

BTKeysAtSameLevel2 Write-Up

Isaac Abboudi

The big O of the program should be linear, or $O(n)$. The bulk of the program is contained within the single main loop that iterates over the sequence of characters within the String *treeInStringRepresentation*.

At the start of the method we instantiate a char array -the cost of this is $O(n)$ as it's an array copy- and the List of lists that will be returned. Then we enter the main loop:

-keep in mind we are tracking the depth as we go, starting at 0-

For each character in the array

If it is an integer 1-9

 then add to the integer list that lives at index[depth] of the master list

else if its an opening parenthesis -> '('

 increment depth

else if its a closing parenthesis -> ')'

 decrement depth

When I tested the doubling ratio of the program I kept receiving results between 1.3 and 1.8, never consistently 2. I don't know that there are other factors I'm missing but even so I would assume they'd push the results to farther from 2 rather than close to 1.