Note: All the code that you write for the following labs must be uploaded to Github.

C++ Pass by Reference

- Implement a program with a function, named **SwapNumbers** that swaps the value of two integers as shown in the code below. This function must have two parameters. In this regard, the arguments for the **SwapNumbers** function must be passed in by reference using references. Then build and execute your program.

```
int main(){

     int varA = 25;
     int varB = 100;

     cout << "varA before swap: " << varA << endl; //varA is 25
     cout << "VarB before swap: " << varB << endl; //varB is 100

     SwapNumbers(varA, varB);

     cout << "varA after swap: " << varA << endl; //varA is 100
     cout << "VarB after swap: " << varB << endl; //varB is 25

     return 0;
}
```

- Reimplement the program above, this time the arguments to **SwapNumbers** function must be passed in by reference using pointers. Then build and execute your program.

C++ Classes

1. Create a program as follows:
    a. Your program must have a class name **Rectangle**
    b. The Rectangle class must have the following **private** member variables:
        - The length of a rectangle (as a floating point value)
        - The width of a rectangle (as a floating point value)
    c. The Rectangle class must have a **default** constructor only in which both member variables are initialized to zero
        - Remember to also include a destructor, which does nothing for now
    d. The Rectangle class must have the following **public** member functions
        - Accessor methods for assigning values to each of the two member variables
        - Accessor methods for retrieving values from each of the two member variables
        - A member function which calculates and returns the area of a rectangle as a floating point value, using the two member variables.
    e. Within **main**
        - Create an instance of a rectangle using the Rectangle class
        - Use the accessor methods to assign values for the length and width of the rectangle object
            - The length and width values must be provided via user input on the console
        - Output onto the console, the area of the rectangle object. This area must be calculated using the rectangle object's member function specified in **d**.
    f. Note the following:
        - The declaration of the Rectangle class must be in its own a header file
        - The implementation the Rectangle class must be in its own cpp file
        - The main function must be implemented in its own cpp file
    g. Build and execute your program

2. Make the following additions to the program in **1**
    a. Add an overloaded constructor to the Rectangles class
        - This constructor must be used to initialize the length and width variables
    b. Create another instance of a rectangle in main, using the Rectangle class
        - With this newly added instance, the length and width variables must be assigned using the overloaded constructor
            - These values must be provided via user input on the console
    c. Output onto the console, the area of this newly created rectangle object, similar to what was specified in **1(d)**.

3. Create a program as follows:
    a. Your program must have classes for the following shapes
        - A square
        - A triangle
        - A circle
    b. Each of the classes specified in **a** must be declared within a namespace named **shapes**
    c. Each of the classes must have **private** member variables as follows:
        - The square class
            - Side length
        - The triangle class
            - Base
            - Height
        - The circle class
            - Radius
    d. Each class specified in **a** must have appropriate accessor methods for the member variables specified in **b**
    e. Each of the classes specified in **a** must have a default constructor, and an overloaded constructor, both of which must be used to initialize the member variables specified in **b**
    f. Each class must have destructor
    g. Declare and implement another class in your program named **Area**, as follows:
        - This class must **not** have any member variables
        - This class must **not** have any constructors
        - This class must have three **public static** member functions as follows:
            - Calculate the area of a square
                - This function must have one parameter whose type is the square class
                - This function must calculate and return the area of a square based on a square object passed into it
            - Calculate the area of a triangle
                - This function must have one parameter whose type is the triangle class
                - This function must calculate and return the area of a triangle based on a triangle object passed into it
            - Calculate the area of a circle
                - This function must have one parameter whose type is the circle class
                - This function must calculate and return the area of a circle based on a circle object passed into it

h. Within main, your program must do the following:
    - Your program must execute continuously until the user quits the program via user input on the console
    - Your program must provide the user with the following options on the output onto the console:
        - Calculate the area of a square
        - Calculate the area of triangle
        - Calculate the area of a circle
        - Quit
    - If the user selects the calculation of a particular shape, then the following must happen:
        - An object of that shape must be created using the appropriate class
        - The user must be prompted to enter the values to be assigned to the member variables of the created object. The values provided by user must be assigned to the object in question
        - The area of the created object must be calculated using the appropriate static member function from the Area class
        - The calculated area must be output onto the console
i. Note the following:
    - Each of the three classes must be declared its own a header file
    - Each of the three classes must be implemented in its own cpp file
    - The main function must be implemented in its own cpp file