Terry Su
May 4, 2020

Structured Programming Notes Review

1. **Define structured programming.**

A structured program is one which uses only one in and one out type structures. Structured programming is the action of creating a structured program.

2. **Name the three structures in structured programming.**

Three structures in structured programming are:
- Sequence
- Decision
- Loop.

3. **Define algorithm.**

An algorithm is a finite series of ordered steps necessary to solve a problem. A computer program is the implementation of an algorithm.

4. **Define flowchart.**

A flowchart is a pictorial representation of an algorithm. For our purposes we only need 4 symbols (input, process, output and question) to represent any of the three structures.

5. **Logical operators work with _____ expressions.**

Logical operators work with boolean expressions.

6. **Complete the following truth table:**

a	b	c	(a and not b)	(not a or c) and not b
T	T	T	F	F
T	T	F	F	F
T	F	T	T	T
T	F	F	T	F
F	T	T	F	F
F	T	F	F	F
F	F	T	F	T
F	F	F	F	T

7. **Translate the following into valid Visual BASIC statements:**
   a. Is x between 0 and 1 (inclusive)?
   x >= 0 And x <= 1

   b. Is x positive and is y less than z?
   x > 0 And x = z - y

   c. Is k = -5, 0 or 5?
   k = -5 Or k = 0 Or k = 5

   d. Are x, y and z all equal to "y"?
   x = y And y = y And z = y

   e. Is it not the case that x and y are zero?
   Not(x = 0 And y = 0)

8. **What are the three different types of the if statement?**

Three different types of the "if statement" are:
   - If … (1 way)
   - If… Else (2 way) ,
   - If… ElseIf… (3+ ways)

9. **Which type of if would you use if you had to split something into 5 possible outcomes?**

If you had to split something into 5 possible outcomes you would use the <u>ElseIf</u> type as you require more than 2 possible outcomes.

10. **What does 'else' mean within an if statement?**

Else ('otherwise' in computer languages) is a block of code that is executed if the results of a previous test condition(s) such as If… or ElseIf … evaluates to false.

11. **What are two different types of loop?**

Two different types of loops are:
   - Do While… Loop
   - Do… Loop While

12. **What do all the structures have in common?**

All three structures in structured programming use the same, basic four components: input, process, output, and question. Additionally, they all rely on the same relational/logical operators while making decisions in the process.

13. **What is nesting?**

Nesting is a term referred to when codes are compiled of structures within structures. For example, an If statement within a Do While… Loop. Nesting is used to solve more complex problems.

14. What three things does a for loop do?

Three things a for loop do are:
- Initializes the control variable before the loop (eg count = 1)
- Checks the while condition at the beginning of the loop (eg Do While count <= 10)
- Increments the control variable at the end of the loop (count = count + 1)

15. A for loop is a special case of the _____ statement.

A for loop is a special case of the <u>Do While</u> statement

16. What are three very important questions to ask yourself when faced with a programming problem (this would occur during your Analysis of a problem)?

Three very important questions to ask yourself when faced with a programming problem are:
1. Can I picture clearly in my mind what the correctly working program will produce for output? (Do I understand the problem?)
2. What is the input into the program?
3. What structure(s) are involved? (sequence, decision or loop)
   - <u>Sequence:</u>
     a. What calculation(s) are required?
   - <u>Decision:</u>
     a. What different things need to be done
     b. What question do I ask (has to be yes/no answer)
   - <u>Loop:</u>
     a. What needs to be done inside the loop (what needs to be done over and over)
     b. When will the loop stop?
     c. Where does this exit condition go in the loop

17. What is tracing?

Tracing is the action to go through a program, step by step, recording the values of the variables as they change and any output produced. This allows you to better comprehend the purpose and function of the program.

18. What two things does tracing help you do?

Tracing helps you determine what a program does. This means it can help you to:
1. Determine what an unknown piece of code does.
2. Determine where the logic errors are in your own code. For example, perhaps a variable is expected to take on certain values at certain times and tracing may show that it's not not happening. When you see the problem you can find a way to correct it.

19. What's a bug? Who 'coined' this computer term?

A bug is an error in a computer programming that needs to be fixed. It was Grace Hopper that 'coined' this computer term after an early computer failure was found by her to be a moth stuck in a relay.

