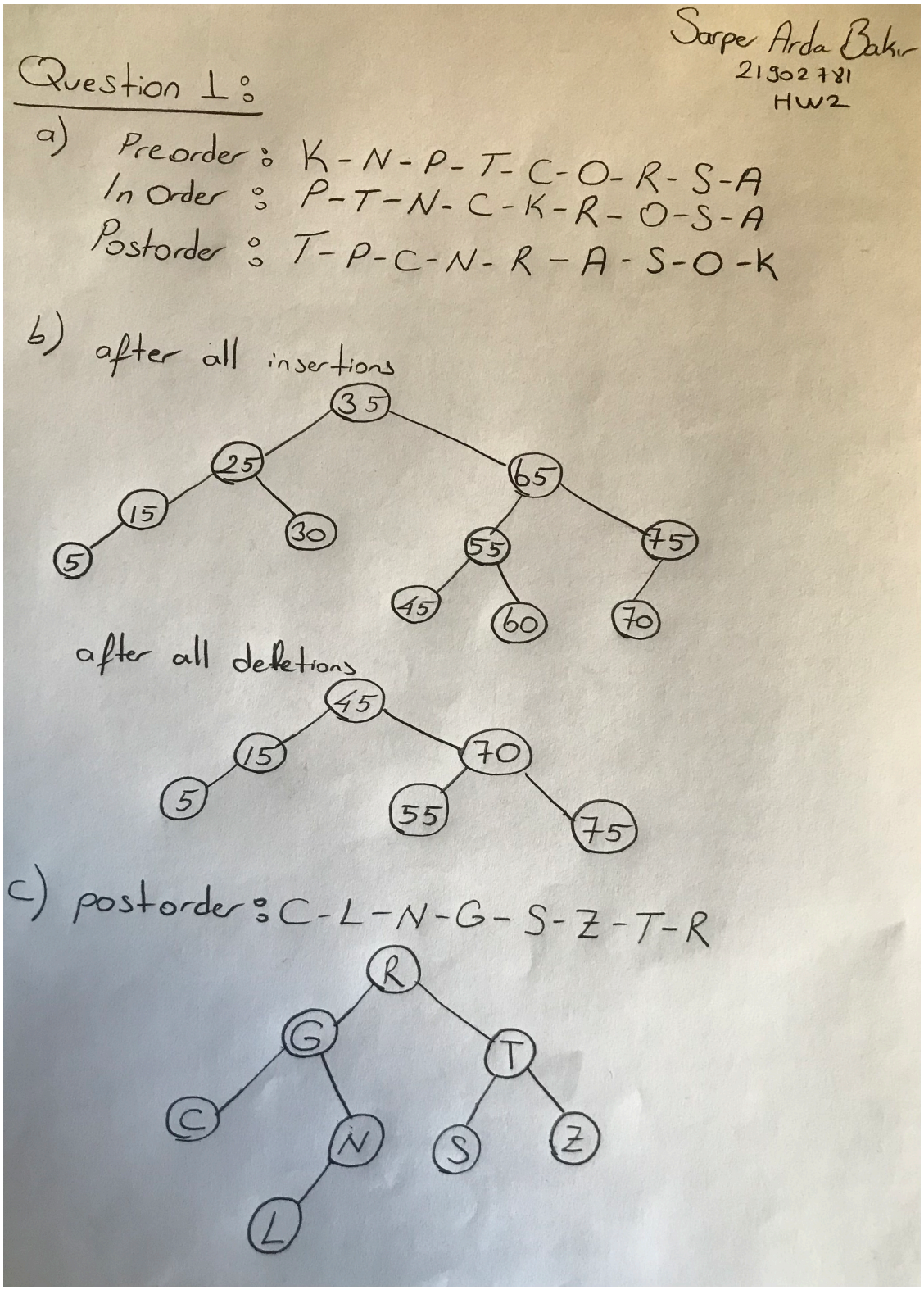
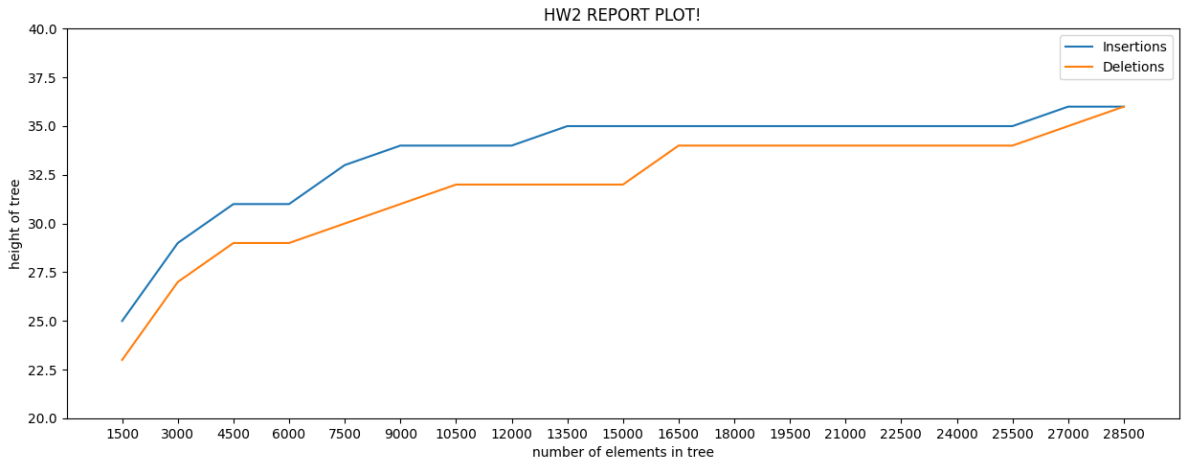
CS 202, Fall 2022

Homework 2 – Binary Search Trees

Name Surname: Sarper Arda Bakır Id:21902781



Question 3:



In this homework, I examine the Binary Search Trees and their heights compared to their number of nodes. Firstly, I created the random numbers, which are between zero and 30000, and store in array. Then, I inserted this numbers to Binary Search Trees one by one. I printed Tree's height in every 1500 new nodes are inserted. After that, I shuffled the array 30000 times and I deleted the nodes randomly from Tree. I also printed Binary Search Tree's height in every 1500 new nodes are deleted. In this way, I got results by height Analysis() method. When I compared empirical results, they seems similar with the theoretical ones. Deletion nodes' height are much closer than insertion nodes' height to theoretical height. However, empirical results is still higher than theoretical one. Even so, both results' Big-O notation is O(logn). If I inserted sorted numbers into it instead of randomly generated numbers, Binary Search Tree's height should be O(n); because every new inserted nodes should goes to root's right pointer. To sum up, randomly inserted nodes is much efficient than sorted nodes inserted. In addition, theoretical Binary Search Tree's height is O(logn).