/**
* Title : Binary Search Trees
* Author : Deniz Yüksel
* ID: 21600880
* Section : 1
* Assignment : 2
* Description : Answers to question 1 and 3 of to the assignment
*/

ANSWER TO QUESTION 1)

a)

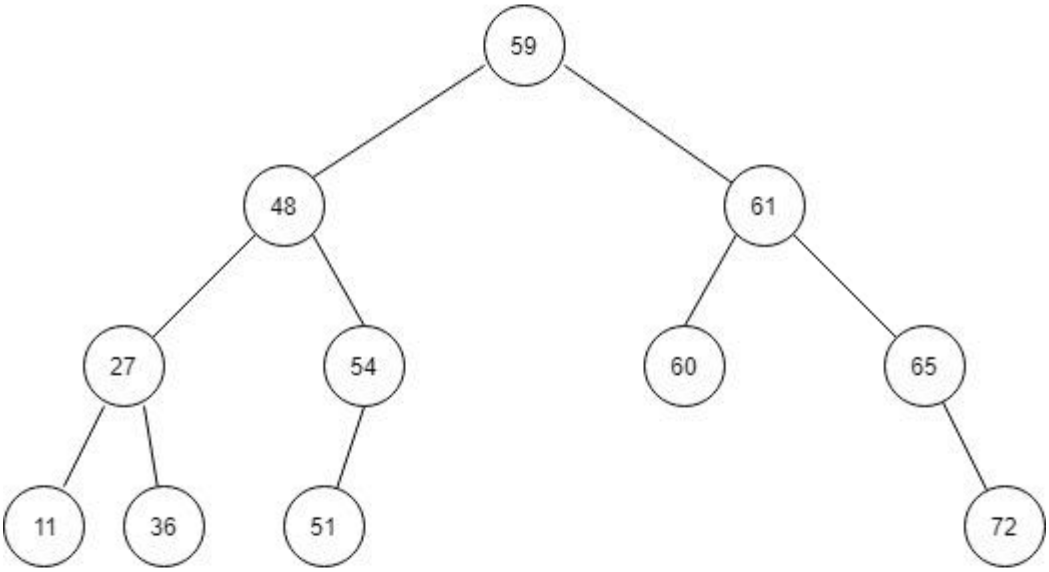
preorder: /, -, 3, *, 5, 8, +, ^, 4, 1, 7

inorder: 3, -, 5, *, 8, /, 4, ^, 1, +, 7

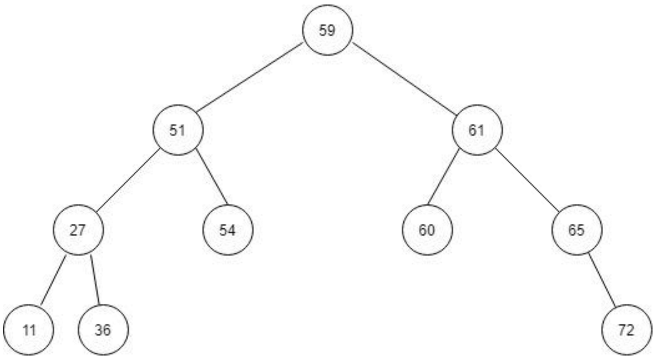
postorder: 3, 5, 8, *, -, 4, 1, ^, 7, +, /

evaluation of the mathematical expression: [3 – (5 * 8)] / [4^1 + 7] = -37 / 11

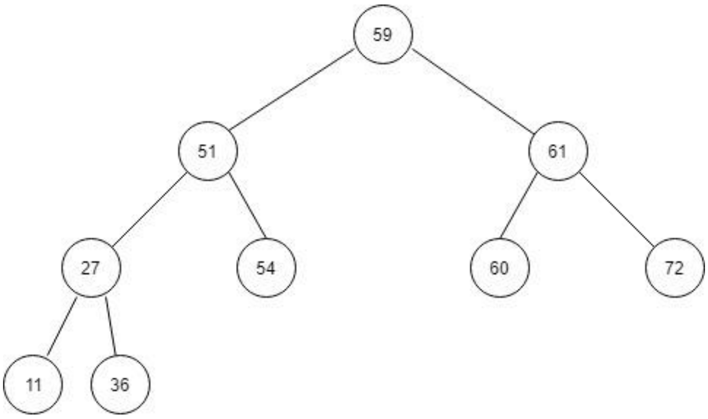
b) Initial tree after insertions:



