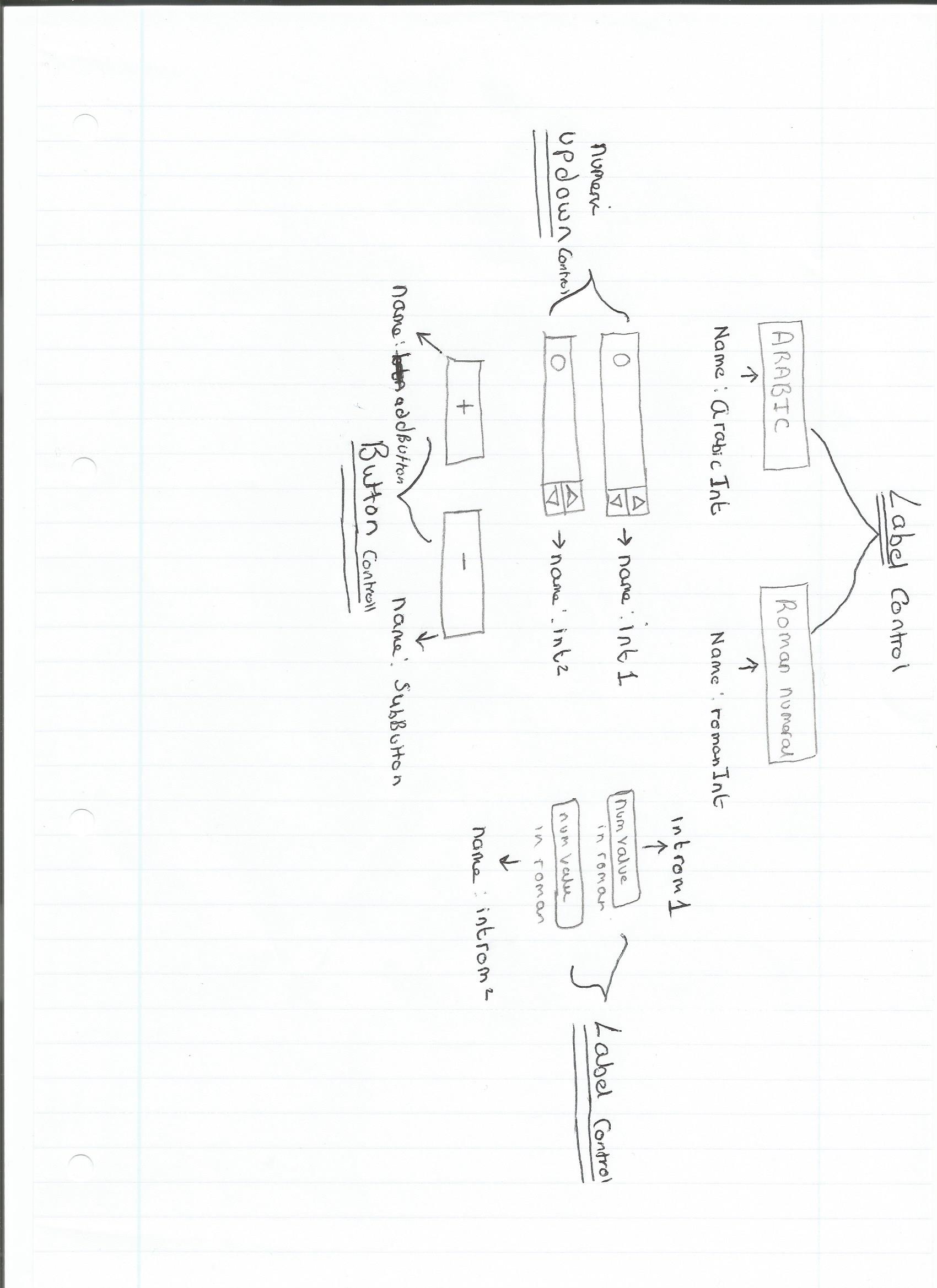
**FORM INTERFACE**



**CODING**

Public Class Form1

Dim output As Integer

*(DIM variable that we are defining. The variable output that we are going to get is going to be an integer (whole number)*

