PROGRAMMING LANGUAGE 2 (SPM 3112)

VISUAL BASIC PROGRAMMING: PART 2

NOOR AZEAN ATAN
MULTIMEDIA EDUCATIONAL DEPARTMENT
UNIVERSITI TEKNOLOGI MALAYSIA

Topics

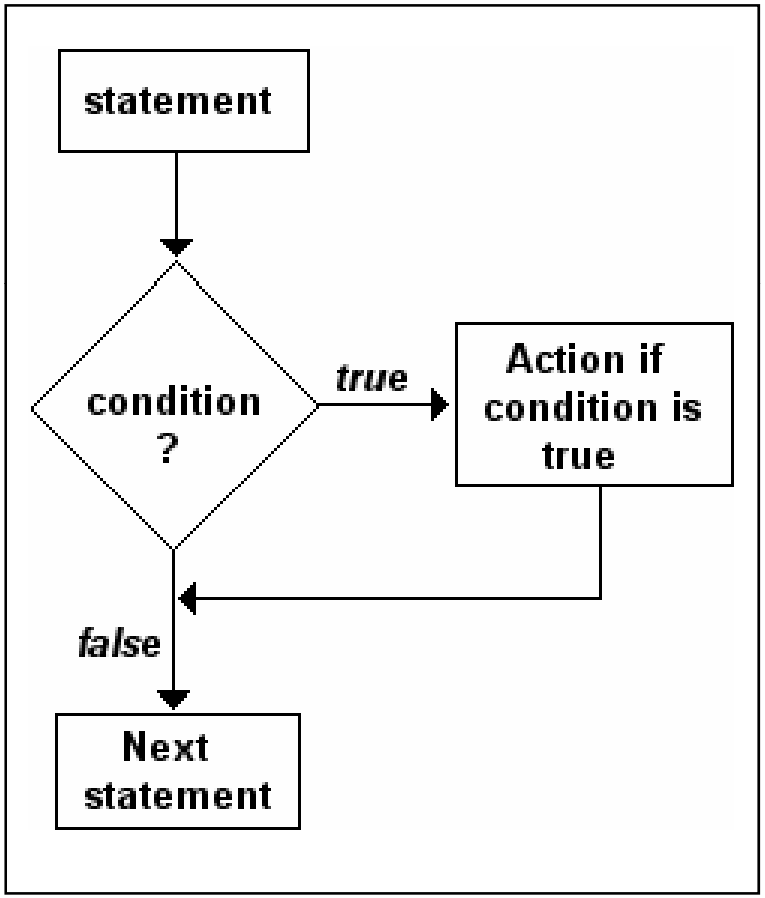
- Selection
  - if-then, if-then-else
  - Select case
- Repetition
  - While
    - While/Wend
    - Do While & Do Until
  - Do/Loop
    - Do/Loop (While) & Do/Loop (Until)

Introduction: Flow of control

- Normal flow of control is sequential
  - statements are executed one after another
  - in the order they are written
- Special structures to change normal flow of control:
  - Selection
    - Conditionally execute next statement (if-else, switch-case)
  - Repetition
    - Repeatedly execute next statement (with condition) (while, do-Loop)

SELECTION

Selection: IF-THEN



Selection: IF-THEN

- **Syntax:**

```
If ( condition ) Then
        perform action (when condition is true)
End If
```

Selection: IF-THEN

- **Example:**

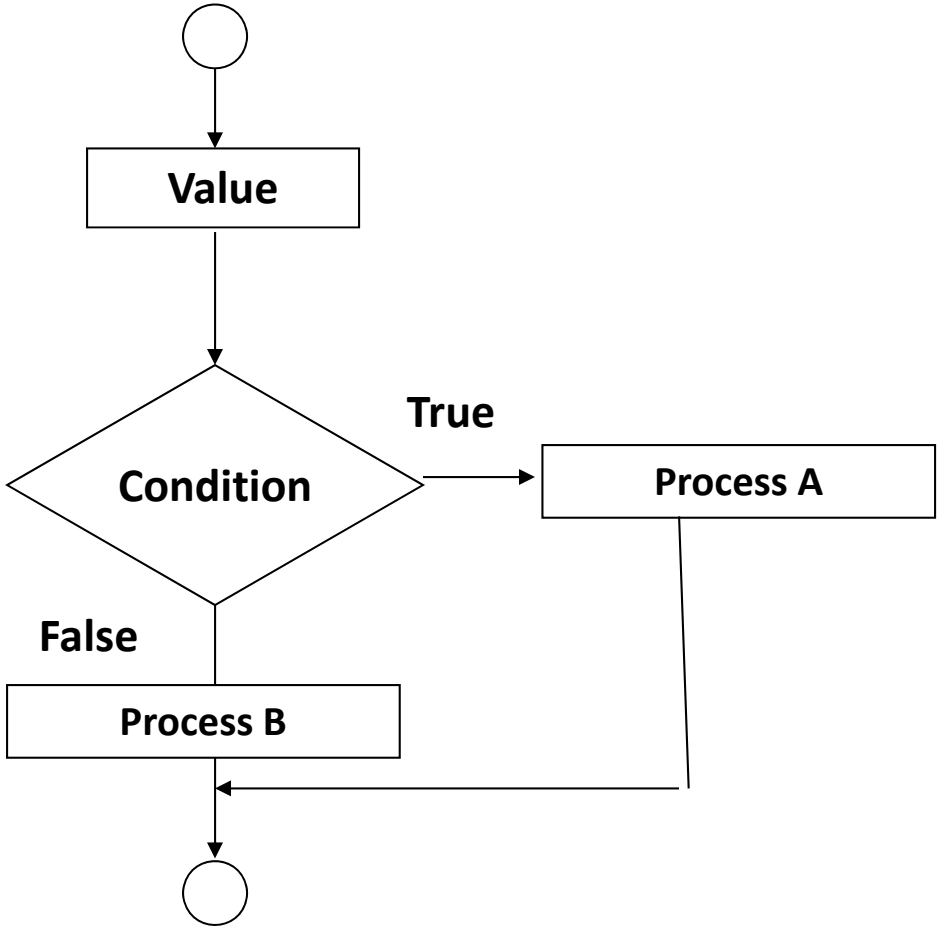
- **If (markahUjian < 40) Then**
  Label3.Caption = “Fail"
  End If

