Simon Chau
SID: 862048772
schau014@ucr.edu

Assignment 3

2.

Insert function:

Time Complexity: O(log(n)), Space Complexity: O(1)

This function has a space complexity of O(1) since it only adds one new node to the tree. Building the tree with n nodes has a complexity of O(n) though.

Preorder function:

Time Complexity: O(n), Space Complexity: O(n)

The space complexity of this function is O(n) since the function call takes O(1) space, but it recursively visits each node in the tree, so it is actually O(n*1) = O(n).

FindMax function:

Time Complexity: O(n), Space Complexity: O(n)

This function is in place, as there is a new Node called MaxNode is declared and set to the highest valued node while the function traverses the tree. The space complexity is O(n) since each function call has the space of O(1) but the function visits all n nodes. So O(n * 1) = O(n).

MakeBST function:

Time Complexity: O(nlog(n)), Space Complexity: O(n)

When this function is called, it is not sorted in place, and std::sort actually has a space complexity of O(log(n)). When added with O(n) from the size of the vector needed to store the integers and strings of the nodes, the space complexity becomes O(n + log(n)) which is O(n).

3. Times for each function in seconds

Size	Build Tree	Preorder()	FindMax()	MakeBST()
100	0.000322978	0.002532	3.127e-6	0.002498
1000	0.010955028	0.00389	2.309e-5	0.119528
10000	0.475599963	0.036074	0.000105981	9.143192
100000	54.901834672	18.820266	0.002209046	N/A

My MakeBST function time for 100000 elements is N/A since I keep getting a seg fault. I'm pretty sure my call stack is over 1 mb with a size of 100000 and I was reading that on windows systems, programs terminate once this is violated.

Note: my test.cpp should print the times in nanoseconds or microseconds. I just converted them to seconds in the table to keep it uniform!