Dim roman As String

*(roman is defined as a string so in our case a list of characters which denotes roman numerals.)*

Private Sub Label1\_Click(sender As System.Object, e As System.EventArgs) Handles arabicInt.Click

End Sub

Private Sub NumericUpDown1\_ValueChanged(sender As System.Object, e As System.EventArgs) Handles int1.ValueChanged

End Sub

*(here the numericbox has been assigned name as int1.)*

Private Sub addButton\_Click(sender As System.Object, e As System.EventArgs) Handles addButton.Click

output = int1.Value + int2.Value

roman = ""

*(when you use the button function, the addition one is this case, the output value will be integers 1 value plus integers 2 value. In our case the integers values 1 and 2 are assigned to the numericbox. So whatever value is in their will be our output values. The roman value is defined to show strings characters so once we set our values the roman numerals will correspond to whatever we assign to it. Below we set our roman numerals values)*

If output < 0 Then

output = 0

End If

If output > 5000 Then

output = 0

End If

*(this is our loop statement, where it goes through each statement until required result is satisfied. For example, the above loop states any value that is less than 0 will be shown to the user as 0. If that’s not the case it moves onto the next statement, where any output value over 5000 will be shown as 0.)*

arabicInt.Text = output

*(this sets the arabicint to the value of the output, hence the 2 values from integer 1 and 2 will be shown under the Arabic label box)*

While output >= 1000

*(when the output is greater than or equal to 1000 the roman numeral will be assigned M value.)*

roman = roman + "M"

*(this denotes what the roman value will be. It can be any roman numeral that satisfies the input data plus M that will be shown as the value.)*

output = output – 1000

*(Takes away 1000 from the output so the while loop can find what the next number is going to be)*

End While

*(ends the while loop)*

While output >= 900

roman = roman + "CM"

output = output – 900

*(takes away 900 from the output because this is the assigned value CM in roman numerals.)*

*(Loop continues for each value stated and their corresponding roman numeral value is assigned to that.eg below output is greater than or equal to 500 roman numerals will be D)*

End While

While output >= 500

roman = roman + "D"

output = output - 500

End While

While output >= 400

roman = roman + "CD"

output = output - 400

End While

While output >= 100

roman = roman + "C"

output = output - 100

End While

While output >= 90

roman = roman + "XC"

output = output - 90

End While

While output >= 50

roman = roman + "L"

output = output - 50

End While

While output >= 40

roman = roman + "XL"

output = output - 40

End While

While output >= 10

roman = roman + "X"

output = output - 10

End While

While output >= 9

roman = roman + "IX"

output = output - 9

End While

While output >= 5

roman = roman + "V"

output = output - 5

End While

While output >= 4

roman = roman + "IV"

output = output - 4

End While

While output >= 1

roman = roman + "I"

output = output - 1

End While

romanInt.Text = roman

End Sub

Private Sub Form1\_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load

End Sub

Private Sub subButton\_Click(sender As System.Object, e As System.EventArgs) Handles subButton.Click

*(this will handle the values for when we minus the numbers together.)*

output = int1.Value - int2.Value

roman = ""

If output < 0 Then

output = 0

End If

If output > 5000 Then

output = 0

End If

arabicInt.Text = output

While output >= 1000

roman = roman + "M"

output = output - 1000

End While

While output >= 900

roman = roman + "CM"

output = output - 900

End While

While output >= 500

roman = roman + "D"

output = output - 500

End While

While output >= 400

roman = roman + "CD"

output = output - 400

End While

While output >= 100

roman = roman + "C"

output = output - 100

End While

While output >= 90

roman = roman + "XC"

output = output - 90

End While

While output >= 50

roman = roman + "L"

output = output - 50

End While

While output >= 40

roman = roman + "XL"

output = output - 40

End While

While output >= 10

roman = roman + "X"

output = output - 10

End While

While output >= 9

roman = roman + "IX"

output = output - 9

End While

While output >= 5

roman = roman + "V"

output = output - 5

End While

While output >= 4

roman = roman + "IV"

output = output - 4

End While

While output >= 1

roman = roman + "I"

