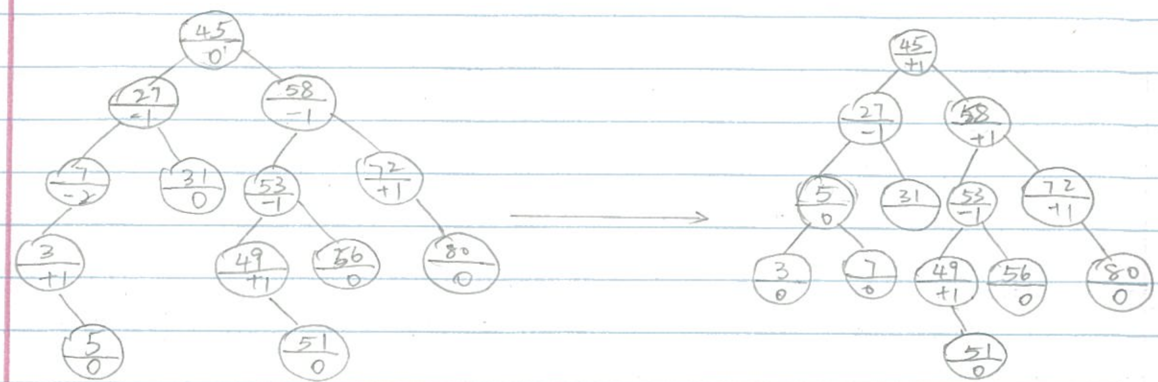
Name: Xiao. Houze

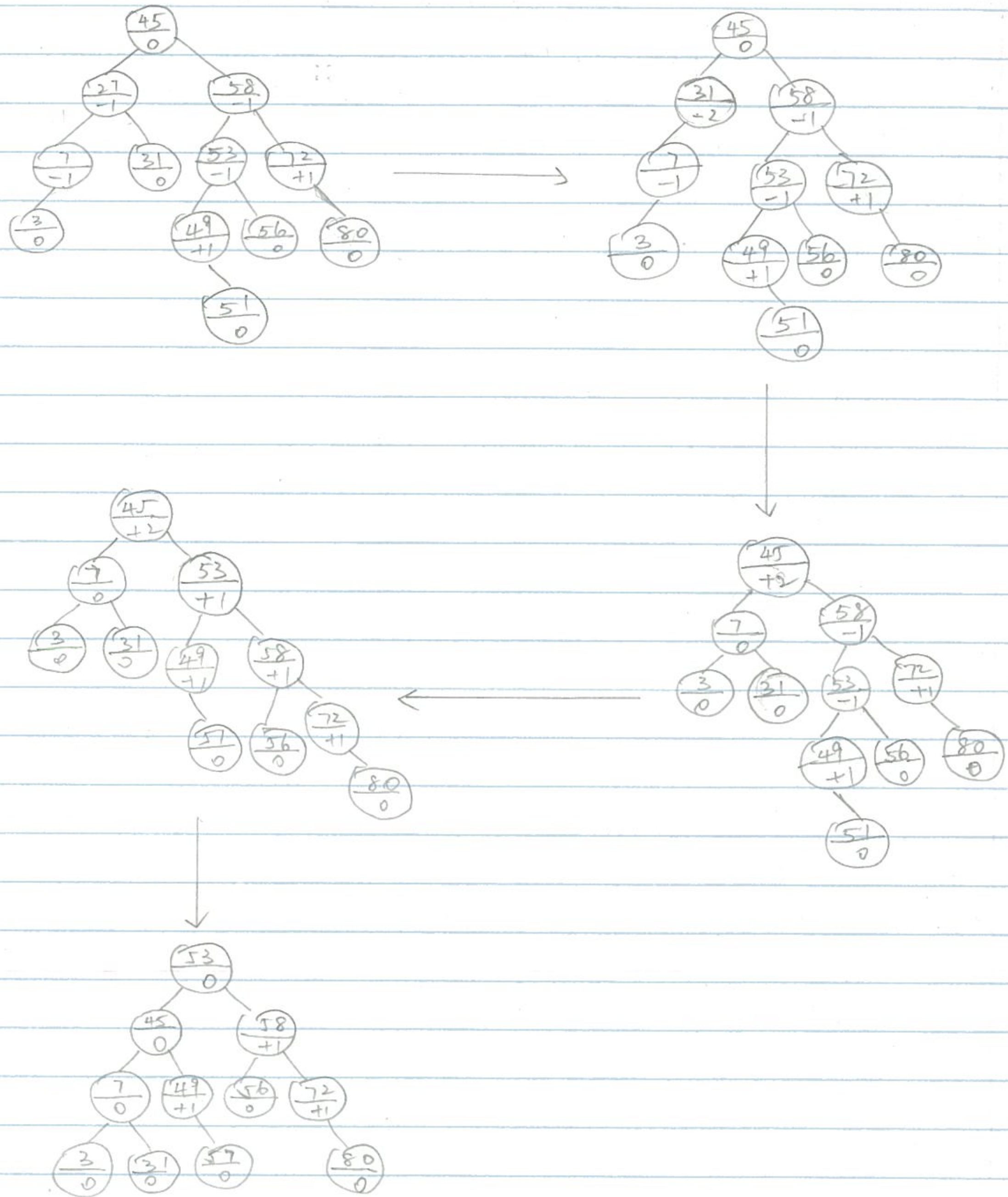
ID: 20508228

CS 240 Assignment 3.

1. a).



b).

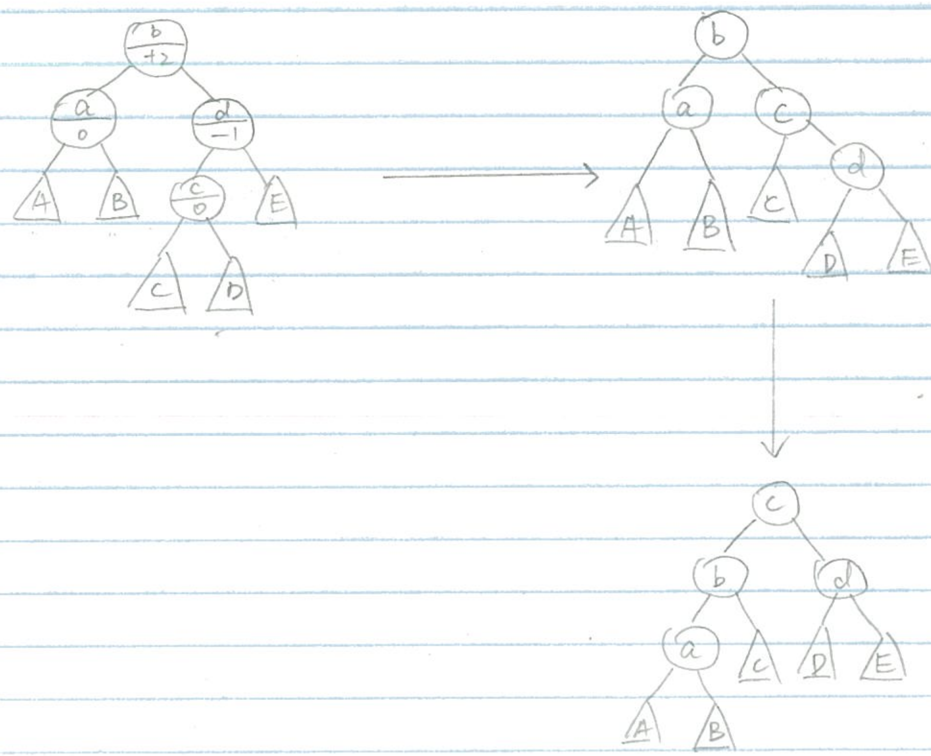


c). For example, for a foo tree with number of node 7

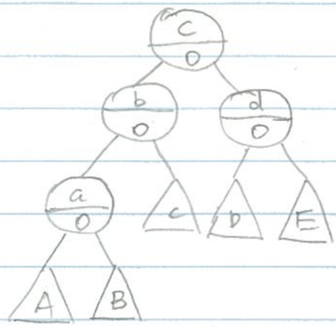
We can observe that if we fill each node from left to right at each level of the tree, we will get the minimum height, which is $\lfloor \lg 7 \rfloor$. If we want to get the maximum of a foo ~~tree~~ with 7 nodes, there must be a node from the second last level with balance factor ± 2.
Thus, the least level of foo tree is trying to fulfill every level from left to right, the best case is $\Theta(\lfloor \lg n \rfloor)$.
For the worst case, there is a node from the last second level of a foo tree with balanced factor ± 2, so height is $\Theta(\lfloor \lg n \rfloor + 1)$.
Thus, a foo tree with n nodes has height $O(\log n)$.

d). Height (T):
1. If T. left = T. right = nil, return 0.
2. If T. balance = 1, return Height (v. right)
3. If T. balance = -1, return Height (v. left)
4. If T. balance = 0, return Height (v. left) // or return Height (v. right).

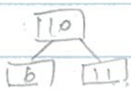
2 a).



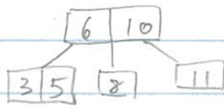
b).



