CS 1428
Fall 2018
Gentry Atkinson
Lab 3

Introduction:

Over the last 3 labs you have learned to use variables, operators, loops, and branching statements. You will now expand your understanding of these coding building blocks by applying them in new ways. This week's lab will require the use of the Boolean operator **&&** (and) and || (or).

The purpose of this lab is to better understand how to combine variables, operators, loops, and conditional statements.

Directions:

1- Launch Code::Blocks and start a new file. Name it your_last_name_lab2.cpp.

2- Include the standard header for this lab:

```
//Your Name
//CS1428 Fall 2018
//Lab 3
```

3- Include the iostream standard library and start your main function:

```
#include <iostream>
using namespace std;
int main() {
```

4- Implement the following pseudo-code to create a simple calculator:

```
//DECLARE TWO INTEGER VARIABLES AND ON CHAR VARIABLE
//PRINT A SHORT WELCOME MESSAGE
//START A LOOP THAT CONTINUES WHILE BOTH INTEGERS ARE NOT 0
        //PROMPT THE USER FOR ONE INTEGER VALUE
        //PROMPT THE USER FOR A SECOND INTEGER VALUE
        //PROMPT THE USER FOR AN OPERATOR (+,-,* OR /)
        //PRINT THE VALUE OF THE OPERATOR ACTING ON THE TWO INTEGERS
                //EX. INPUT: 2, INPUT: 5, INPUT: +, DISPLAY: 7
//END LOOP
//PRINT A SHORT PARTING MESSAGE
```

5- Your client requires a simple program which can classify triangles. You should write a small segment of code which will repeatedly take input from a user for the lengths of the three legs of a triangle and then print **acute, right, obtuse, or invalid** based on the following relationships:

Obtuse	**c^2 > a^2 + b^2**
Right	**c^2 = a^2 + b^2**
Acute	**c^2 < a^2 + b^2**
Invalid	**c >= a + b**

Notice that c must be the largest of the three numbers. Your code should only come up with one classification for each each set of three numbers. You should create some method to exit the loop and inform the user of how to exit the loop when he or she is done input values.

6- Your client is happy with the code you produced for problem 5 but wants their in house development team to be able to maintain it. Go back through your solution to problem 5 and add a comment explaining each action that is being performed.

7- You want to calculate the sum of all odd numbers from 1 to 100. You are trying to find the most efficient way to do that. Copy the following code and fill in the bodies of the loops to calculate the number you want:

```
int output = 0;
int i, j, k;
for (i = 1; i < 100; i += 2){

}
cout << output << endl;
output = 0;
j = 1;
while (j < 100 ){

}
cout << output << endl;
output = 0
k = 0;
do {

}while (k != 100);
cout << output << endl;
```

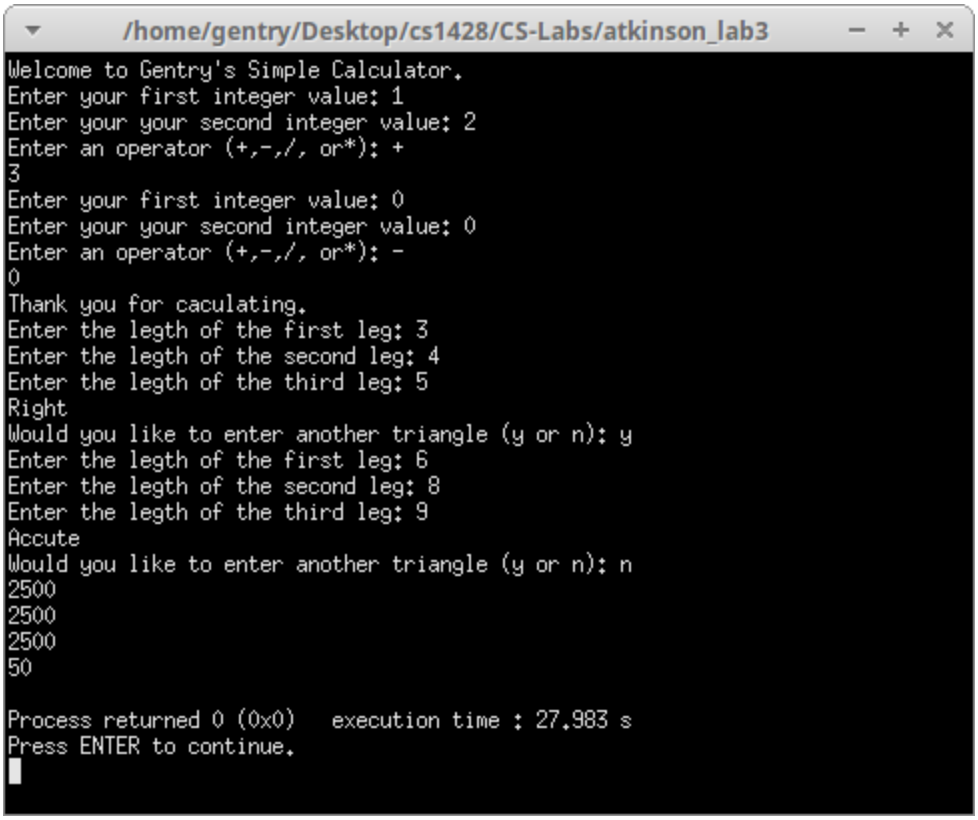
Your code should find the same value for **output** in each of the three loops. Do not change any of the code that's already been written. You should only fill in the bodies of the loops.

8- Do not copy the following code. Predict what its output will be after executing and then print your prediction on a single like using a **cout** statement.

```
int output = 0;
int j;

for(int i = 0; i < 10; i++) {
        j = 0;
        while (j < i) {
                output++;
                j++;
        }
}
cout << output << endl;
```

9- Save you work. Build and Run your file. Fix any errors. Your output should look something like this:



10- Submit your .cpp file through TRACS.