output = output - 1

End While

romanInt.Text = roman

End Sub

End Class

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Number | Description of test | Test Data | Expected outcome | Actual Outcome |
| 1 | ADDING 2 VALID NUMBERS | NUMBERS 50 +22 | 72 LXXII | FIGURE 1 |
| 2 | ADDING VALID 2ND NUMBER | 1+2 | 3 III | FIGURE 2 |
| 3 | VALID SUBTRACTION | 10-5 | 5 V | FIGURE 3 |
| 4 | SUBTRACTION EQUAL 0 | 52-52 | 0 | FIGURE 4 |
| 5 | SUBTRACTION LESS THAN 0 | 79-99 | 0 | FIGURE 5 |
|  |  |  |  |  |

**TESTING TABLE:**

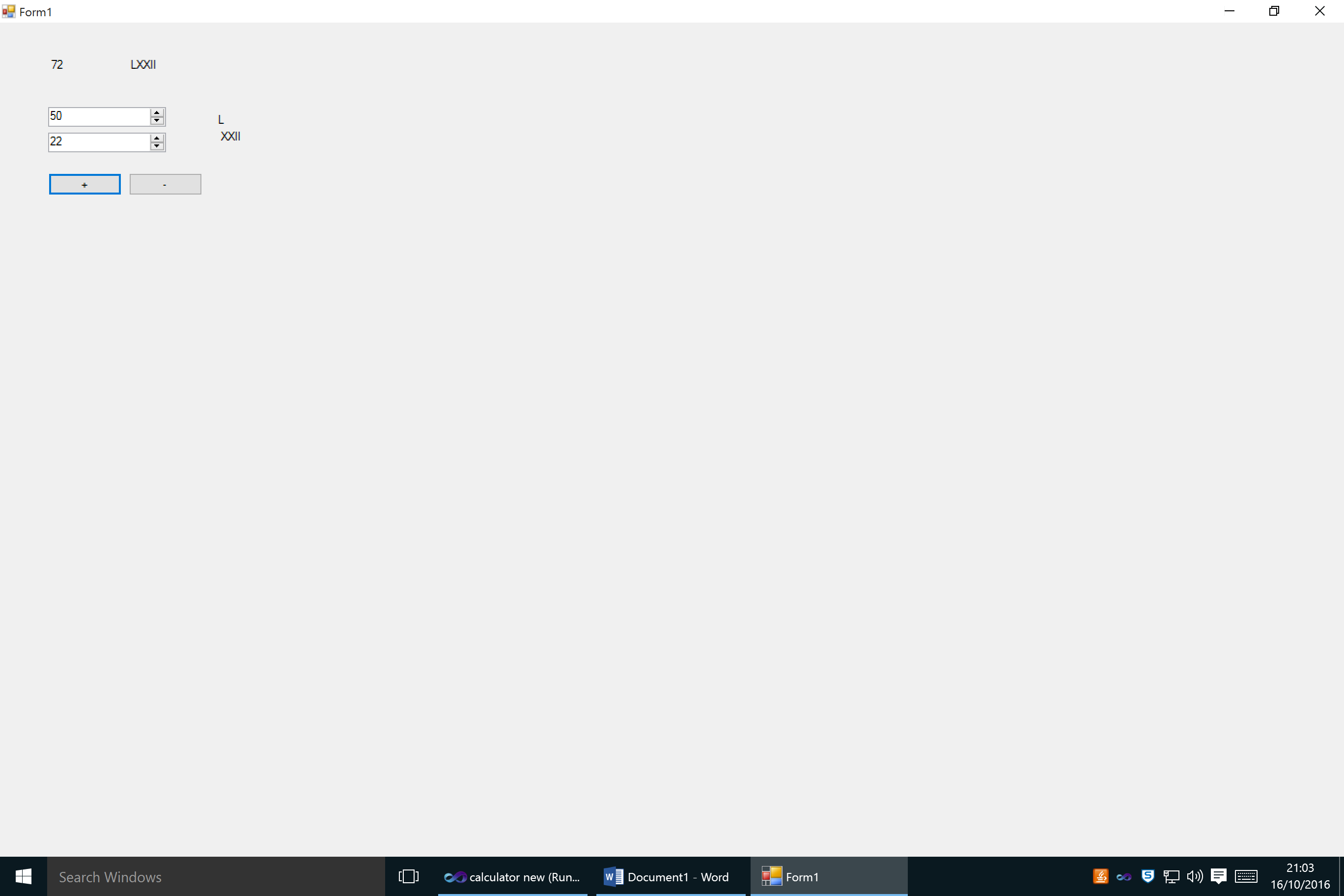


FIGURE 1: ADDING 2 NUMBERS (VALID 1ST NUMBER)

