UNIT 2 LAB 1: PSEUDOCODE

Learning Objectives and Outcomes
- Use pseudocode to define a program's structure.
- Formulate solution algorithms for calculations by properly following the order of operations.

Required Setup and Tools
- Standard lab setup (hard drive with appropriate image)
- Lab Manual

Procedure
Complete the tasks from the bottom of page 9 and page 10 in the lab manual (attached). Do not worry about the missing video.

NOTE: Save a copy of the pseudocode as it will be used in Unit 3 Lab 3.2.

Submission Requirements
Submit your written response in email to srich@itt-tech.edu by the beginning of Unit 3 (January 10, 2015).

If email is absolutely not available, hand in your work at the beginning of Unit 3.


Lab 2: Input, Processing, and Output
This lab accompanies Chapter 2 (pp. 29-55) of Starting Out with Programming Logic & Design.

Lab 2.1 - Pseudocode

Critical Review

Pseudocode is an informal language that has no syntax rules and is not meant to be compiled or executed.

The flow the program takes is sequential. For example, before you ask for input, you should display what information you want from the user.

Follow the rules for naming variables: (1) must be one word, no spaces, (2) usually no punctuation characters, only letters and numbers, and (3) name cannot start with a number.

"Display" is the keyword used to print something to the screen. Any information needed to be displayed to the user should be put inside quotation marks such as Display "This is how you print something to the screen". When using display to print both a string and the value of a variable, a comma is used, such as Display "Here is the average", average.

"Input" is the keyword used to get the user to enter data. The data value entered by the user will be placed in the variable that follows the keyword input such as Input variableName.

"Set" is the keyword used before a calculation. Standard math operators are used, such as + - * / MOD ^. Operators can be combined in one calculation, but it is wise to group expressions together using parentheses. Remember the order of operations. Some examples are Set sale = price - discount and Set average = (test1 + test2 + test3) / 3.

Demo Video: View lab2-1.wmv in the Lab 2 folder on the accompanying Lab Demo Media and Startup Files CD.

This lab requires you to focus on variable assignment and calculations. Read the following program prior to completing the lab.

Write a program that will calculate the cost of installing fiber optic cable at a cost of .87 per ft for a company. Your program should display the company name and the total cost.

Step 1: Examine the following algorithm as a base for how the program should flow. (Reference: Designing a Program, page 31).
1. Display a welcome message for your program.
2. Get the company name.
3. Get the number of feet of fiber optic to be installed.
4. Multiply the total cost as the number of feet times .87.
5. Display the calculated information and company name.

Step 2: Think of good variable names for the following pieces of data that will need to be stored within this program.

<table>
<thead>
<tr>
<th>Purpose of Variable</th>
<th>Variable Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stores the cost of fiber</td>
<td>fiberCost</td>
</tr>
<tr>
<td>Stores the company name</td>
<td></td>
</tr>
<tr>
<td>Stores the number of feet to be installed</td>
<td></td>
</tr>
<tr>
<td>Stores the calculated cost of installed fiber</td>
<td></td>
</tr>
</tbody>
</table>

Step 3: Complete the following pseudocode based on the algorithm and the variables you declared above.

```
Display "Welcome to the Fiber Optic Calculator Program"
Set fiberCost = ?
Display "What is the company name?"
Input ?
Display "How many feet of fiber will be installed?"
Input ?
Set ?
Display "For the company", ?, "the total cost will be $", ?
```
UNIT 2 LAB 1: PSEUDOCODE

Learning Objectives and Outcomes
- Create the necessary flowcharts to describe a program's structure.
- Formulate solution algorithms for calculations by properly following the order of operations.

Required Setup and Tools
- Standard lab setup (hard drive with appropriate image)
- Lab Manual

Procedure
- Complete Lab 2.2: Flowchart from the lab manual.

Submission Requirements
Submit your flowchart, in Video or Raptor format, in email to srich@itt-tech.edu by the beginning of Unit 3 (January 10, 2015).
Lab 2.2 – Flowchart

Critical Review

A flowchart is a diagram that graphically depicts the steps that take place in a program. Symbols are used to depict the various steps that need to happen within a program. Flow lines are used between the symbols to indicate the flow of the program.

Ovals are used as terminal symbols, which indicate a start and stop to a program.

Parallelograms, the data symbol, are used for input and display statements.