20. **What are the two different types of programming errors called? Clearly explain each of the error types.**

Two different types of programming errors are:
- <u>Syntax errors</u>: syntax errors are the technical errors in code such as incorrect spelling, grammar, and punctuation. A syntax error causes a failure in the execution of the program.
- <u>Logic errors:</u> a logic error is when you told the computer/program to do something wrong within its process of solving a problem. It causes the program to operate incorrectly but not crash. A logic error produces unintended or undesired output or other behaviour.

For the following questions, Good Style and a correct Algorithm will be used in determining the correctness of your response. You good variable naming and variable use wherever appropriate. Remember the 5 steps process of most algorithms: Declaration, Initialization, Input, Processing/Calculation, Output

21. **Write the code (a click event) which will print a message (to a label) depending on a value input by the user from a textbox, (if the number is 1, print "good", 0 print "bad" and anything else print "ugly").**

```
Private Sub cmdProcess_Click()
'Declare
Dim intInput As Integer
'Initialize
intInput = 0
'Input
intInput = Val(txtInput.Text)
'Process/Output
If intInput = 1 Then
    lblMessage.Caption = "good"
ElseIf intInput = 0 Then
    lblMessage.Caption = "bad"
Else
    lblMessage.Caption = "ugly"
End If
End Sub
```

22. Write the code (a click event) which will ask the user to enter "yes or "no" in an inputBox. Loop until the user does so (Classic Edit Loop). After the user enters "yes" or "no", print a message "great" (if the user entered "yes") or "Same to you fella" (if the user entered "no"). Check your notes on the use of an inputBox.

```
Private Sub cmdGo_Click()
 'Declare
Dim strYesOrNo As String
'Initialize
strYesOrNo = ""
'Input/Process/Calculations
picOutput.Cls
Do
   strYesOrNo = LCase(InputBox("Please, enter 'yes' or 'no'", "yes or no", "", 100, 100))
Loop While strYesOrNo <> "yes" And strYesOrNo <> "no"
If strYesOrNo = "yes" Then
   picOutput.Print "Great"
ElseIf strYesOrNo = "no" Then
   picOutput.Print "Same To You Fella"
End If
End Sub
```

23. Write the code (a click event) which will ask the user to enter 250 values and print the sum and average of all of those numbers.

```
Private Sub cmdGo_Click()
'Declare
Dim intTotal As Integer
Dim intInput As Integer
Dim intCount As Integer
'Initialize
intTotal = 0
intInput = 0
intCount = 0
'Input/Process/Output
Do
   intCount = intCount + 1
   intInput = Val(InputBox("Please, enter number " & intCount, "250 numbers", "", 100, 100))
   intTotal = intTotal + intInput
Loop While intCount < 250
picOutput.Cls
picOutput.Print "the total of the 250 numbers is " & intTotal
picOutput.Print "the average of the 250 numbers is " & Format(intTotal / 250, "0.00")
End Sub
```

24. **Write the code (a click event) which will ask the user to enter 50 values and will keep a count of those entered numbers which were greater than 10.**

```
Private Sub cmdGo_Click()
'Declare
Dim intMore10 As Integer
Dim intInput As Integer
Dim intCount As Integer
'Initialize
intMore10 = 0
intInput = 0
intCount = 0
'Input/Process/Output
Do
   intCount = intCount + 1
   intInput = Val(InputBox("Please, enter number " & intCount, "50 numbers, tracking all +10 numbers",
"", 100, 100))
   If intInput > 10 Then
      intMore10 = intMore10 + 1
   End If
Loop While intCount < 50
picOutput.Cls
picOutput.Print "There were " & intMore10 & " numbers that were greater than 10."
End Sub
```

25. **Write the for loop statement which will cause an integer variable, intX, to have the values that could be output.**
    a. **1,2,3,4,5,…, 53**

```
For intX = 1 To 53
```

    b. **5,4,3,2,1**

```
For intX = 5 To 1 Step  -1
```

    c. **5,10,15,20,…,150**

```
For intX = 5 To 150 Step 5
```

    d. **39,36,33,…,0**

```
For intX = 39 To 0 Step  -3
```