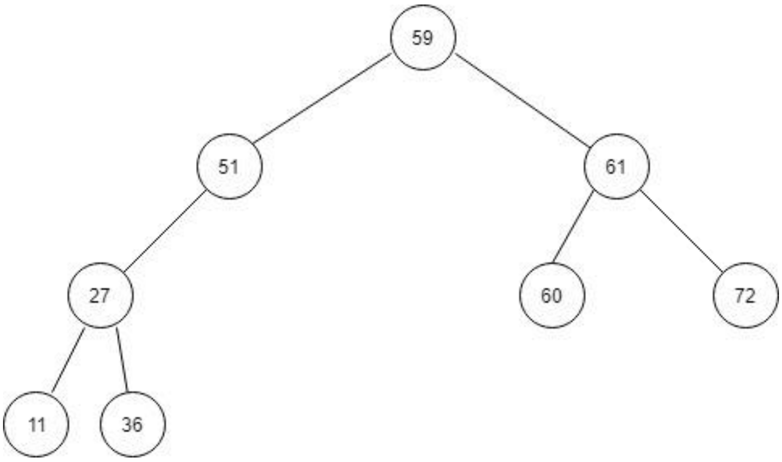
Tree after deletion of 48:



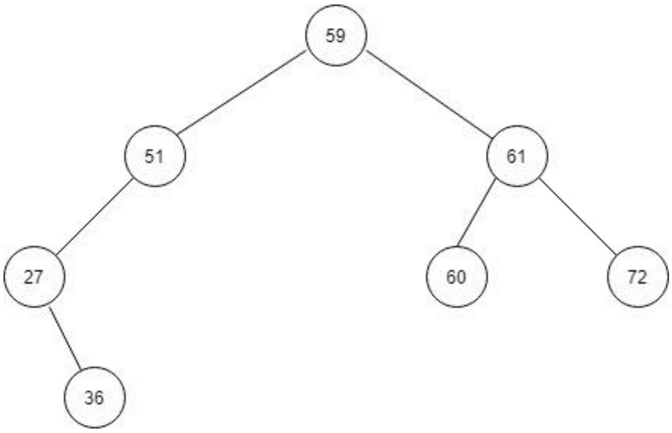
Tree after deletion of 65:



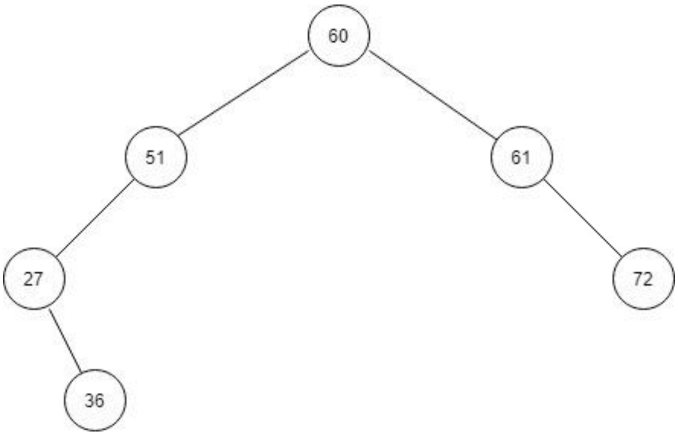
Tree after deletion of 54:



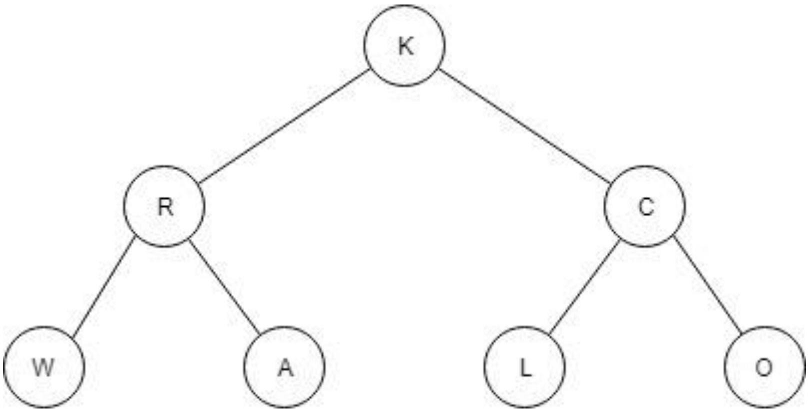
Tree after deletion of 11:



Tree after deletion of 59:



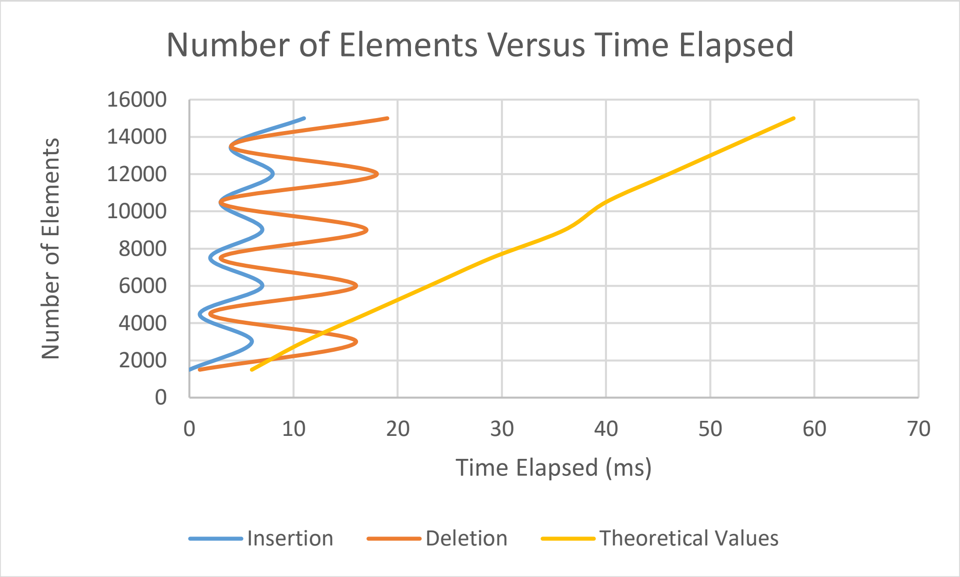
c) Constructed tree from its postorder traversal:



Inorder traversal of the tree: W, R, A, K, L, C, O

ANSWER TO QUESTION 3)

My computer's processor is 2.60 GHz so when I convert it to miliseconds, its processor speed is close to 3.846 milisecond. I calculated the worst case plot data from multiplications of 3.846 with 1500,3000… , 15000 because the worst case of insertion and deletion is O(n).



The graph abow is the Number of Elements Versus Time Elapsed graph. It has 4 series of insertion, deletion, theoretical insertion and theoretical deletion. In theory, the insertion and deletion is both O(n) for the worst case. For theoretical insertion, the graph is linear in worst case as the number of elements are increasing in the array, as the tree has more nodes. With a same behavior, theoretical deletion is also O(n) and it takes longer time to delete elements if the tree is larger. Insertion in my case has a wavy function since array elements are random, it does not have the worst case as O(n) but we are favoring from the randomness. We also do not know if our tree is balanced or not, but in general as in the graph, the blue line shifts slowly to the right which implies that it is taking more time. The same occurence is for deletion. If there are more elements, the orange line is shifting to the right. However deletion takes longer time than insertion does in the graph. There are more operations in deletion in which there are 3 cases we discussed in the class. However insertion is just one more operation after finding the correct place.

If we had a sorted array, I think that the tree would be perfectly unbalanced because every element would be lesser or higher than the previous one. Then it would behave like an array and we cannot favor from the randomness. Therefore in my opinion, it would get closer to the theoretical worst case line.