Programming for Computing

Assignment 1 (40% of Module)

"The Money Credit Counter"

This assignment consists of three main elements, the first two of which should be submitted in e-format via Blackboard, as stated in the "Hand In" section of this document. The third element will be viewed by the lecturer after the hand in date. These main elements are listed below and are described in more detail in the remainder of this document, along with some hints and source code to assist you:

1. Creating the software application
2. Written work
3. GitHub Repository (version control of your source code)

Please Note.

This is an individual assignment and **NOT** a group task. Should you require additional information, please contact me using the following:

E : P.O'Neill@shu.ac.uk
T : 0114 225 6990

In addition, before you start to write any source code, **PLEASE SEE**, "Section 4 Written Work" and the subsection, "Version Control" of this document.

Creating the Software Application

Write a program in C#, within the IDE Visual Studio 2017, to emulate a simple "Money Credit Counter". Mainly for this task you will need to use 'If' statements, e.g., to validate the status of the Credit Counter etc.. Also, you will need to validate that the "**Please Enter the Cost per Credit**", is set and that the Credit value reflects the money entered.

Once you have set the "**Please Enter the Cost per Credit**" field and you click using the left mouse button on any of the coins shown in the *GroupBox* "**Coins**", the value associated with that coin in the "**Coins Entered**" *GroupBox* should be incremented, reflecting the number of times that coin has been selected.

In addition, with each selection of a coin, the "**Total Value**" shown in 'pence only' and 'pounds and pence' should increase by the value of the coin selected and be displayed in their respective formats. Once the "**Total Value**" exceeds the 'Cost per Credit', the textbox to the right of "**You Presently Have This Number Of Credits**" should display the number of credits purchased, which is the total money entered, divided by the cost per credit.

Section 1 Basic Credit Counter – (15 marks)

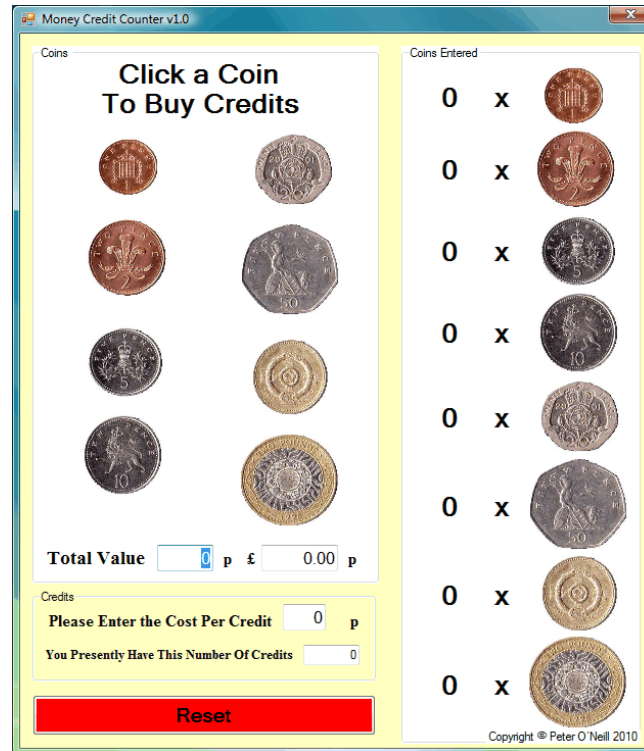| No.	Task	Mark
1.	Create a Graphical User Interface (GUI) that contains all of the elements shown within the main window of Figure 1, see figure below:	**15**

Figure 1 The Money Credit Counter Main Window.

Section 2 Basic Credit Counter's Functionality – (50 marks)

Write the required source code, to create the functionality as stated in the following tasks:

| No.	Task	Mark
2.	On selecting any of the coins in the "Coins" GroupBox , when the TextBox to the right of "Please Enter the Cost Per Credit" shows ' ' or '0', then have that application show a MessageBox, stating "You have not set the Cost of a Credit".	5
3.	On selecting any of the coins in the "Coins" GroupBox, when the TextBox to the right of "Please Enter the Cost Per Credit" shows a value, then the application increment the same coin's value, as shown in figure 2, (on the following page) e.g., 1 p selected twice and 2 p selected once:	5
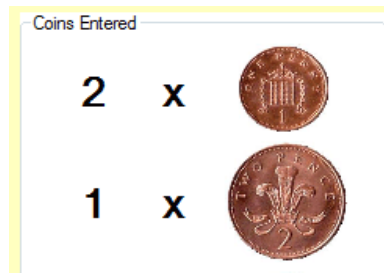
Figure 2 Coins Entered GroupBox,
Showing number of times the respective coins were selected.

