# Applications Programming

Boolean Expressions and Control Structure

# Questions in VBA

- What kind questions can be asked and answered in VBA?
  True/False questions, such as:

  - Is it true that the user entered a positive number?

  - Is it true that the value stored in a cell is exactly the same as the literal string value "September"?

- What can't be done by a computer program directly? Understand the semantics of any data value in general, such as:

  - What is the value stored in a cell in general?

  - Is the number 42 the correct answer?

  - What does the user input "okay" mean?

# Boolean Values and Variables

- Boolean data type: Boolean

- Boolean literal values: True, False

- Boolean variable declaration:
  dim validInput as Boolean

- Assign boolean value to boolean variable:
  validInput = True;

- Boolean expression: describes a True/False condition

# Relational Operators

- A comparison using relational operator is the smallest building block of a Boolean expression.

- A relational operator compares two values and returns true or false based on the comparison.

- There are six relational operators in VBA:

  - Equal to: (X = Y)

  - Not equal to: (X <> Y)

  - Less than: (X < Y)

  - Greater than: (X > Y)

  - Less than or equal to: (X <= Y)

  - Greater than or equal to: (X >= Y)

# Logical Operators

- Not
  if A is true,
  then (Not A) is false

| A | Not A |
|---|-------|
| TRUE | FALSE |
| FALSE | TRUE |

- And
  if A is true and B is true,
  then (A And B) is true

| A | B | A And B |
|---|---|---------|
| TRUE | TRUE | TRUE |
| FALSE | TRUE | FALSE |
| TRUE | FALSE | FALSE |
| FALSE | FALSE | FALSE |

- Or
  if A is true or B is true,
  then (A Or B) is true

| A | B | A Or B |
|---|---|--------|
| TRUE | TRUE | TRUE |
| FALSE | TRUE | TRUE |
| TRUE | FALSE | TRUE |
| FALSE | FALSE | FALSE |

# Mix math, relational, logical operators

- Evaluation priority:

  - Anything inside ( ) has the higher priority

  - Math expressions first
    => to a literal value

  - Comparisons (relational operations) or boolean variables/function calls
    => to a True/False value

  - Logical operations in the order of not, and, then or
    => further to True/False value

  - From left to right for operations with same priority

- Example:
  ' s is string typed, x is double typed and y is integer typed
  s = InputBox("Please enter a number")
  dim isValid as Boolean
  isValid = IsNumeric(s) And x > y * 0.5

# Control Flow Graph

- A control-flow graph is a graph that shows all possible paths that might be traversed by a computer when the computer is executing this program.