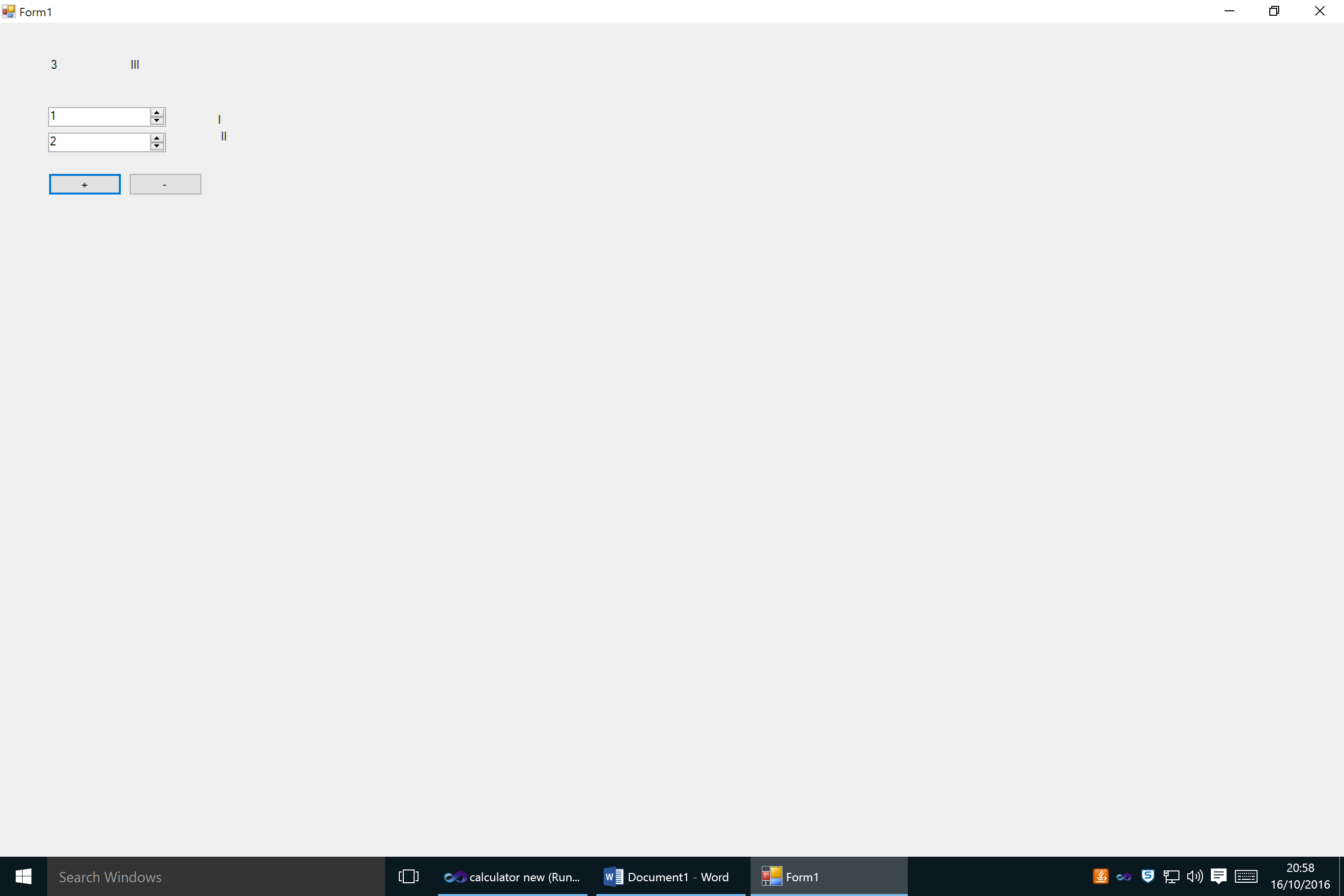


FIGURE 2: VALID 2ND NUMBER

