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 Loop Diagram]

- Begin the loop
- Evaluate the condition
- If true, execute the statements
- If false, end the loop
Do – While Loops

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

Do - Until Loops

![Flowchart showing Do-Until loop structure]
Do - Until 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

\[ \text{sr.Peek} = 1\text{st character of current line in file} \]

• If EOF has been reached

\[ \text{sr.Peek} = -1 \]
92 annual flows isn’t such a hard problem, but these are derived from 35,580 daily flows – too much for a spreadsheet.
Peeking Into a Data File

Do While sr.Peek <> -1
  More data in the file?
  yes
    Get more data from file
    Process the data
    Execute remaining statements
  no
  Do While sr.Peek <> -1
    flow = sr.ReadLine
    sum = sum + flow
    count = count + 1
  Loop

Using Peek

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
  Dim flow, sum, count, average As Double
  Dim sr As New StreamReader(File.OpenText
  sum = 0
  count = 0
  Do While sr.Peek <> -1
    flow = sr.ReadLine
    sum = sum + flow
    count = count + 1
  Loop
  average = sum / count
  ListBox1.Items.Add("Count = " & count)
  ListBox1.Items.Add("Average = " & average)\nEnd Sub

Count = 92
Average = 77996
For – Next Loops

For \( i = m \) To \( n \)
\[ \text{statement(s)} \]
Next

- \( i \) – 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.

```vbnet
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 1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim n, m As Double
    n = CSb1(TextBox1.Text)
    m = CSb1(TextBox2.Text)
    ListBox1.Items.Clear()
    For i As Double = 0 To n Step m
        ListBox1.Items.Add(i)
    Next
End Sub

Example - Factorial

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim i, x As Long
    i = CSb1(TextBox1.Text)
    x = Factorial(i) + 1
    ListBox1.Items.Add(“Factorial of “ + i + “ is “ + x) + 1
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: 10

Factorial of 1 is 1
Factorial of 2 is 2
Factorial of 3 is 6
Factorial of 4 is 24
Factorial of 5 is 120
Factorial of 10 is 3628800
Factorial of 16 is 84043776
Factorial of 18 is 923562373
Factorial of 20 is 2432902008

Go! Stop
Nested For – Next Loops

Example

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    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

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