4. On selecting any of the coins in the "**Coins**" GroupBox, have the application accumulate the total value of money selected and enter the result into the first TextBox to the right of the Label displaying "**Total Value**", shown here in the "Credits" GroupBox :
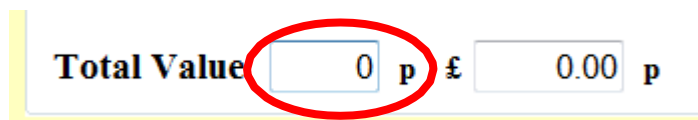
10

Figure 3 Total Value in pence

5. On selecting any of the coins in the "**Coins**" GroupBox, have the application accumulate the total value of money selected, then format this value into pp.nn and enter the result into the second TextBox to the right of the Label displaying "**Total Value**", shown here:
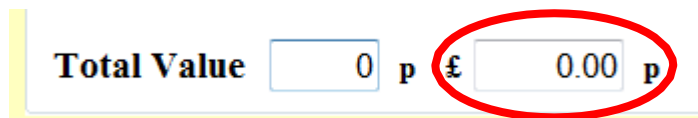

Figure 4 Pounds and pence Format

20

Please Note

pp is the number of pounds and nn is the number of pence.

6. On selecting any of the coins in the "**Coins**" GroupBox, have the application calculate the number of credits purchased by dividing the Total Value in Pennies (see Figure 3) by the value displayed in "Please Enter the Cost Per Credit" and show the result in the TextBox to the right of "**You Presently Have This Number Of Credits**".

10

Section 3 The Interface - (10 marks)

No.	Task	Mark
7.	Have the application (e.g., Figure 1, Money Credit Counter Main Window. on page 2 of this document) always appear in the center of the screen on execution.	3

8. Give the Money Credit Counter Main Window e.g., Figure 1, on page 2   of this document display the Title, "Money Credit Counter v2.0".   3

9. Have the Application (e.g., Figure 1, Money Credit Counter Main Window. on page 2 of this document) only display the Close widget and NOT the MaximizeBox property and the MinimizeBox property as shown here:
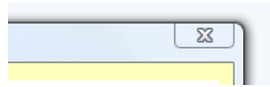
4

Figure 5 Main Window showing Close widget only

Written Work

Section 4 Writing - (10 marks)

In two pages or less, discuss the two following areas:

No.	Task	Mark

10. Explain in everyday language what each line of the following source code does and the overall functionality created.

```
string attempt;

attempt = My_Dialogs.InputBox("Enter your password");
while (attempt != "basic")
{
    MessageBox.Show("Wrong! - try again.");
    attempt = InputBox("Enter your password");
}

MessageBox.Show("Welcome to the program!");
```

10

The answer to the above question is not to exceed one page of text. E.g., I am looking for quality not quantity.

Version Control Using GitHub- (15 marks)

GitHub is widely used by software houses, particularly were a considerable number of software developers are collaborating on the same project, (in most cases these days), but it can also be useful when one person is developing as you can revert to previous versions.

No.	Task	Mark

11. In this section we would like you to keep GitHub of your source code while you are writing it. A minimum of 5 "Commits" to the GitHub is required. You will need to invite me, using this email addresses acespo@my.shu.ac.uk.

10

(Use Google and or Youtube searching for GitHub in Visual Studio 2017.)

12. Include throughout your Source Code comments to assist the reader in understanding your code structure and logic

5

Handing In via Blackboard

You are to Hand in a:

- Word Document.
- Zip file.

The Word Document will contain the following:

1. Your answers to the "Written Work".

Zip files will contain:

- Your Visual Studio 2017 Project directory, which should include all the source code in C# and the executable version of your application.

Your Zip file and Word Document should be titled <Your Full Name Student Number.zip>, e.g., in my case, "**O'Neill P 209244.zip**".

In the Word Document and Source Code files you should have, **<Your Full Name and Student Number Assignment Designation>** in the Header of each page. The 'Footer' (Word Document Only) should have 'Programming for Computing' and the **page numbers**.

In Windows, you can copy a shot of the active window by pressing <Shift> plus <Prt-sc> keys

This is an all-electronic hand-in. Work will **NOT** be marked if handed-in on paper or disc.

Cheating. This is an individual assignment - I will report any cheating to the examinations board.

Hand In Date: Date and Time stated on Blackboard

You will hand in the required files on, or before the date and time stated on Black Board. I have provided instructions on how to submit your Word document and Zip file via Blackboard.

In order to pass this module you are required to score a minimum of 40% in each assignment. Failing to accomplish this mark can result in you being required to, 'Resit' a single assignment, or undertake the complete year and in some cases, being asked to leave the University. (Please see University's Procedures for more information.)

Hints etc

Ask if you get stuck - I won't tell you the answer, but I will guide!

E : P.O'Neill@shu.ac.uk
T : 0114 225 6990

Break the application into smaller parts. The 'Marking' schema above should give you some ideas.

In addition, create the functionality for ONE coin first. You should find that the remaining coins resemble the first one you do.

Useful Source Code

Formatting the currency:

```
= String.Format("{0:n2}",  Convert.ToDouble(txt_Total_In_Pence.Text);
```