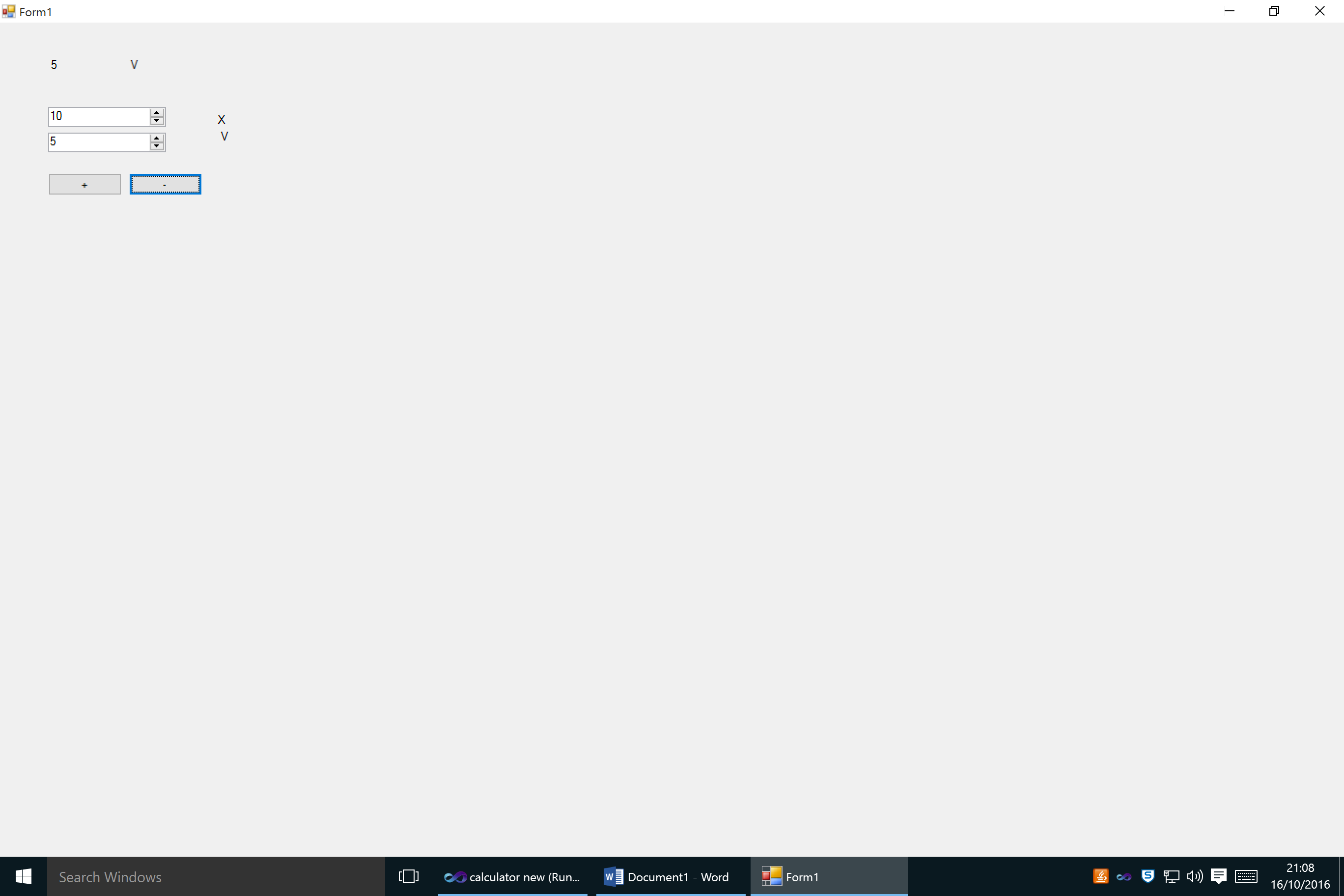


FIGURE 3: VALID SUBTRACTION

