## İsmet Alp Eren / 21703786 / CS-202 / Section-01 / HW-2

## Question 1-)

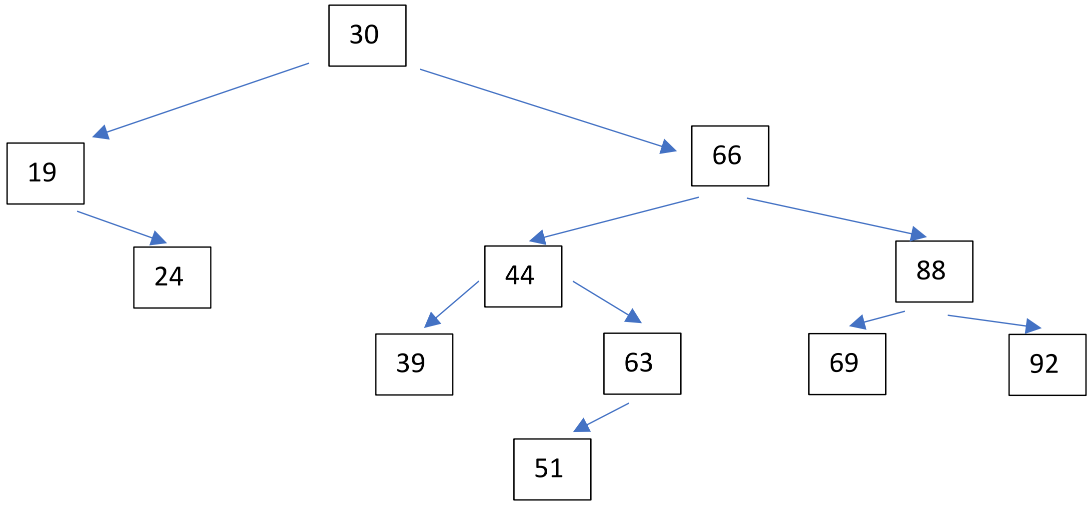
Preorder of tree: *-(13)^(3)(2)+(4)/(21)(7)

Inorder of tree: (13-(3^2))*(4+(21/7)) = 4*7 = 28

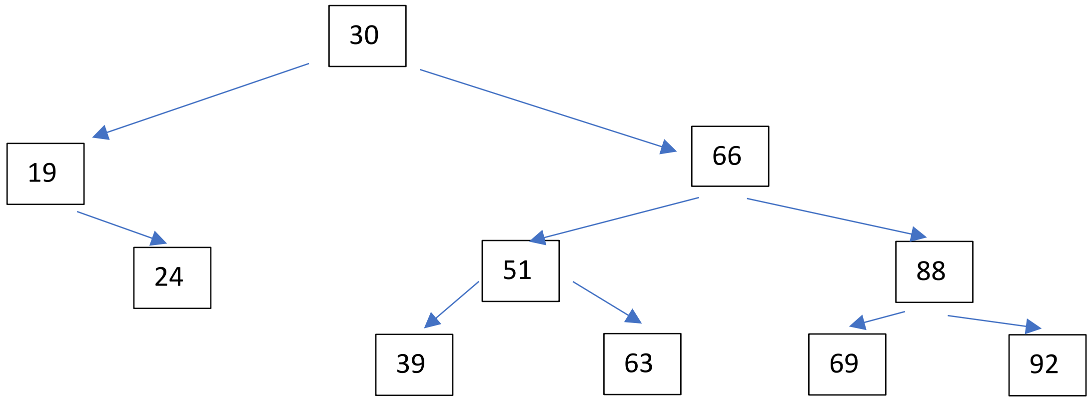
Postorder of tree: (13)3(2)^-4(21)(7)/+(25)*

