
    Console.Write("Enter length: ")
    numTwo = Console.ReadLine()
    Console.Write("Enter height, -1 to ignore:")
    numThree = Console.ReadLine()

    answer = calculate(numOne, numTwo, numThree)

    If numThree = -1 Then
        Console.WriteLine($"Area {answer}")
    Else
        Console.WriteLine($"Volume {answer}")
    End If
End Sub
```

0 4 . 1

Complete the trace table using the program in **Figure 3**.

[3 marks]

numOne	numTwo	numThree	Final output
5	6	-1	
10	4	0	
3	5	10	

0 4 . 2

Describe **one** way that the program in **Figure 3** could be made more robust.

[1 mark]

22. No.7

A theme park charges £15 per person for a daily ticket. If there are six or more people in a group, the group is given a £5 discount.

Write a VB.NET program to calculate the total charge for a group of people visiting the theme park.

The program must:

- get the user to enter the number of people in a group
- calculate the total charge by:
 - charging £15 per person
 - reducing the total charge by £5 if there are six or more people
- output the total charge.

You **should** use meaningful variable name(s) and VB.NET syntax in your answer.

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

[6 marks]

23. No.12

Figure 11 shows a binary search algorithm that has been programmed in VB.NET.

Figure 11

```
Dim animals As string() = {"cat", "dog", "hippo",  
"llama", "ox", "rat", "tiger", "wolf"}  
Console.Write("What animal would you like to find? ")  
Dim animalToFind As string = Console.ReadLine()  
Dim validAnimal As Boolean = False  
Dim start As Integer = 0  
Dim finish As Integer = animals.Length - 1  
While validAnimal = False And start <= finish  
    Dim mid As Integer = (start + finish) \ 2  
    If animals(mid) = animalToFind Then  
        validAnimal = True  
    ElseIf animalToFind > animals(mid) Then  
        start = mid + 1  
    Else  
        finish = mid - 1  
    End If  
End While  
Console.WriteLine(validAnimal)
```

Complete the trace table for the program in **Figure 11** if the user input is `wolf`

Part of the table has already been filled in.

You may not need to use all the rows in the table.

[4 marks]

<code>animalToFind</code>	<code>validAnimal</code>	<code>start</code>	<code>finish</code>	<code>mid</code>
<code>wolf</code>	<code>False</code>	<code>0</code>	<code>7</code>	<code>3</code>

1 2 . 2

Figure 12 shows a line of VB.NET code that creates an array of fruit names.

Figure 12

```
Dim fruits() As String = {"banana", "apple", "orange",  
                          "pear", "grape", "pineapple"}
```

Extend the program in **Figure 12**. Your answer must be written in VB.NET.

The program should get the user to enter a word and perform a **linear** search on the array `fruits` to find if the word is in the array or not.

The program should:

- ask the user what word they would like to find
- output the message `True` if the word is found
- output the message `False` if the word is not found.

You must write your own linear search routine and **not** use any built-in search function available in VB.NET.

You **should** use meaningful variable name(s) and VB.NET syntax in your answer.

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

[7 marks]

Dim fruits() As String = {"banana", "apple", "orange", "pear", "grape", "pineapple"}				

1 2 . 3 State why a binary search cannot be used on the array `fruits`

[1 mark]

1 2 . 4

Figure 13 shows an algorithm, represented using pseudo-code, that should display currency names in reverse alphabetical order, starting with yen.

There are errors in the logic of the algorithm.

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

Figure 13

```

1  SUBROUTINE diffCurrencies(currencies)
    currencies ← ['baht', 'dollar', 'euro',
2      'koruna', 'lira', 'rand',
      'rupee', 'yen']
3  RETURN currencies[x]
4  ENDSUBROUTINE
5
6  FOR i ← 8 TO 0 STEP 1
7      OUTPUT(diffCurrencies(i))
8  ENDFOR

```

Rewrite line 1 and line 6 from **Figure 13** to make the algorithm work as intended.

[3 marks]

Line 1 _____

Line 6 _____

24. No.13

A programmer is writing a game. The game uses a 3 x 3 grid containing nine squares.

Figure 14

	A	B	C
1			
2			
3			X

In the game, a square on the grid is referred to by a letter and a number. For example, square C3 in **Figure 14** contains an X.

Figure 15 shows part of a VB.NET program that checks the grid reference entered by a player.

The grid reference is valid if:

- there are exactly two characters
- the first character entered is A, B or C
- the second character entered is 1, 2 or 3.

Figure 15