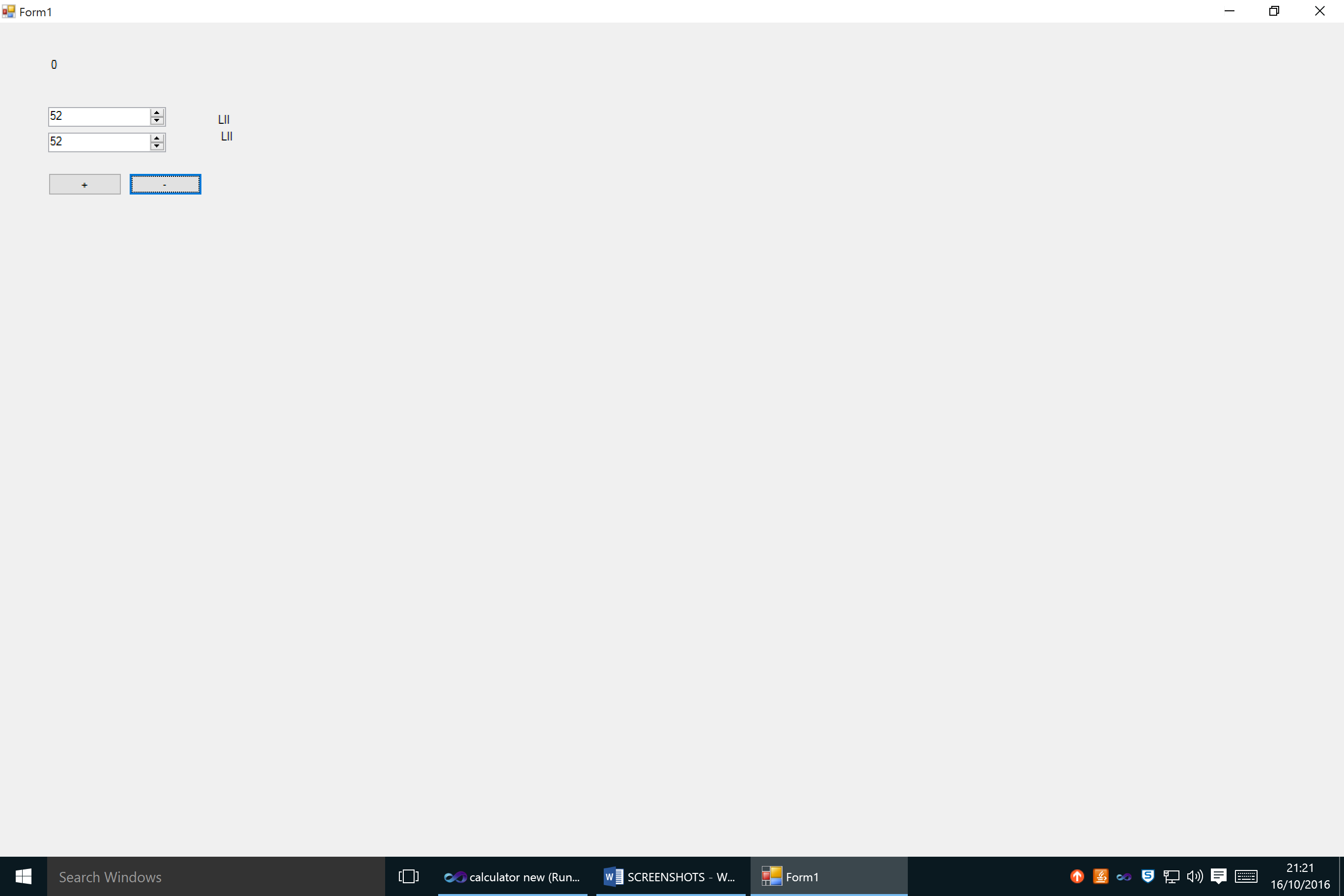


FIGURE 4: SUBTRACTION = 0