- Each node in the graph represents a statement

start

function call

return

function call

return

End
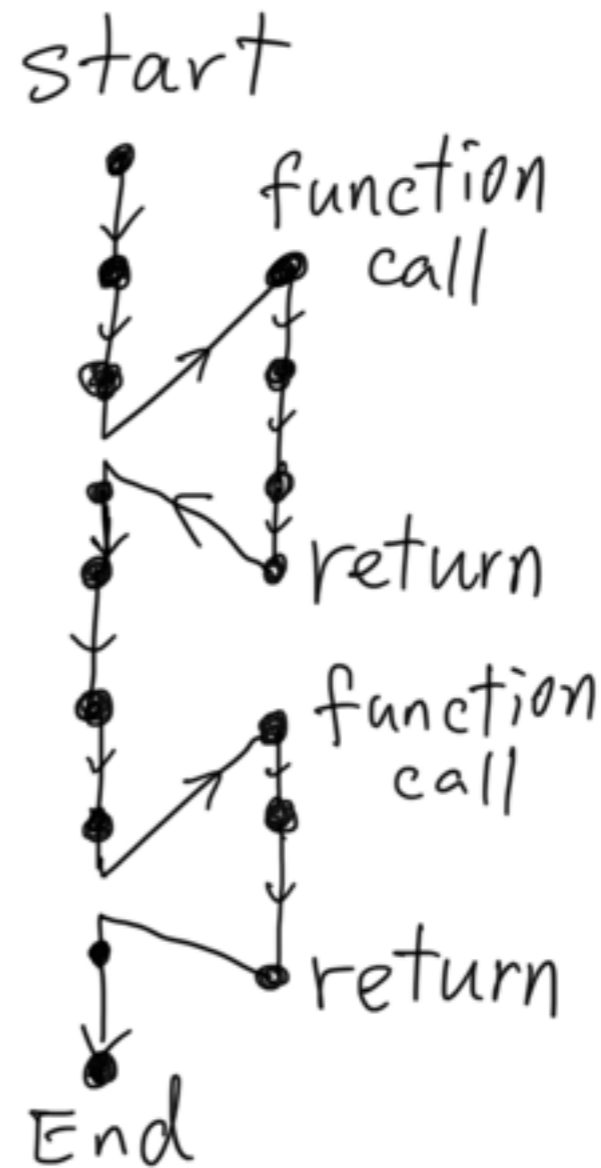
# Decision Structure

- Most programs need the ability to execute different sequences of instructions based on different conditions

- Decision structures allow a program's logic to have more than one path of execution

- Decision structure determine which code segments are executed based on the conditions at the time of executing the program

- Decision structures are used in selection and repetition statements

- conditions are described by boolean expressions that are evaluated to True or False based on the current data at the execution time

# if-else statement

- syntax:
  If condition Then
      ' if-block statements
  Else
      ' else-block statements
  End If

- condition is described by a boolean expression
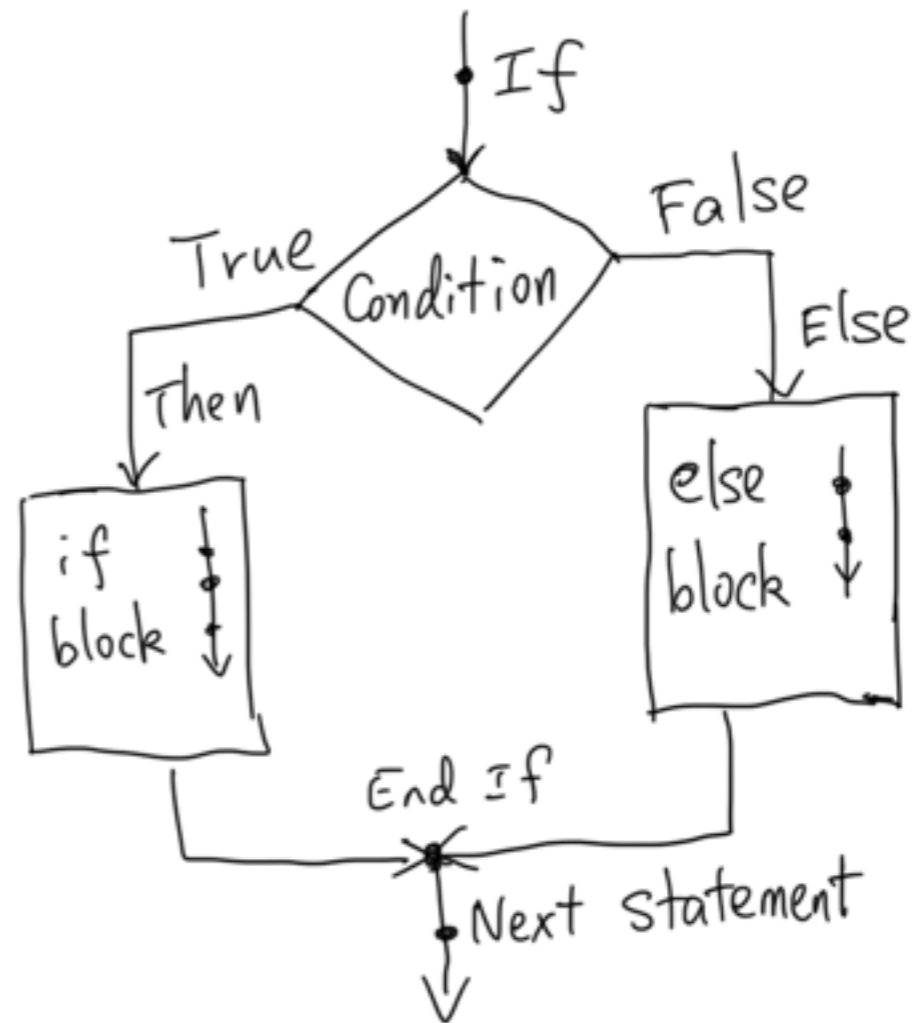
# if statement

- Else block is optional

- syntax:
  If condition Then
      ' if-block statements
  End If

- condition is described by a boolean expression

- If you don't want If block, negate the condition and then omit the Else block