Trees
Burak Öztürk
21901841
Section 1
Assignment 2

Do your findings related to average number of rotations in the AVL tree agree with theoretical results?

Theoretical number of rotations in an AVL tree for k insertions:

For ascending and descending order, calculation is simple.

If we start inserting integers in order and write down each insertion's caused rotations down, we will notice a pattern:

```
0
01
0121
01212121
0121212121212121
```

(Note that every time tree becomes a full BST, I continued in a new line which happens in every insertion with number (2^i-1), $i > 0$)

This pattern happens because AVL trees work in a recursive loop that causes one of the subtrees of a node to act like it's parent did on the prior loop. Only difference between ascending and descending order is which subtree will new item be added to. Thus, ascending order and descending order have mirror characteristics.

So number of rotations for k insertions is

Let sum = 0. For each insertion number i (range [1, k]), if i != 2^a where a is any positive integer and if i is odd add 1 to sum. Else if i != 2^a where a is any positive integer and if i is even add 2 to sum.

Resulting sum is the number of rotations.

My results:

Array Size	Random	Ascending	Descending
1000	699	990	990
2000	1403	1989	1989
3000	2061	2988	2988
4000	2744	3988	3988
5000	3381	4987	4987
6000	4038	5987	5987
7000	4671	6987	6987
8000	5300	7987	7987
9000	5944	8986	8986
10000	6563	9986	9986

Results are pretty close but doesn't match up one to one for some reason that I can't deduce.

For random insertions, theoretical average results are harder to calculate but upper and lower boundaries are easy.

Assume that we have a set of numbers to insert to an AVL tree. If we insert the median and divide the list from the median value's place and apply this to the subsets as well recursively at the same time, the tree will always stay complete and never will need rotations. This is the lower bound that is zero. Obviously my results are all above zero.

Worst case for rotations is actually is (or very close to) ordered insertion and my random results are all below that as well.

My weighted average (weight: array size) of rotations for every insertion is 0.67 and it falls into the boundaries. As well as every one of the results.

Do different patterns of insertion affect the number of rotations in the AVL tree? If so, explain how. If not, explain why not.

It does. I explained best and worst cases for rotation above. Since both of those explanations rely on patterns of insertion of values to an AVL tree, it is clear that insertion pattern has a big effect on number of rotations in the AVL tree.