

IST 256
Lab Week 3, Day 2 – September 16, 2009

1. Understanding For Loops – showing the variables of a program

For this problem, you are not to put any part of the program into Visual Studio. You are to write your answers on this lab sheet. (Think of this as a practice exam question.)

A. Consider the following program for a button:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim counter, sum As Integer
    counter = 0
    sum = 0

    ' add integers from 1 to 7
    For counter = 1 To 7 Step 1
        sum = sum + counter
    Next

    ' show the number to the user
    Label1.Text = CStr(sum)
End Sub
```

As this program is executed, it will keep the variables in memory, and the values of the variables will change during the execution of the program.

1. In the table below, write the values of the variables counter and sum after they are initialized (in the first assignment statement) under the column labeled “init”.
2. Then for each loop iteration, write the values of the variables counter and sum at the end of that loop iteration (from the assignment statements during the loop and from the automatic assignment counter = counter + 1 at the end of the for loop). These are labeled Val1 through Val7 in the table.

Variable	init	Val1	Val2	Val3	Val4	Val5	Val6	Val7
counter								
sum								

What number will be shown to the user in Label1.Text at the end of the program? (write it here):

2. Compute Interest – extending the compute interest program

For this application, you should open Visual Studio and create a new project named something like ComputeInterest. This new project should be in the same lab projects folder as usual.

In the **initial form**, change the name and text properties of the form for the new ComputeInterest project.

In class, an application was developed for a simpler version of the ComputeInterest program. In this application, the user typed in numbers for the following:

Initial bank balance

Number of years to compute a new balance

Interest rate, given as a decimal number

There were two buttons, one called Calculate, and one for Close.

When the user clicked on the Calculate button, a new bank balance was computed by computing the interest each year and adding it into the bank balance for that year.

Here is the program:

```
Public Class Form1

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Dim counter, numyears As Integer
        Dim balance, interestrate, interestamount As Single

        ' Get the user balance, number of years, and interest rate
        balance = CSng(TextBox1.Text)
        numyears = CInt(TextBox2.Text)
        interestrate = CSng(TextBox3.Text)

        ' compute the interest for each year and add to balance
        For counter = 1 To numyears Step 1
            interestamount = balance * interestrate
            balance = balance + interestamount
        Next

        ' show the final balance to the user
        Label6.Text = Format(balance, "currency")

    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Close()
    End Sub
End Class
```

For lab today, you are to extend this Compute Interest application. In the extended version, the user can also type in an amount that he or she proposes to add to the bank balance as an investment each year.

Write this application, allowing the user to type in:

Initial bank balance

Number of years to compute a new balance

Interest rate, given as a decimal number

Annual investment

Add code to the Calculate button to also add in the investment amount each year. What type of variable should you add for the annual investment? Suppose that this variable is called investment. Then the new formula for a new bank balance every year would be:

$\text{balance} = \text{balance} + \text{interestamount} + \text{investment}$

Test your program and write one example here:

Initial bank balance typed by user	
Number of years	
Interest rate	
Annual investment	
New bank balance computed	

3. Understanding While Loops – showing the variables of a program

Again for this problem, you are not to put any part of the program into Visual Studio. You are to write your answers on this lab sheet. (Think of this as a practice exam question.)

A. Consider the following program for a button:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim counter, sum As Integer
    counter = 0
    sum = 0

    ' add integers from 1 to 7
    While counter <= 7
        counter = counter + 1
        sum = sum + counter
    End While

    ' show the number to the user
    Label1.Text = CStr(sum)
End Sub
```

As this program is executed, it will keep the variables in memory, and the values of the variables will change during the execution of the program.

1. In the table below, write the values of the variables counter and sum after they are initialized (in the first assignment statement) under the column labeled “init”.
2. Then for each loop iteration, write the values of the variables counter and sum at the end of that loop iteration (from the assignment statements during the loop. These are labeled Val1 through Val7 in the table.

Variable	init	Val1	Val2	Val3	Val4	Val5	Val6	Val7
counter								
sum								

What number will be shown to the user in Label1.Text at the end of the program? (write it here):

4. Compute Savings – modifying the compute interest program

For this application, you are going to create a similar application to the one that you did in the last example, called something like ComputeSavings. Before starting this next program, keep Visual Studio open to the ComputeInterest project. Go to the Form1.vb tab and copy the program that you wrote for the Calculate button. Save this program in a text editor such as Notepad or Word so that you can refer to it while writing the ComputeSavings program.

Now open a new project with Visual Studio named something like **ComputeSavings**. This new project should be in the same lab projects folder as usual.

For this example, you are to write a similar application to the ComputeInterest project, except that this time the user wants to achieve a certain amount of savings and needs to compute how many years it will take to achieve their goal amount.

Write this application, allowing the user to type in:

- Initial bank balance
- Interest rate, given as a decimal number
- Annual investment
- Goal savings amount**

There should be two buttons, one for Compute Number of Years and one for Close. The Compute Number of Years button should get the user inputs and use a while loop to compute new bank balances until the goal amount is achieved.

Look at the code that you saved for the ComputeInterest program.

- After the declarations of variables, set the numberyears variable to start at 0.
- Modify the For loop in that program to be a **while loop** in your new ComputeSavings program. The condition for the while loop should be to keep computing a new balance as long as the balance is less than the savings goal.
- Note that you must add a statement to the while loop to explicitly add one to the number of years variable during the loop.
- After the while loop, display the number of years to the user.

Test your program and write one example here:

Initial bank balance typed by user	
Interest rate	
Annual investment	
Savings goal	
Number of years to achieve goal	

To hand in your lab for the week, staple together the lab handouts from Monday and Wednesday, together with printouts of the two programs from the two labs. If you need more time, you may hand in the labs next week on September 23.