```
Dim check As Boolean = False
While check = False
    Dim square As String = ""
    While square.Length <> 2
        Console.Write("Enter grid reference (eg C2): ")
        square = Console.ReadLine()
        square = square.ToUpper()
    End While
End While
```

The VB.NET function `ToUpper ( )` converts letters into uppercase, eg `b1` would be converted to `B1`

Extend the program from **Figure 15** so it completes the other checks needed to make sure a valid grid reference is entered.

Your extended program must:

- use the variable `check`
- repeat the following steps until a valid grid reference is entered:
 - get the user to enter a grid reference
 - output an appropriate message if the grid reference entered is not valid.

You **should** use meaningful variable name(s) and VB.NET syntax in your answer.

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

[6 marks]

```
Dim check As Integer = False
```

```
While check = False
```

```
    Dim square As String = ""
```

```
    While square.Length <> 2
```

```
        Console.Write("Enter grid reference (eg C2): ")
```

```
        square = Console.ReadLine()
```

```
        square = square.ToUpper()
```

```
    End While
```

```
End While
```

25. No.14

50 students have voted for the music genre they like best.

Figure 16 shows an **incomplete** algorithm, represented using pseudo-code, designed to output the highest or lowest results of the vote.

The programmer has used a two-dimensional array called `results` to store the genre and the number of votes for each genre.

Parts of the algorithm are missing and have been replaced with the labels **L1** to **L3**.

Figure 16

```

SUBROUTINE showResults(method, numberOfGenres)
    results ← [['Pop', 'Post-Punk', 'Techno', 'Metal',
                'Dance'], ['7', '19', '14', '1', '9']]
    pos ← 0
    high ← -1
    IF method = 'HIGHEST' THEN
        FOR i ← 0 TO numberOfGenres - 1
            Votes ← STRING_TO_INT(results[L1][i])
            IF votes > high THEN
                high ← votes
                pos ← L2
            ENDIF
        ENDFOR
    ELSE
        OUTPUT 'not yet working'
    ENDIF
    IF high ≠ -1 THEN
        OUTPUT results[0][pos], ' with ', results[1][pos]
    ENDIF
ENDSUBROUTINE

OUTPUT 'Show the genre with the HIGHEST or LOWEST number
of votes? '
method ← USERINPUT
showResults(L3, 5)

```

State what should be written in place of the labels **L1** to **L3** in the algorithm in **Figure 16**.

[3 marks]

L1 _____

L2 _____

L3 _____

26. No.15

A group of people have a meal in a restaurant. Instead of one person paying for the whole meal, each person will pay for what they eat.

Write a VB.NET program that asks each person in the group how much they are paying towards the meal and works out when the bill is fully paid. Each person can pay a different amount.

The program should:

- get the user to enter the total amount of the bill
- get a person to enter how much they are paying towards the bill
- subtract the amount entered from the bill:
 - if the amount left to pay is more than 0, output how much is left to pay and repeat until the amount left to pay is 0 or less
 - if the amount left to pay is 0, then output the message `Bill paid`
 - if the amount left to pay is less than 0, then output the message `Tip is` and the difference between the amount left to pay and 0

You **should** use meaningful variable name(s) and VB.NET syntax in your answer.

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

[8 marks]

27. No.16

Question 16 is about a dice game played against a computer.

The aim of the game is to get as close to a score of 21 as you can, without going over 21. If your score goes over 21 then you lose.

The player's score starts at 0.

For each turn:

- two dice (each numbered from 1 to 6) are rolled
- the total of the two dice rolls is added to the player's score
- the value of each dice and the player's new total score is output
- if the current score is less than 21, the player is asked if they would like to roll the dice again: if the player says yes, they get another turn; otherwise, the game ends.

At the end of the game, the program should work as follows:

- if the final score is 21, output a message to say the player has won
- if the final score is greater than 21, output a message to say the player has lost
- if the final score is less than 21, the program generates a random number between 15 and 21 inclusive:
 - if this random number is greater than the player's final score, output a message to say the player has lost
 - otherwise, output a message to say the player has won.

Figure 17 shows the output of a program that plays this dice game.

Figure 17

```
Roll 1: 1
Roll 2: 4
Current score: 5
Would you like to roll again? yes

Roll 1: 1
Roll 2: 6
Current score: 12
Would you like to roll again? yes

Roll 1: 1
Roll 2: 2
Current score: 15
Would you like to roll again? yes

Roll 1: 6
Roll 2: 1
Current score: 22
You lost!
```

Write a VB.NET program to simulate this game.

The first line has been written for you in the answer grid.

The dice rolls are carried out by the program generating random numbers between 1 and 6. You will need to use the VB.NET function `r.Next(a, b)` which generates a random integer in the range `a` to `b` starting at `a` but finishing one before `b`.

You should use meaningful variable name(s) and VB.NET syntax in your answer.

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

[11 marks]

```
Dim r As Random = New Random()
```

[illegible]

