NWU®

NAME & SURNAME: ..... Student Nr: .....

1. Question 1 *(Short questions)* **[10]**

1.1. Provide the code needed to write data to a text file named "Output.txt" in the project folder, using an existing StreamWriter object called "saver".

_____ **saver = File.CreateText(Output.txt)**_____ (2)

1.2. Which property of a control needs to be changed from private to public to allow it to be accessed from another form?

_____ **Modifiers** _____ (1)

1.3. When working with files in C#, which method can be called to verify that a file operation was successful?

_____ **The ShowDialog method**_____ (1)

1.4. When creating an object of the Random class, you can specify a _____**seed value**_____ which determines the point from which the object will begin generating random numbers. If you do not specify a value, it uses the system clock. (1)

1.5. What is the difference between passing an argument by reference and passing an argument by value?

By reference: **When passing an argument by value, the method receives a <u>copy of the argument's value</u>. Changes made to the parameter inside the method do not affect the original variable outside the method**

_____

By value: **When passing an argument by reference, the method receives a <u>reference to the actual variable, not a copy</u>. Changes made to the parameter inside the method directly affect the original variable.**

<u>One mark for answering both correctly</u>

_____ (3)

1.6. How many times will the loop below iterate? _____**5 times**_____(1)

```
int count = 3;
decimal cost = 500m;
while (count < 8)
{
        cost += 150;
        count++;
}
```

1.7. What will be the final value of cost be, after the loop completes? _____**1250m**_____ (1)

2. Question 2 (Short Programming Questions) [8]

2.1. Consider the method header below, name the different parts in the method header.

```
[2.1.1]  [2.1.2]  [2.1.3][2.1.4]
{
        MessageBox.Show("The amount of tax is: " + tax.ToString("P2"));
}
```

2.1.1. (_____**Access modifier**_____)
2.1.2. (_____**Return type**_____)
2.1.3. (_____**Method name**_____)
2.1.4. (**Parentheses or parameter(s) or parameter list**) (4)

2.2. Consider question 2.1.4 above. Provide an example of code that can be used there, where the method accepts only one default argument. Tip: Look at the method body.

_____**(decimal tax = 0.05m) or similar, but variable name must be tax**_____(1)

2.3. What would be the output of the given code.

```
int val  =  51;
```

Messagebox.Show(val); _____**51**_____

Messagebox.Show(++val); _____**52**_____

Messagebox.Show(val ++); _____**52**_____

Messagebox.Show(val); _____**53**_____ **Half mark each** (2)

2.4. What is the difference between the for and while loop?

_____**For is counter based and for a known number of iterations**_____

_____**While is for while a certain condition remains true.**_____ **Half mark each** (1)

3. Question 3 (Programming Questions) [20]

3.1. Write a method declaration for a method, which can be used in any class that calls it, as follows:

- The method name must be called **ShowRandomValue**.
- The purpose of the method is to display a randomly generated number in a message box.
- The randomly generated number must be between 10 and 100, including 100.
- The method is declared in one class and called from another class.
- The object, through which the method will be called, is named **dataHandler** (already created).

Write the method, and then write the statement to call this method using the **dataHandler** object.

Method Declaration:

```
public void ShowRandomValue()
{
    // Create a Random object
    Random rand = new Random();

    // Generate a random number between 10 and 100 (inclusive)
    int randomNumber = rand.Next(10, 101); // Upper bound is exclusive

    // Display the random number in a message box
    MessageBox.Show("Random Number: " + randomNumber.ToString());
}
```

Calling the Method:

```
dataHandler.ShowRandomValue();
```

- 1 mark for using the **public** access modifier, indicating the method can be accessed from other classes.
- 1 mark for correctly naming the method **ShowRandomValue**.
- 2 marks for generating a random number between 10 and 100 (using **rand.Next(10, 101)**).
- 2 marks for correctly displaying the number in a **MessageBox**.
- 1 mark for providing the correct call to the method using the **dataHandler** object.
- 1 mark for correct syntax and structure throughout the method and call.

