```cpp
/*
PROGRAM: FunctionFun
AUTHOR: Caleb Reister
DATE: 2013-11-11
DEV ENV: Visual Studio 2012 Pro
DESCRIPTION: Identical to LoopFun but is function-based
 */

#include <iostream>
#include <iomanip>

using namespace std;

int GetInt(int low, int high);
void RangeMean(int low, int high);
void box(int height, int width);
void triangle(int height);

int main()
{
    int height;
    int width;

    cout << "Enter a box height (between 3 and 10): ";
    height = GetInt(3, 10);
    cout << "Enter a box width (between " << height << " and 20): ";
    width = GetInt(height, 20);

    cout << endl << endl
        << "Intermediate values between " << height << " and " << width << ":" << endl;
    RangeMean(height, width);
    cout << endl << endl << endl;

    box(width, height);
    cout << endl << endl;
    triangle(height);

    cin.get();
    cin.ignore();
}

int GetInt(int low, int high)
{
    /*
    FUNCTION: GetInt
    DESCRIPTION: uses cin to get an integer within a range

    ARGUMENT LIST:
        - low: the starting value of the range
        - high: the final value of the range

    RETURN VALUES:
        - input if the number is in range, loops until a valid value is entered otherwise
    */
    int input;
```

```
56      cin >> input;
57      while (input < low || input > high)
58      {
59          if (input < low)
60              cout << "The number you entered is too low, try again: ";
61          else if (input > high)
62              cout << "The number you entered is too high, try again: ";
63          cin >> input;
64      }
65      return input;
66  }
67
68  void RangeMean(int low, int high)//show intermediate values between high and low,
        calculate average
69  {
70      /*
71      FUNCTION: RangeMean
72      DESCRIPTION: prints the intermediate values and average of the arguments
73
74      ARGUMENT LIST:
75          - low: start of the range
76          - high: end of the range
77
78      RETURN VALUES: void
79      */
80      for (int IntMed = low; IntMed <= high; IntMed++)//IntMed is the intermediate value to
            be printed
81          cout << IntMed << " ";
82      cout << endl << "Average: " << (low + high) / 2;
83  }
84
85  void box(int width, int height)
86  {
87      /*
88      FUNCTION: box
89      DESCRIPTION: creates a box out of '*' with the set width and height
90      ARGUMENT LIST:
91          - width: the box width
92          - height: the box height
93
94      RETURN VALUES: void
95      */
96      int horizontal;
97      int vertical;
98      for (horizontal = 1; horizontal <= width; horizontal++)//1st horizontal line
99          cout << "* ";
100     for (vertical = 2; vertical < height; vertical++)//vertical lines
101         cout << setw((width - 1) * 2) << endl//set width of print to 2 * number of '*' to
                account for ' '
102             //in horizontal lines, subtracted 1 to offset width so it doesn't create '*'
                    past end of rectangle
103             << left << "*" << right << "*";
104     cout << endl;
105     for (horizontal = 1; horizontal <= width; horizontal++)//2nd horizontal line
106         cout << "* ";
```

```
107 }
108
109 void triangle(int height)
110 {
111     /*
112     FUNCTION: triangle
113     DESCRIPTION: prints a triangle starting with "**", adding "**" each row
114     ARGUMENT LIST:
115         - height: the number of rows to print
116     RETURN VALUES: void
117     */
118     int vertical;
119     int horizontal = 4;
120     cout << "* *";//triangle always starts with * *
121     for (vertical = 1; vertical < height; vertical++)//vertical position of curosr
122     {
123         cout << endl;
124         for (int start = 1; start <= horizontal; start++)//content for each line
125             //start defines starting point of * creation keeps adding "* " until
                    horizontal is reached
126             cout << "* ";
127         horizontal += 2;// add to number of '* ' to create after each line, +=2 was
                assigned
128     }
129 }
```