

Steps:

Set total to zero

Set counter to the first number plus a step

While counter less than the second number

test the counter

if counter is odd Add it to sum

add the step to the counter

Print the sum

The screenshot shows a Windows application window titled "Form1". It contains three text input fields arranged horizontally. The first field is labeled "FirstNumbe" and is empty. The second field is labeled "Second Number" and is empty. The third field is labeled "Sum" and is empty. Below the first two fields are two buttons: "Calculate" and "Exit".

© Dr Izeddin Hidar 2007

The screenshot shows the same "Form1" window. The "Calculate" button is now highlighted, indicating it has been pressed. The "FirstNumbe" field contains the value "5", the "Second Number" field contains the value "8", and the "Sum" field contains the value "7".

© Dr Izeddin Hidar 2007

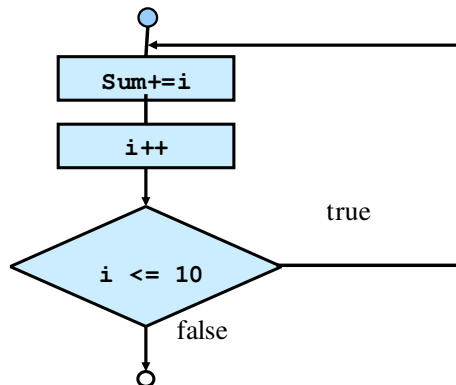
```
private void ButCalculate_Click(object sender, EventArgs e)
{
    int FirstNumber, SecondNumber, Sum=0,i;
    FirstNumber = Int32.Parse(TxtFirstNumber.Text);
    SecondNumber = Int32.Parse(TxtSecondNumber.Text);
    i = FirstNumber+1;
    while (i < SecondNumber)
    {
        if (i % 2 == 1)
            Sum += i;
            i++;
    }
    TxtSum.Text = Sum.ToString();
}

private void ButExit_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

© Dr Izeddin Hidar 2007

2. do while Repetition Structure

- Action is performed.
- An action is to be repeated
 - Continues while statement is true
 - Ends when statement is false
- Contain either a line or a body of code



© Dr Izeddin Hidar 2007