Description

This lab accompanies Chapter 3 of *Starting Out with Programming Logic & Design*.

Part 1: Questions

Complete the assignment in Topic 3: Lab B named, "Lab 3b Questions", in Canvas.

Part 2: Activity

Definitions

Algorithm

An algorithm is a set of well-designed logical steps that must take place in order to solve a problem.

Module

A Module is a group of statements that exists within a program for the purpose of performing a specific task.
Modules are commonly called procedures, subroutines, subprograms, methods, and functions.
The code for a module is known as a module definition. To execute the module, you write a statement that calls it.
The format for a module definition is as follows:

```
Module moduleName( )
      Statement
      Statement
      Etc.
End Module
```

Calling a module is normally done from the main ( ) module such as:

```
Call moduleName( )
```

Generally, local variables should be used and arguments should be passed by reference when the value of the variable is changed in the module and needs to be retained. For example:

```
Module main( )
      Declare Real number
      Call inputData(number)
      Call printData(number)
End Module

//accepts number as a reference so the changed value
//will be retained
Module inputData(Real Ref number)
      Display "Input a real number: "
      Input number
End Module

//number does not to be sent as a reference because
//number is not going to be modified
Module printData(Real number)
      Display "The number is ", number
End Module
```

Function

A function definition is similar to a module definition but includes the return type of the function. Here is the general format:

```
Function DataType FunctionName(ParameterList)
  statement
  statement
  etc.
  Return value
End Function
```

A function must have a Return statement. This causes a value to be sent back to the part of the program that called the function.

The parameter list is the list of variables that the function uses to take input. Following, is an example of a function called, “add”. It has two parameters. The each parameter is listed as so:

```
Function Integer add(Integer value1, Integer value2)
```

Calling a function is done in order to make the module execute:

```
Set sum = add(25, 44)
```

A function can have local variables defined within. Variables can either be local or the function can access variables in the global scope. A local variable is created inside a function and cannot be accessed by statements that are outside a function unless they are passed. A local variable that needs to be used in multiple functions should be passed to the necessary functions.

An argument is any piece of data that is passed into a function when the function is called. A parameter is a variable that receives an argument that is passed into a function. A global variable can be accessed by any function within the program but should be avoided if possible.

Problem Description

Write a random Lottery number choosing utility for the California state lotter called, “Fantastic 5”. This utility prompts the user for the upper range of the numbers. Assume that the range starts at 1. Your program will choose 5 random numbers and display them for the user. They do not need to be displayed in order. You must break your code up into single-use modules and functions. Avoid the use of global variables.

Create the pseudocode for this program first. You should write the pseudocode in a text file called, “fantastic5.txt”.

After you have created the pseudocode, implement the program in C++. Call the program file, “fantastic5.cpp”. The programs output should look like this:

```
Enter the maximum allowed lottery number: 100 <Enter>
Your numbers are: 2 53 33 98 4
```

What to Submit

Submit the following documents to the module named "Topic 3: Lab 3b" using the "Lab 3a Activity" assignment:

- The pseudocode file called, “fantastic5.txt”
- The source code file called, “fantastic5.cpp”.
- A screenshot of the execution of your program. **Include the entire window of the editor in which you wrote your code and executed it**. Be sure not to include other things in the image like your desktop.