Homework Assignment #1

Due: January 16, 2020, by 5:30 pm

Jiahong Zhai 1005877561

Question 3

We can do this like merge sort.

We first split the tree with smaler number of nodes recursively, by split a big tree into two small trees, until we get a lot of trees with at most 3 nodes. We put these new small trees to the bottom level of the other tree's half branch. If the tree been splited is B1, then we make new tree to the left child of the nodes in the bottom level of B2's left branch(left branch means root's left descendants). If the tree been splited is B2, then we make new tree to the rigt child of the nodes in the bottom level of B1's right branch(right branch means root's right descendants).

Split the tree in to half recursively takes O(logn) time, and heiht of a tree is logn, and we split the tree with smaller # nodes, so WC timecomplecity is $O(\min(h_1; h_2))$.

The # nodes at bottom level ≈ half of nodes in the tree. We put the tree with smaller # nodes to the bottom level of the tree with bigger # nodes. So it won't fill the complete level. So we add at most 1 level. So, the new tree's height is at most $\max(h1, h2) + 1$.