Lab 08 – Array Tree of Integers

# Problem

Assigned the task of creating an array tree of integers. Needed to code methods including : insert, printInOrder, printBreadth and GetDepth. Need to code the tree in a manner that sorts the values, so that the methods will make sense to code. Will need to populate the array with a sentinel value ( I chose -1) to ensure that we sort print the whole tree.

# Proposed Solution

Create a sorted array tree of integers that will place the integers in a sorted manner with 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. The print breadth tree should start each “branch” printing left to right, but not print if the value is the sentinel value.

# Tests and Results

Correct have all methods working. Insert is putting the numbers in a sorted manner, printInOrder is printing the numbers in correct ascending order, printBreadth order is printing each line left to right, but not printing any -1 values, and getDepth is returning the depth of the searched for value if found, or returning -1 if not found.

# Problems Encountered

Had an issue with my insert method, my original for loop was not functioning and thus the array was filled with the value of 0 instead of the sentinel value of -1. After fixing the insert method, fixing the problems with my print in order method and my get depth method were easily rectified.

# Conclusions and Discussions

I enjoy arrays, and the way that a tree inserts things presorted is a great change of pace than having to write a sort method for every program. The printBreadthOrder is really only useful to figure out how the tree is populated in a branch / child pattern.

# Additional Questions

**Lab Report Questions:**

1. Show how the array will look after each step when inserting the numbers {14, 17, 7, 20, 5, 11, 15}

Step One: Root

14

Step Two: Root

14

17

Step Three: Root

14

7 17

Step Four: Root

14

7 17

20

Step Five: Root

14

7 17

5 20

Step Six: Root

14

7 17

5 11 20

Step Seven: Root

14

7 17

5 11 20

15

(Removal of 14 from tree)

Step Eight: Root

7

5 17

11 20

15