While Loops

Chapter 3
Console Input/Output (I/O)

- Console I/O is persistent: each line of input and output remains in the console window for the lifetime of the program.
- In Visual Logic, console I/O options are under the More>> button in the edit dialog.
- In Visual Logic, console I/O is indicated by the console screen icon (small rectangle) at the top of the flowchart element.
- The end-of-output symbol (§) always appears at the end of the console output expression.
- The position of the end-of-output symbol (§) determines the starting location for the NEXT line of console I/O.
Begin

Output: "This is line 1"
"This is line 2"$

Output: "This is NOT line 3"
$

Output: "This is line 3"$

End
While Loop – Count 1 to 5

Begin

Count = 1

Count <= 5

True

Output: "Count is " & Count

Count = Count + 1

False

Output: "All Done!"

End
While Loop – Count Backward from ______ to __________

Note: the greater than equal to $\geq$

Decreasing the LCV

```plaintext
60 59 58 57 56 55 54 53 52 51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34
33 32 31 30 29 28 27 26 25 24 23 22 21 20
```
While Loops

- **While loops** are used to repeat actions
- In Visual Logic, the While loop flowchart element is a **six-sided figure** with a condition and two exit arrows—true and false
- When control flows to the While loop, the **condition is evaluated**
- If the condition is true, the control flows out through the true arrow into the body of the loop
- At the end of the loop body, the control flows back to the While loop and the condition is evaluated again
- This process **repeats** until the condition eventually evaluates to false, at which time the control flows through the false arrow to the statement after the While loop
A common template for a While Loop

1. Initialize LCV
2. Test LCV
   - True: Do Something
   - False: Update LCV
3. After Loop

LCV = Loop Control Variable
Display Even Numbers

Which is the LCV?

Number
While Loop – Count Backward from _______ to __________

Note: the greater than equal to \( \geq \)

Decreasing the LCV
Displaying Output in Columns of 5

A Guide to Working with Visual Logic
LCV: initial value?
Loop Condition?
Update LCV:
Output in Loop Body:
While Loops – Multiplication Tables

A constant is an assigned variable whose value does not change throughout the program.

Why do we need constant variables?

10 times 1 is 10
10 times 2 is 20
10 times 3 is 30
10 times 4 is 40
10 times 5 is 50
10 times 6 is 60
10 times 7 is 70
10 times 8 is 80
10 times 9 is 90
10 times 10 is 100
10 times 11 is 110
10 times 12 is 120
10 times 13 is 130
10 times 14 is 140
10 times 15 is 150
10 times 16 is 160
10 times 17 is 170
10 times 18 is 180
10 times 19 is 190
10 times 20 is 200
10 times 21 is 210
10 times 22 is 220
10 times 23 is 230
10 times 24 is 240

All Done!
While Loop – Post Test

Begin
Number = 2
Input: MaxNumber
Output: Number
Number = Number + 2
Number <= MaxNumber
End

Enter the largest even number you want 0
2

While Loop
Condition: Number <= MaxNumber
Loop Type: Post-Test
OK Cancel
Pre-Test and Post-Test While Loops

- 2 types of while loops: Pre-Test and Post-Test
- A Pre-Test loop tests the condition BEFORE the body is executed. If the condition is false initially (i.e. the first time) then the loop is NEVER executed.
- In a Post-Test loop, the body is executed one time before the looping condition is tested
- A Post-Test loop guarantees one execution of the loop body regardless of the condition!
- After one pass through the loop there is no difference between the pre-test and post-test loops
Sum of 5 numbers

Variables:

Count is the LCV

sum – accumulates the sum as the numbers are entered
Grocery Checkout Problem

- A user purchases several items. The price of each item is entered. The Sales Tax is calculated and added to the total price to determine the total amount due.
- Do we know how many items there are?
- Will each user buy the same number of items as another user?
- How many times should we repeat the loop?
- The number of repetitions varies!
- Use a Sentinel value to control the loop
What is the sentinel value in this example?
-1

What is the LCV?
Where is it initialized?
Where is it updated?
Pseudo-Code: Sentinel Value Grocery Checkout

Assign: SubTotal = 0

**Input:** ItemPrice

While (ItemPrice <> -1)

Assign: SubTotal = SubTotal + ItemPrice

**Input:** ItemPrice

EndWhile

Assign: Total = SubTotal + 0.6 * SubTotal

Output: “Your purchase total is “ & Total
A template for a While Loop with Sentinel – Reading Input

Important Note:
• Read the 1st Input BEFORE the loop,
• Read subsequent input INSIDE the Loop – last step.
Average of Input Numbers - Sentinel

```
Please type a value for NUMBER:10
Please type a value for NUMBER:20
Please type a value for NUMBER:30
Please type a value for NUMBER:-1
Count is 3
Sum is 60
Average is 20
```
Sentinel Value, Counter, Accumulator

- **Sentinel Values** are “end of data” values that indicate all data has been processed. Sentinel values are **NOT part of the data set** and should **NOT be processed**.

- **Counters** are variables typically used in a loop to help **count and control** the number of times the loop body is executed and are usually **incremented by one** (but not necessarily!)

- **Accumulators** maintain a **running total**; they are variables used inside a loop to help **calculate totals** (and averages). Accumulators are typically **incremented/updated by the value of a variable**.