Lab 07 – Tree of Integers

# Problem

Assigned the task of creating a tree of integers. Needed to code methods including : insert, printInOrder and GetDepth. Need to code the tree in a manner that sorts the values, so that the methods will make sense to code.

# Proposed Solution

Create a sorted tree of integers that will place the integers in a sorted manner while insert, then will traverse the tree correctly when attempting to print them in order. The depth method should check the looked for value with the position in the tree, and then go left or right depending on < or >, or return the depth if we have found the value. If we reach the end of the tree without finding the value, then return -1.

# Tests and Results

After running the program 10 times, was able to get correct in order traversals and the correct depth returns of 4 and -1 for the values 6 and 14 respectively. Trees are difficult.

# Problems Encountered

Had some major issues with the depth method, until I realized I was incrementing my int value for depthNum twice in a place due to recursion. Upon fixing this error, my program worked fine.

# Conclusions and Discussions

Good practice **again** with recursion. Trees are complex, but it is great that they are presorted based upon an insert method.

# Additional Questions

**Lab Report Questions:**

1. Demonstrate one-by-one how the insertion process work for the following number 54, 90, 72, 59, 87, 15, 16, 83, 25, and 27.

Root

Step One: 54

Step Two : 90

Step Three: 72

Step Four: 59

Step Five: Root

54

90

72

59 87

Step Six: Root

54

15

Step Seven: Root

54

15 90

16 72

59 87

Step Eight: Root

54

15 90

16 72

59 87

83

Step Nine: Root

54

15 90

16 72

25 59 87

83

Step Ten: Root

54

15 90

16 72

25 59 87

27 83