```cpp
/*================================================
Program: LoopFun
Author: Caleb Reister
Date: 10/31-11/4/2013
Dev Env: Visual Studio 2012
Description: Loop Practice
    - Get two int values from user for width and height
    - Check to make sure values entered are in valid range
    - Print numbers between width and height
    - Create a rectangle outline out of '*' with a width and height
        corrosponding to the numbers entered
    - Create a triangle from '*' with a height corrosponding to the user input
        - The width at the tip has two '*' and each layer below adds 2 '*'
================================================*/

#include <iostream>
#include <iomanip>

using namespace std;

void main()
{
    int width;//user input
    int height;//user input
    int horizontal;//used in multiple loops (reset each time), declared here to
        //prevent the repeated creation and deletion of variables for the same purpose
    int vertical;//used in loops

    //============================================

    //get width and height
    cout << "ENTER WHOLE NUMBERS ONLY..." << endl
        << "Please enter the height of a rectangle (between 3 and 10): ";
    cin >> height;
    cout << "Please enter the width of a rectangle (between " << height << " and 20): ";
    cin >> width;

    //============================================

    //check input for errors
    //attempted to compensate for any random input, but did not work
    //could be possible if auto variable type were used
    while (width < height || width > 20 || height < 3)
    {
        cin.clear();//clear fail state in case the user entered an invalid typ
        if (height < 3 || height > 10)//check height
        {
            cout << "The height you entered is out of range, try again: ";
            cin >> height;
        }
        if (width < height || width > 20)//check width
        {
            cout << "The width you entered is out of range, try again: ";
            cin >> width;
        }
```

```
56      }
57
58      //================================================
59
60      cout << endl;
61
62      //show intermediate values between width and height
63      for (int BoxDiff = height; BoxDiff <= width; BoxDiff++)
64      {
65          //created BoxDiff can change without messing up the box values
66          cout << BoxDiff << " ";
67      }
68
69      cout << endl << endl;
70
71      //================================================
72
73      //create rectangle
74      for (horizontal = 1; horizontal <= width; horizontal++)//1st horizontal line
75      {
76          cout << "* ";
77      }
78      for (vertical = 2; vertical <= (height - 1); vertical++)//vertical lines
79      {
80          cout << setw((width - 1) * 2) << endl//set width of print to 2 * number of '*' to
                account for ' '
81              //in horizontal lines, subtracted 1 to offset width so it doesn't create '*'
                    past end of rectangle
82              << left << "*" << right << "*";
83      }
84      cout << endl;
85      for (horizontal = 1; horizontal <= width; horizontal++)//2nd horizontal line
86      {
87          cout << "* ";
88      }
89
90      cout << endl << endl;
91
92      //================================================
93
94      //create triangle
95      cout << "* *";//triangle always starts with * *
96      horizontal = 4; //added 2 to number of * above
97      for (vertical = 1; vertical <= (height - 1); vertical++)//vertical position of curosr
98      {
99          cout << endl;
100         for (int start = 1; start <= horizontal; start++)//content for each line
101             //start defines starting point of * creation keeps adding "* " until
                    horizontal is reached
102         {
103             cout << "* ";
104         }
105         horizontal += 2;// add to number of '* ' to create after each line, +=2 was
                assigned
106     }
```

```
107
108     //=====================================================
109     //keep console open
110     cin.ignore();
111     cin.get();
112 }
```