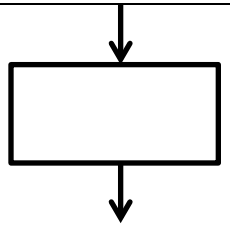
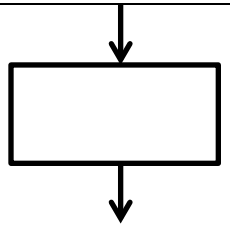
CIS1400 -- Programming Logic and Technique

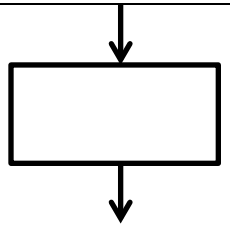
Statement Conversions

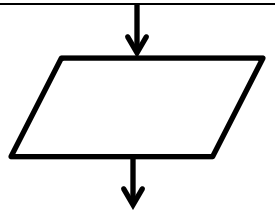
Sequential Statements

Statement	Pseudocode	Flowchart	Visual Basic

Variable Declaration	`Declare Integer varName`
	`Declare Real varName`
	`Declare String varName`
	`Declare Boolean varName`
			`Dim varName As Integer`
			`Dim varName As Double`
			`Dim varName As String`
			`Dim varName As Boolean`

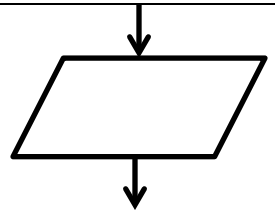
Constant Declaration	`Constant Integer constName = value`
	`Constant Real constName = value`
	`Constant String constName = "value"`
	`Constant Boolean constName = True or False`
			`Const constName As Integer = value`
			`Const constName As Double = value`
			`Const constName As String = "value"`
			`Const constName As Boolean = True or False`

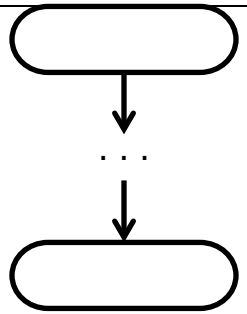
Assignment/Calculation	`Set varName = expression`		`varName = expression`

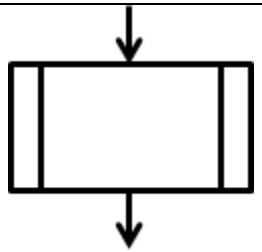
Input	`Input varName`		`varName = CInt(Console.ReadLine())`
			`varName = CDbl(Console.ReadLine())`
			`varName = Console.ReadLine()`

CIS1400 -- Programming Logic and Technique

Statement Conversions

Statement	Pseudocode	Flowchart	Visual Basic
Output	`Display "Enter a value: "` `Display "Your variable is ", varName`		`Console.Write("Please input a value: ")` `Console.WriteLine("Your variable is " & varName)`

Module Definition	`Module ModName*(dataType Ref** parmName)`
	`...`
	`End Module`
			`Sub SubName*(ByVal|ByRef parmName AS dataType)`
			`...`
			`End Sub`

Module Call	`Call SubName(argVal)`		`SubName(argVal)`

** multiple arguments/parameters separated by commas*

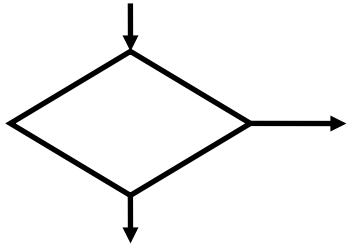
*** Ref keyword only needed for pass by reference parameters*

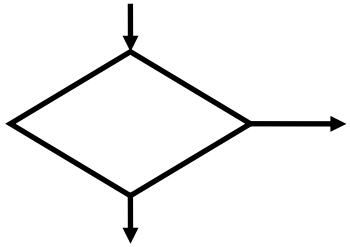
CIS1400 -- Programming Logic and Technique

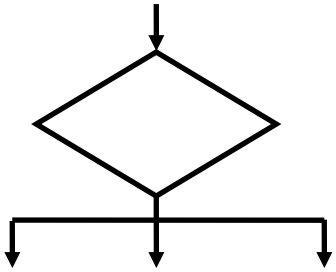
Statement Conversions

Selection Statements

Statement	Pseudocode	Flowchart	Visual Basic

Single Alternative	If *condition* Then
	  StatementsIfTrue
	End If
			If *condition* Then
			  StatementsIfTrue
			End If

Dual Alternative	If *condition* Then
	  StatementsIfTrue
	Else
	  StatementsIfFalse
	End If
			If *condition* Then
			  StatementsIfTrue
			Else
			  StatementsIfFalse
			End If

Multiple Alternative	Select *testExpr*
	  Case Val1:
	  StatementsIfVal1
	  Case Val2:
	  StatementsIfVal2
	  Default:
	  StatementsIfDefault
	End Select
			Select Case *testExpr*
			  Case *Expression1*
			  StatementsIfExpression1
			  Case *Expression2*
			  StatementsIfExpression2
			  Case Else
			  StatementsIfElse
			End Select