

# Repetition

*CE 311 K - Introduction to Computer  
Methods*

*Daene C. McKinney*

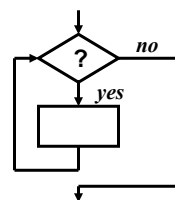
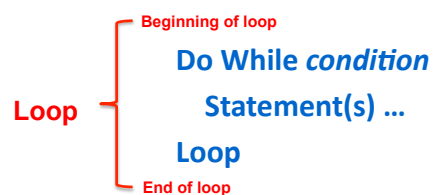
## Introduction

- Do – While Loops
- Do – Until Loops
- Peek
- For – Next Loops

## Repetition - Loops

- Control structures used to repeat statements until a condition is met
- Do loop
  - *Indeterminate* looping structure
  - Condition stops the loop
- For – Next loop
  - *Determinate* looping structure
  - Number of cycles stops the loop

## Do – While Loops



## Do – While Loops

```

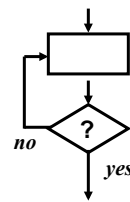
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim sum, count As Long
    Do While count <= 10
        sum = sum + count
        count = count + 1
        ListBox1.Items.Add(sum)
        ListBox2.Items.Add(count)
    Loop
End Sub

```

The screenshot shows a Windows form titled 'Form1' with two list boxes and two buttons. The 'Sum' list box contains the following values: 0, 1, 3, 6, 10, 15, 21, 28, 36, 45, 55. The 'Count' list box contains the following values: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11. Below the list boxes are two buttons labeled 'Go!' and 'Stop'.

## Do - Until Loops

Loop {
 Beginning of loop  
**Do**  
Statements...  
Loop Until condition  
End of loop



## Do - Until Loops

The screenshot shows a Windows form titled 'Form1'. It contains two list boxes, one labeled 'Sum' and one labeled 'Count'. The 'Sum' list box contains the following values: 0, 1, 3, 6, 10, 15, 21, 28, 36, 45, 55. The 'Count' list box contains the following values: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11. Below the list boxes are two buttons: 'Go!' and 'Stop'.

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim sum, count As Long
    sum = 0
    count = 1
    Do
        sum = sum + count
        count = count + 1
        ListBox1.Items.Add(sum)
        ListBox2.Items.Add(count)
    Loop Until count > 10
End Sub
```

## Notes on Loops

- Avoid “Infinite” loops
- Any “Do-While” statement can be converted to an “Do-Until” statement, and vice-versa
- “Do-While” tests at the top of the loop
- “Do-Until” tests at the bottom of the loop

## Processing Data With Loops

- Data in files
  - Display
  - Select/Search
  - Calculation/Transform
- Detect the end of a file (EOF)
  - Peek: determine when end of file is reached
  - Counters: Calculate # of elements in a list
  - Accumulators: Sum numerical values in lists
  - Flags: Test for certain conditions

## Peek

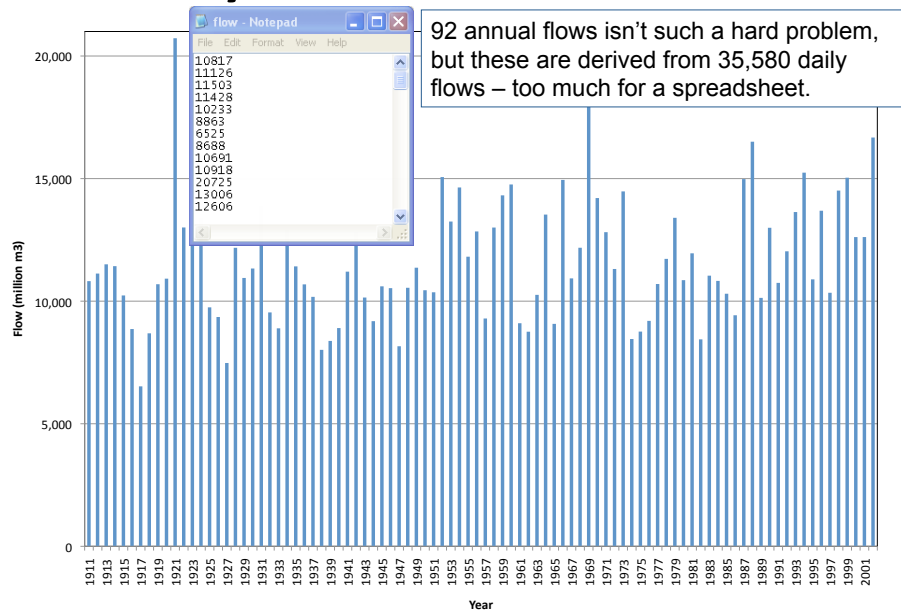
- Read data from a file using a Do Loop
- How do we detect the End Of File (EOF)?
- Value of
  - `sr.Peek = 1st character of current line in file`
- If EOF has been reached
  - `sr.Peek = -1`

# Central Asia

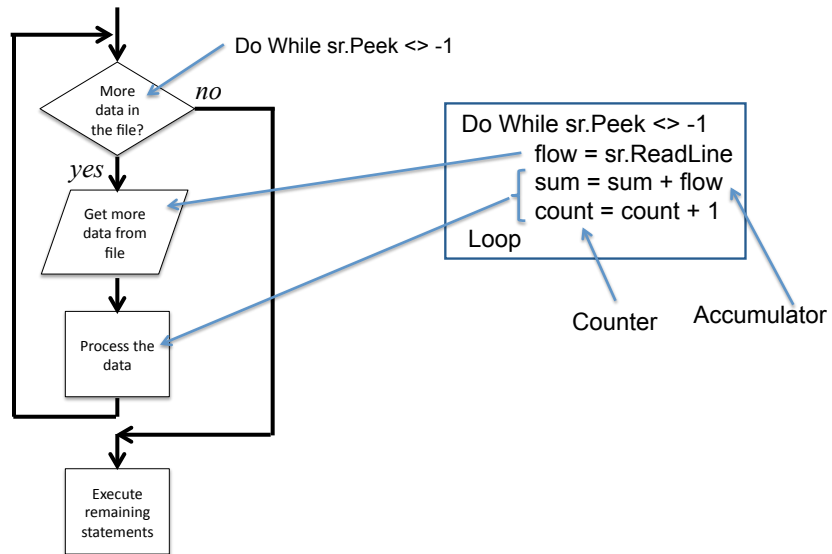
**Toktogul Dam**  
 140 m  
 19.5 km<sup>3</sup>



# Naryn River Annual Flows



## Peeking Into a Data File



## Using Peek

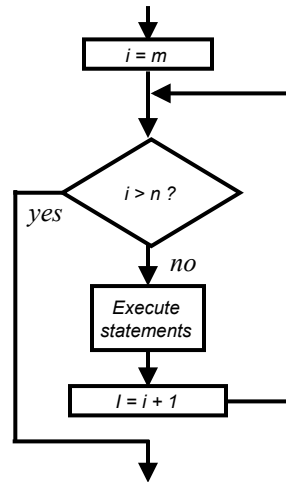
```