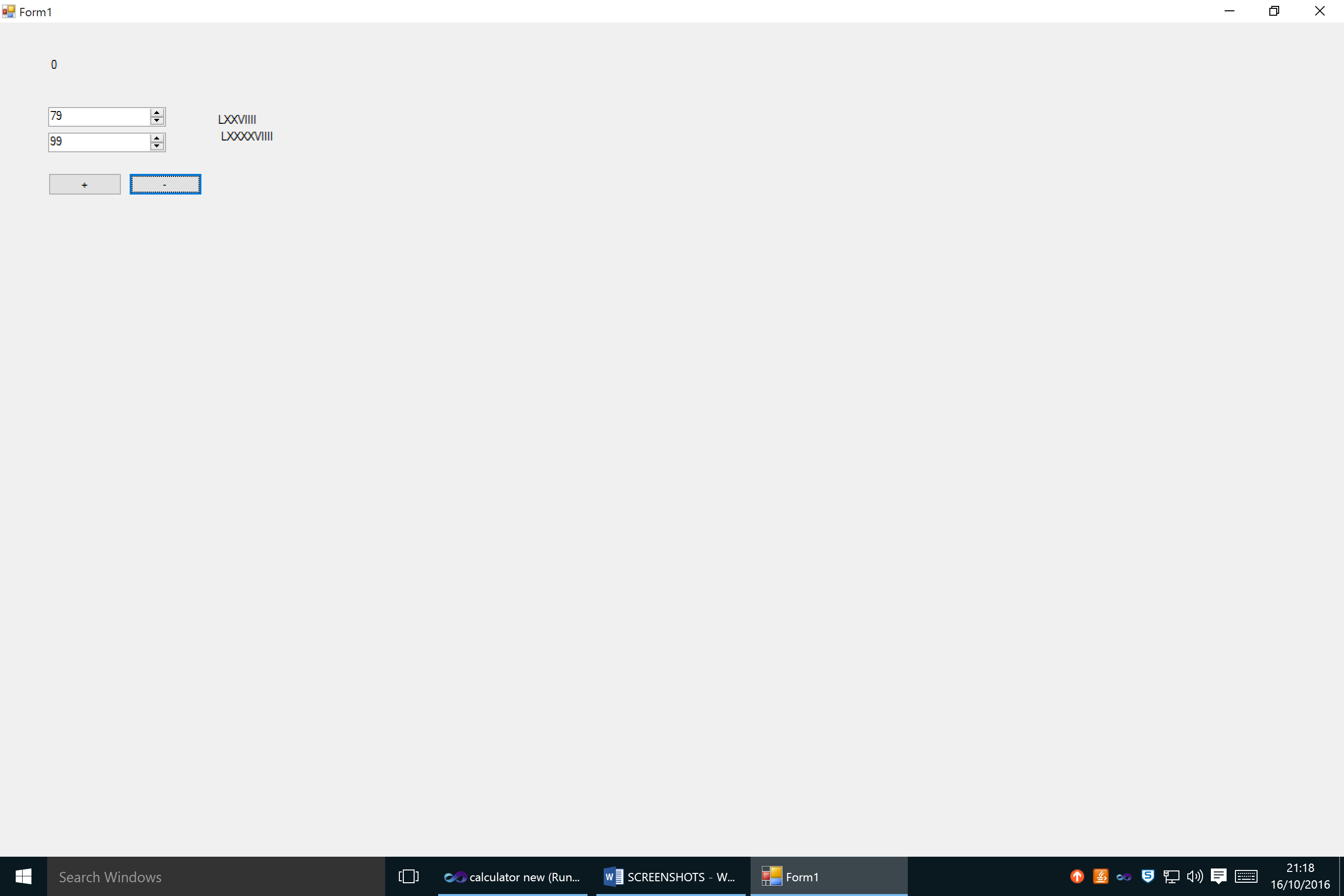


FIGURE 5: SUBTRACTION LESS THAN 0