Patrick Walsh

ECE241 Project 1

Informational Questions on Project

5. Time spent on buildBST(): 0.00601 seconds

6. Time spend on searchBST():0.0 seconds <NO NOTICEABLE TIME DETECTED>

- As the height is only 11 with all 1113 members of the BST, that means it will only take a maximum of 11 recursive calls for the function to find its stock. As you can make educated skips during the BST search but you still have to register some points and cannot just skip to the item in one go like with Hash tables, the Time Complexity for searchBST is O(logn). I imagine if there were hundreds or thousands of rows in the BST we would see noticeable time detected.

7. Time spent on linearsearch() with 100 terms: 0.003 seconds

8. Time spent on searchBST(): 0.0 seconds <NO NOTICEABLE TIME DETECTED>

9. With the times received by my system, about two linear searches would equal up to approximately the time spent building the BST.

10. The system I used to find the longest name was a simple for statement with O(n) time. If the system found a stock with a longer name than the variable length, length would change to that variable and print the index of the item. In this case, Semiconductor Manufacturing International Corporation had an index of 674.

11.

- Highest percent change: name: Gamestop Corporation; symbol: GME; val: 1710.0; price:325.00
    - 1784.05797% increase

- Lowest percent change: name: Cogent Inc.; symbol: COGT; val: 2233.0; price:8.77
    - 28.699186991869922% decrease