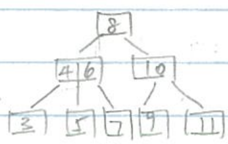
3a).



b).



c).



d). # of nodes is $2 + 6 + 18 + 54 + \cdots$

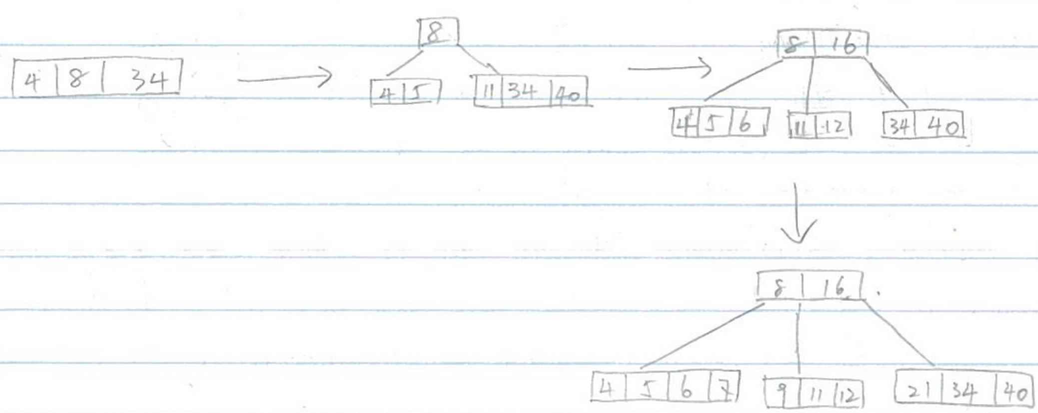
By using geometric series, $n < 2\frac{1-3^h}{1-3}$ where h is the height of the given 2-3 tree.

By solving the equation,

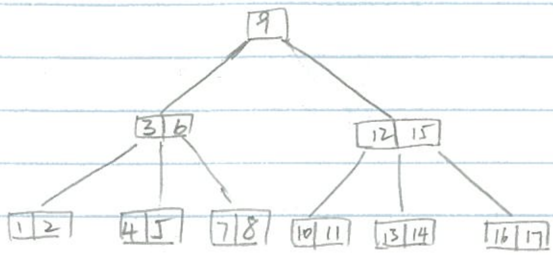
$$n < 2 \cdot \frac{3^h - 1}{2} = 3^h - 1$$

$$\therefore \quad h = \lceil \log_3 (n+1) \rceil$$

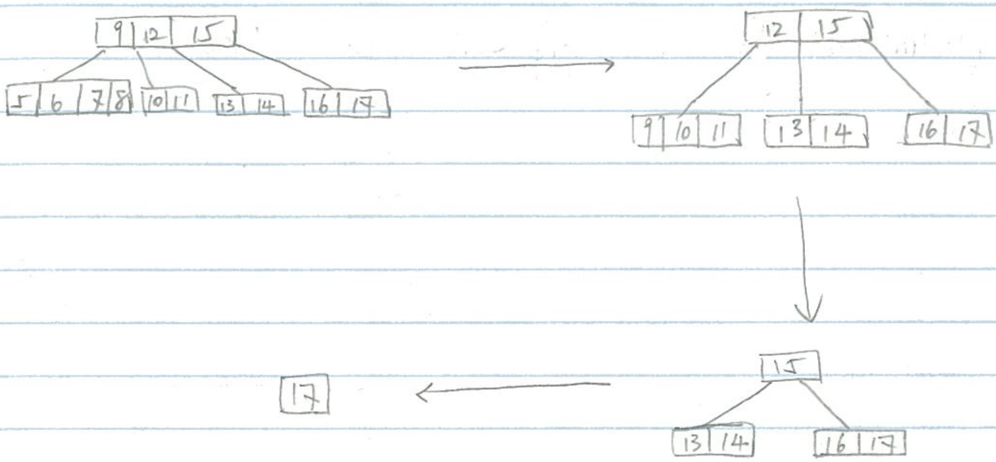
4 a).



b).



c).



5. Using counting sort to solve this problem, creating an initially empty array B in size K. For each value in A, increase the corresponding index of B by 1. After finishing going through all values in A, we go to B, then make each element in B as the sum of all previous elements.

Runtime: $O(n) + O(K) = O(n+K)$ to create B.

To determine how many integers are in range $[a, b]$, we just need to simply do $B[b] - B[a-1]$. We use $B[a-1]$ instead of $B[a]$ because we will not include the lower bound after subtracting $B[a]$. If $a=b=0$, we just return 0. If $a=0$, $b \neq 0$, return $B[b]$. Else, return $B[b] - B[a-1]$.

Runtime: 2 constant runtime for worst case.

1 constant runtime for best case.