  → One statement

2. **If A = 1 Then**
   B = 0
   C =D
   End If

   → More than one statement

Selection: IF-THEN-ELSE



Selection: IF-THEN-ELSE

- **Syntax:**

```
If ( condition ) Then
  TRUE action
Else
  FALSE action
End If
```

Selection: IF-THEN-ELSE

```
Private Sub Command3_Click()
If (Text2 < 40) Then
        Label2.Caption = “Fail"
Else
        Label2.Caption = “Pass!!"
End If
End Sub
```

Selection: IF-THEN-ELSE

Example (with more than one statement):

```
If (Text2 < 40) Then
        Label2.Caption = "Ups Fail !!"
        Form1.BackColor = vbRed
Else
        Label2.Caption = “YES! YES! Pass…"
        Form1.BackColor = vbGreen
End If
```

Selection: IF-ELSE IF-ELSE

- **Syntax:**

```
If ( condition  A) Then
   TRUE action A
Elseif( condition B) Then
  TRUE action B
Else
   FALSE action
End If
```

Nested Selection: IF-IF-ELSE

- **Syntax:**

```
If ( condition  A) Then
   TRUE action A
   if( condition B) Then
         TRUE action B
   end if
Else
   FALSE action
End If
```

- **Checking Gred**
- **>=80 – A**
- **>=70 – B**
- **>= 60 – C**
- **>= 50 – D**
- **< 50 - E**

Pass

Barely Pass

Need to do Better

```
Private Sub Command2_Click()

If (Text2 < 0) Then
      Label2.Caption = "Negative Number"

ElseIf Text2 = 0 Then
      Label2.Caption = "Your Number is Zero"

Else
      Label2.Caption = "Positive Number"

End If
End Sub
```

Selection: SELECT CASE

- Select Case statement → enables multiple choice
  - which 1st case that matches the criterion performed.
- Syntax:

```
Select Case test expression
Case expression list
  actions ... when expression list is TRUE
Case Else
  actions ... when all conditions are FALSE
End Select
```

REPETITION

Repetition: Introduction

- Selection/decision structures → offer options in making decisions and data selection
  - But do not provide program execution to continue until some condition is met.
- To enables program to execute continuously;
  - While structures
  - Do Structures
  - For Structure

Repetition: While

- **Three types of while structures:**
  - **While/wend**
  - **Do-While/Loop**
  - **Do-Until/Loop**

Repetition: While

Syntax:

1. While/Wend

***While* (condition)**
statements
counter=counter+1
Wend

2. Do-While/Loop

***Do While* (condition)**
statements
counter=counter+1
Loop

3. Do-Until/Loop

***Do Until* (condition)**
statements
counter=counter+1
Loop

Repetition: While

- **Example: While/Wend**

```
Dim x As Integer
x = Text3.Text
While x < 20
   x = x + 2
   Print x
Wend
```

Repetition: While

- **Example: Do-While/Loop**

```
Dim x As Integer
x = 1
Do While x < 20
    x = x + 1
Print x
Loop
```

Repetition: While

- **Example: Do-Until/Loop**

```
Dim x As Integer
x = 1
Do Until x > 20
    x = x + 1
Print x
Loop
```

Repetition: Do

- **Two types of Do structures:**
  - **Do/Loop-While**
  - **Do/Loop-Until**

Repetition: Do

- **Syntax:**

1. Do/Loop-While

Do

statements

counter=counter+1

Loop While **(syarat)**

2. Do/Loop-Until

Do

statements

counter=counter+1

Loop Until **(syarat)**

Repetition: Do

- **Example: Do/Loop-While**

```
Dim x As Integer
x = 1
Do
  x = x + 1
  Print x
Loop While x < 20
```

Repetition: Do

- **Example: Do/Loop-Until**

```
Dim x As Integer
x = 1
Do
  x = x + 1
  Print x
Loop Until x > 20
```

Repetition: While

- **Example: For**

```
Sub Command1_Click ()
    Dim iCount As Integer
    For iCount = 100 To 0 Step -10
    Print "iCount = "; iCount
    Next iCount
End Sub
```