Private Sub Button1_Click(ByVal sender As System
Dim flow, sum, count, average As Double
Dim sr As IO.StreamReader = IO.File.OpenText
sum = 0
count = 0
Do While sr.Peek <> -1
    flow = sr.ReadLine
    sum = sum + flow
    count = count + 1
Loop
average = sum / count
average = Math.Round(average)
ListBox1.Items.Add("Count = " & count)
ListBox1.Items.Add("Average = " & average)
End Sub
  
```

## For – Next Loops

**Loop** {
   
   Beginning of loop
   
   **For**  $i = m$  **To**  $n$ 
  
   **statement(s)**
  
   **Next**
  
   End of loop

$i$  – control variable  
 $m$  – initial value of  $i$   
 $n$  – terminating value of  $i$   
 $next$  – increment  $i$  by 1  
 $statements$  – get executed



## Example

- Suppose the population of a city is 200,000 in 2009 and is expected to grow at a rate of 3% per year. Display the population every year up to 2015.

2009	200,000
2010	206,000
2011	212,180
2012	218,545
2013	225,102
2014	231,855
2015	238,810

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim pop As Double
    pop = 200000
    For i As Long = 2009 To 2015
        ListBox1.Items.Add(i & " " & FormatNumber(pop, 0))
        pop = (1 + 0.03) * pop
    Next
End Sub
  
```



## Example

- Step s

```
Private Sub Button1_Click(ByVal sender
Dim n, s As Double
n = Cdbl(TextBox1.Text)
s = Cdbl(TextBox2.Text)
ListBox1.Items.Clear()
For i As Double = 0 To n Step s
    ListBox1.Items.Add(i)
Next
End Sub
```

Form1

n: 10 s: 1

0  
1  
2  
3  
4  
5  
6

Go! Stop

Form1

n: 3.2 s: .5

0  
0.5  
1  
1.5  
2  
2.5  
3

Go! Stop

## Example - Factorial

```
Private Sub Button1_Click(ByVal sender As System.Object, By
Dim i, x As Long
i = Cdbl(TextBox1.Text)
x = Factorial(i)
ListBox1.Items.Add(" Factorial of " & i & " is " & x)
End Sub

Function Factorial(ByVal i As Long) As Long
    Factorial = 1
    For j As Long = 2 To i
        Factorial = Factorial * j
    Next
End Function
```

Form1

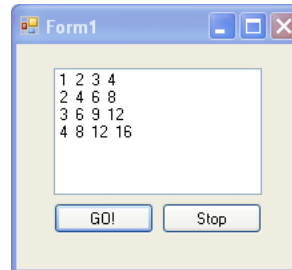
Number 18

Factorial of 3 is 6  
Factorial of 6 is 720  
Factorial of 9 is 362880  
Factorial of 12 is 479001600  
Factorial of 18 is 6402373705728000

Go! Stop

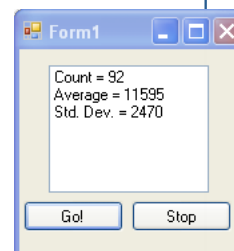
## Nested For – Next Loops

```
Private Sub Button1_Click(ByVal sender
    Dim row, entry As String
    ListBox1.Items.Clear()
    For j As Long = 1 To 4
        row = ""
        For k As Long = 1 To 4
            entry = j * k
            row = row & entry & " "
        Next
        ListBox1.Items.Add(row)
    Next
End Sub
```



## Example

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As Sy
    Dim flow, sum, count, average, stdev As Double
    Dim sr As IO.StreamReader = IO.File.OpenText("C:\temp\flow.txt")
    sum = 0
    count = 0
    Do While sr.Peek <> -1
        flow = sr.ReadLine
        sum = sum + flow
        count = count + 1
    Loop
    sr.Close()
    average = Math.Round((sum / count), 0)
    ListBox1.Items.Add("Count = " & count)
    ListBox1.Items.Add("Average = " & average)
    sr = IO.File.OpenText("C:\temp\flow.txt")
    sum = 0
    Do While sr.Peek <> -1
        flow = sr.ReadLine
        sum = sum + (flow - average) ^ 2
    Loop
    stdev = Math.Round(Math.Sqrt(sum / (count - 1)), 0)
    ListBox1.Items.Add("Std. Dev. = " & stdev)
    sr.Close()
End Sub
```



## Summary

- Do – While Loops
- Do – Until Loops
- Peek
- For – Next Loops