Rectangles, the process symbol, are used for calculations and variable declarations.

On page connectors are used to link a flowchart that continues on the same page. The connecting system starts with the letter A, whereas A would appear in the two connectors that show the flow.

The statements inside the data and the process symbols can be written similarly to the statements used in pseudocode.

**Demo Video:** View lab2-2.wmv in the Lab 2 folder on the accompanying Lab Demo Media and Startup Files CD.
This lab requires you to think about the steps that take place in a program by designing a flowchart. While designing flowcharts can be done with paper and pencil, one mistake often requires a lot of erasing. Therefore, a flowcharting application such as Raptor or Visio should be used. This lab will give you a brief overview of Raptor. Read the following program prior to completing the lab.

Write a program that will calculate the cost of installing fiber optic cable at a cost of .87 per ft for a company. Your program should display the company name and the total cost.

Step 1: Launch Raptor or Visio and convert your pseudocode from Lab 2-1 into a flowchart. Depending on what application you are using, select either lab2-2raptor.rap or lab2-2visio.vsd from the Lab 2 folder on the accompanying Lab Demo Media and Startup Files CD to get started. Once you double click on them to open, do a Save As to save it to your workspace. Watch the demo video 2-2 for instructions on how to input, calculate, and output using both applications.

Step 2: The final step is to insert your finished flowchart into a Word document. Inside Raptor, select File and then Print to Clipboard from the menu. If you are using Visio, select Edit, then Select All, then Edit and Copy. In your Word document, select Edit and Paste.
UNIT 2 LAB 3: VISUAL BASIC

Learning Objectives and Outcomes

- Formulate solution algorithms for calculations by properly following the order of operations.

Required Setup and Tools

- Standard lab setup (hard drive with appropriate image)
- Lab Manual

Procedure

- Complete Lab 2.3: Visual Basic from the lab manual.

Submission Requirements

Submit your Visual Basic code (as a ZIP file) in email to srich@itt-tech.edu by the beginning of Unit 3 (January 10, 2015).
Lab 2.3 – Visual Basic

Critical Review

Console.WriteLine("...")) or Console.Write("...") is similar to the Display keyword in pseudocode. Console.WriteLine includes a return at the end of the display, and Console.Write leaves the cursor on the current line. We use these functions to write information to the screen. You can display string literals or string literals with variables such as:

```csharp
Console.WriteLine("The number is ", anumber)
Console.WriteLine("This is a sentence")
```

The & is used to connect a string literal with a variable.

Console.ReadLine() or Console.Read is used to take in input into a variable. Since variables must be declared before input can be used, this will be further explored in Lab 3.

Dim (meaning Dimensions) is the keyword to declare variables. We will set some values of variables in the lab 2-3, but will reserve explanation of data types such as Double, Integer, and String for Lab 3. To set values of variables in Visual Basic, use the following syntax.

```csharp
Dim dollars As Double = 2.75
Dim myAge As Integer = 35
```

Demo Video: View lab2-3.wmv in the Lab 2 folder on the accompanying Lab Demo Media and Startup Files CD

This lab will focus on how to assign variables and process calculations in Visual Basic.

Step 1: Create a new Visual Basic Console Application and save it to your workspace location.

Step 2: The first part of this program will convert the pseudocode from page 44 to Visual Basic. Under Sub Main() add the following line of code.

```csharp
Console.WriteLine("This part of the program will code Program 2-6 from page 44")
```

You can also use Console.Write() if you want the cursor to stay on the same line as your output.

Step 3: Following the steps on page 44, next declare dollars such as:
Dim dollars As Double = 2.75

Step 4: Using Console.WriteLine() code step 2 from Program 2-6 such as:

   Console.WriteLine("I have " & dollars & " in my account.")

Notice the & are used in Visual Basic and , are used in pseudocode.

Step 5: Reset dollars to 99.95 by adding the following:

   dollars = 99.95

Step 6: Convert step 4 from page 44 to display what is now in the account. You can follow the Visual Basic syntax from Step 4 of this lab.

Step 7: At the very bottom before End Sub add the following lines so that your program will Pause and you can see your output.

   Console.WriteLine("Press any key to continue")
   Console.ReadLine()

Step 8: Save and run your program so that it works without errors. Copy and paste your Visual Basic code into a Word document.