## Part b-)

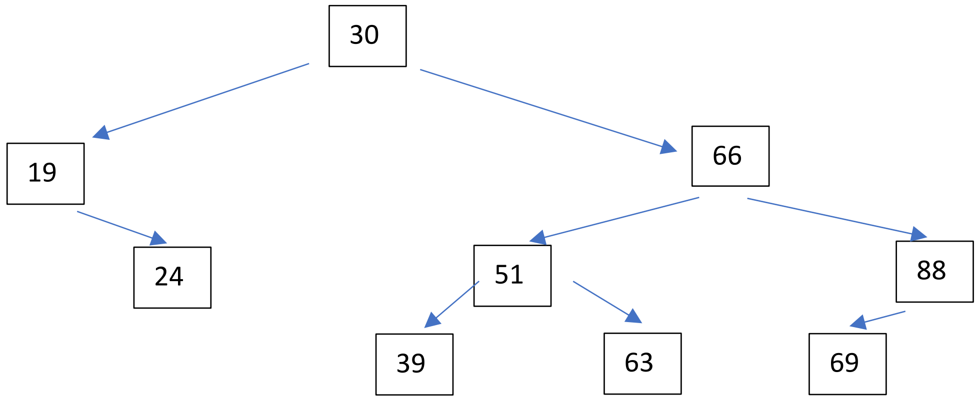
After all additions



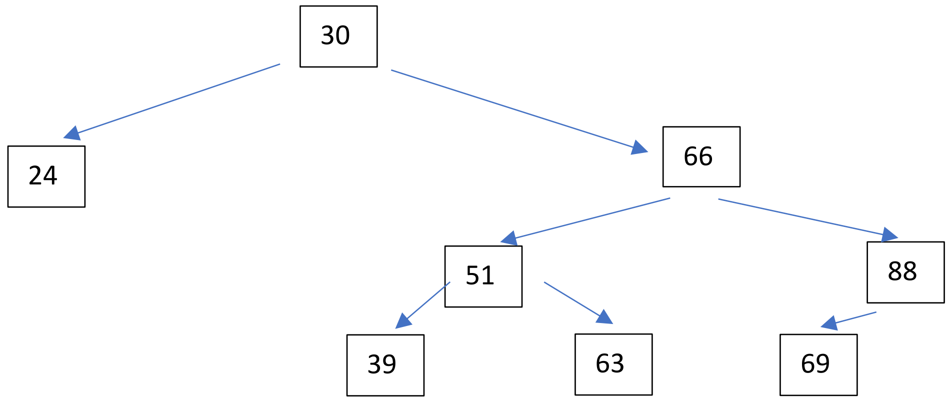

Delete 44



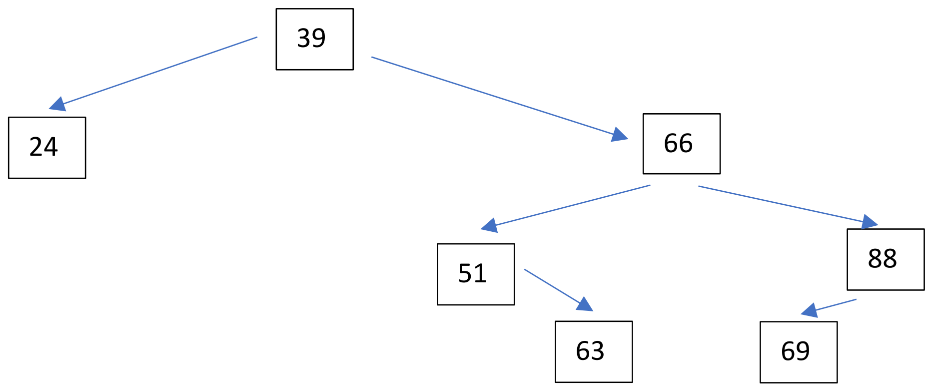
Delete 92



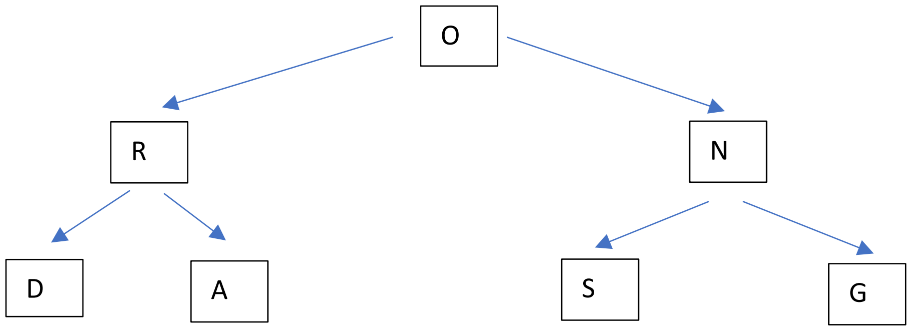

Delete 19




Delete 30

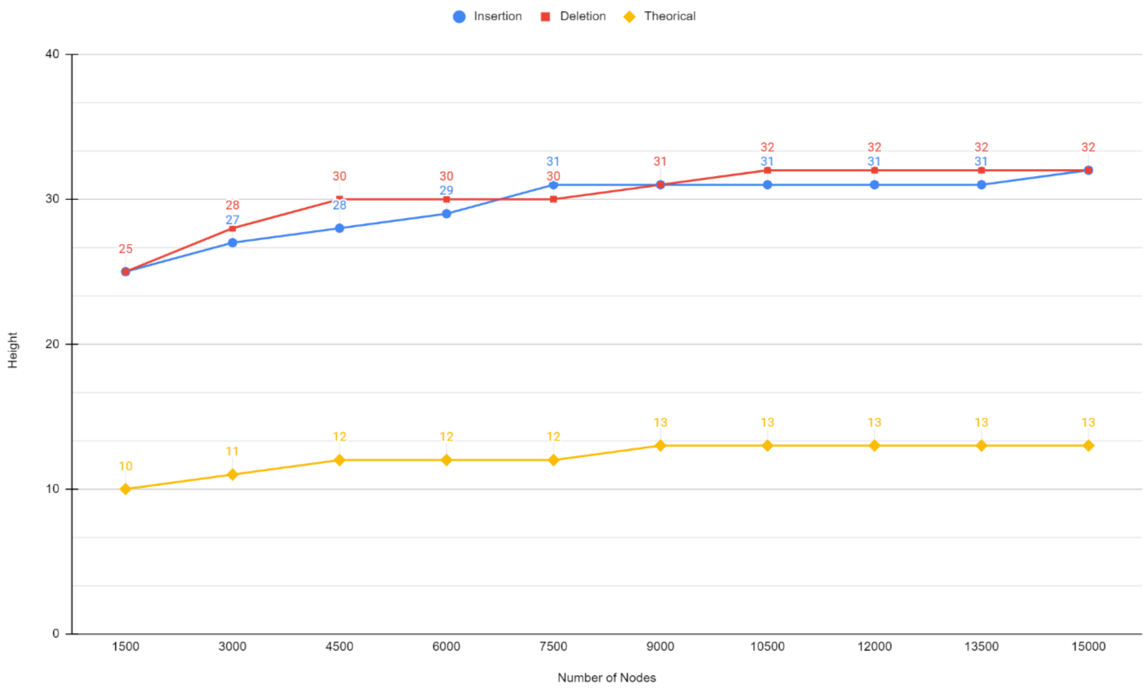


**Part c-)**




Inorder traversal is = D,R,A,O,S,N,G

**Question 3-)**




The theorical results is much less than experimental ones because the theorical values are best case for a binary search tree. $\sum_{k=0}^{n} 2^k$ = ($2^n$-1) where n is the height of the tree for the best case. $\sum_{k=0}^{12} 2^k$ = 8191, with height of 12, binary search tree can have 8191 nodes. $\sum_{k=0}^{13} 2^k$ = 16383.

The difference occurs because we inserting the keys randomly, the height is between worst and best case where n is the worst case and $\sum_{k=0}^{n} 2^k$ is the best case.

If we insert the keys as a sorted way the height of the tree would be n. For example, if number of nodes is 1500 and the numbers are ascending order then height of tree will be 1500 and all nodes will be right child of its parent. For the other cases, the situation is same, their height will be n.