_____ (8)

3.2. Write a method declaration for a method, to be used only within the class where it is declared, as follows:

- The method name must be **CalculateBMIReference**.
- The following arguments should be passed to the method: **weightKg**, **heightM**, **lastName**.
- The method should calculate the **BMI (weightKg / (heightM * heightM))** and concatenate it with the passed **lastName**.
- The concatenated result must be returned to the part of the program where the method was called.
- Once returned to the calling part (outside of the method), the concatenated result must be stored in a variable named **bmiReference** of the appropriate data type.

```
private string CalculateBMIReference(decimal weightKg, decimal heightM, string lastName)
{
    // Calculate BMI
    decimal bmi = weightKg / (heightM * heightM);

    // Concatenate BMI with lastName
    string result = lastName + ": " + bmi.ToString("F2");

    // Return the concatenated result
    return result;
}
```

- 1 mark for using the **private** access modifier, indicating the method is only accessible within the class.
- 1 mark for the correct method name **CalculateBMIReference**.
- 2 marks for specifying the correct argument types: **weightKg** (decimal), **heightM** (decimal), and **lastName** (string).
- 2 marks for calculating BMI with the formula **weightKg / (heightM * heightM)** and concatenating it with **lastName**.
- 1 mark for returning the concatenated result.
- 1 mark for correctly storing the returned value in a **bmiReference** variable in the calling part.

_____ (8)

3.3. Write the C# code that allows a programmer to create an object named calculator of a Form called FinanceForm.cs.

✓ ✓ ✓

_____ **FinanceForm calculator = new FinanceForm();**_____ (3)

3.4. Fill in the missing code that writes and displays the following output in a text file (Note: only one word may be written to the file at a time). Hence, 5 statements are required.

Apple Banana Cherry
Grape Orange
Watermelon

StreamWriter writeThingy = File.CreateText(SaveFileDialog1.FileName);

writeThingy. __ **Write**_____ ("Apple");

writeThingy. __ **Write**_____ ("Banana");

writeThingy. __ **WriteLine**_____ ("Cherry");

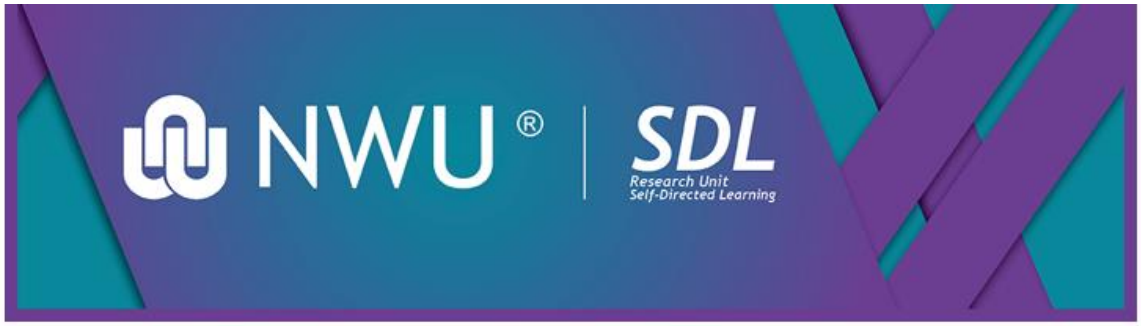
writeThingy. __ **Write**_____ ("Grape");

writeThingy. __ **WriteLine**_____ ("Orange");

writeThingy. __ **WriteLine**_____ ("Watermelon");

Half mark each (3)

. . . . . TOTAL / *TOTAAL*: 40



Licence:

Disclaimer:

This resource is one of a set of resources primarily generated by students. While we strive to maintain a high standard of quality, we cannot guarantee the accuracy, reliability, or completeness of the content provided. Users are encouraged to critically evaluate the materials and adopt them at their own discretion. Neither the founders nor North-West University assume responsibility for any errors, omissions, or for the results obtained from the use of this information.