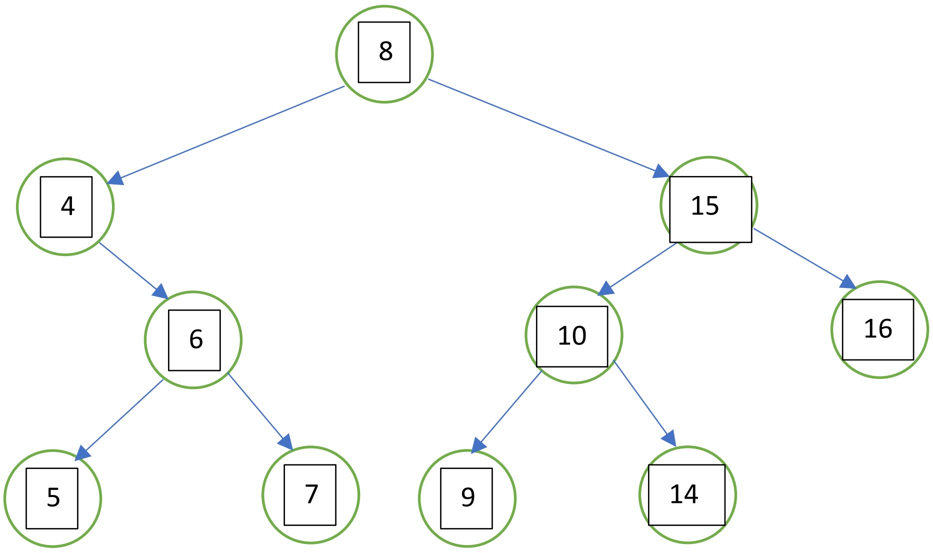
Sertaç Derya Sec3 22003208 HW2

Question 1:

a)



b)

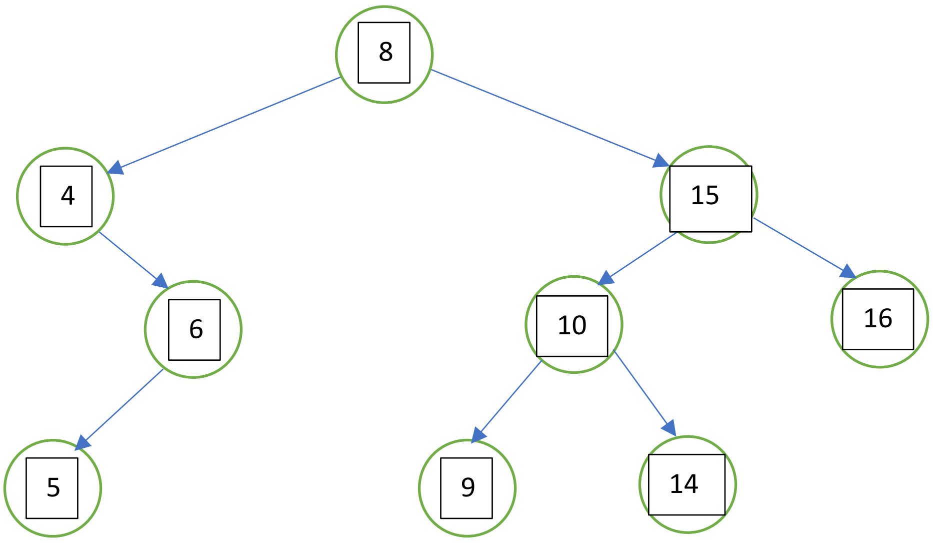
Predorder: 8, 4, 6, 5, 7, 15, 10, 9, 14, 16

Inorder: 4, 5, 6, 7, 8, 9, 10, 14, 15, 16

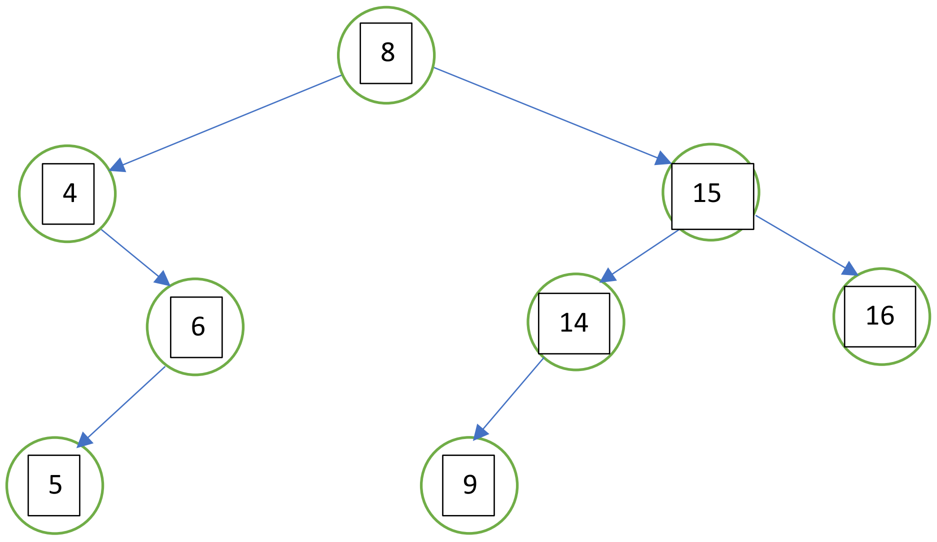
Postorder: 5, 7, 6, 4, 9, 14, 10, 16, 15, 8

c)

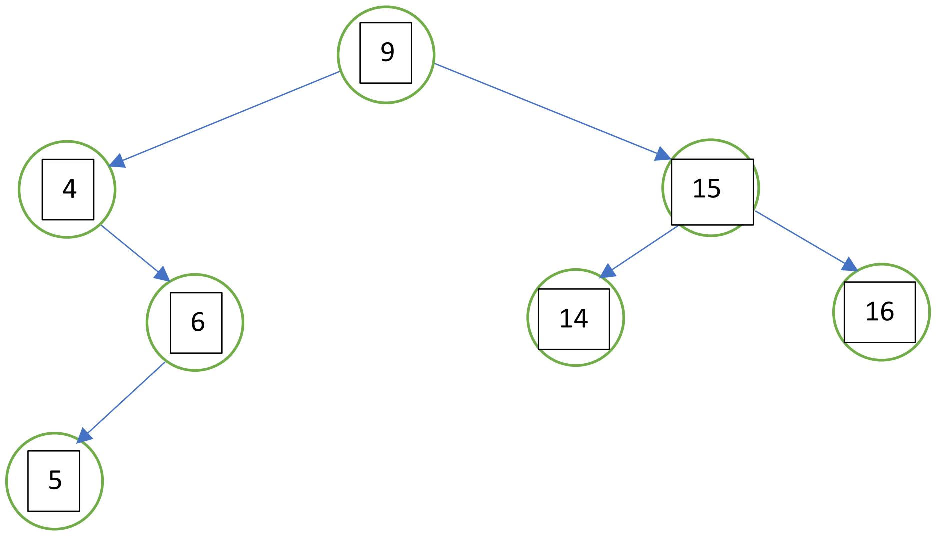
delete 7:



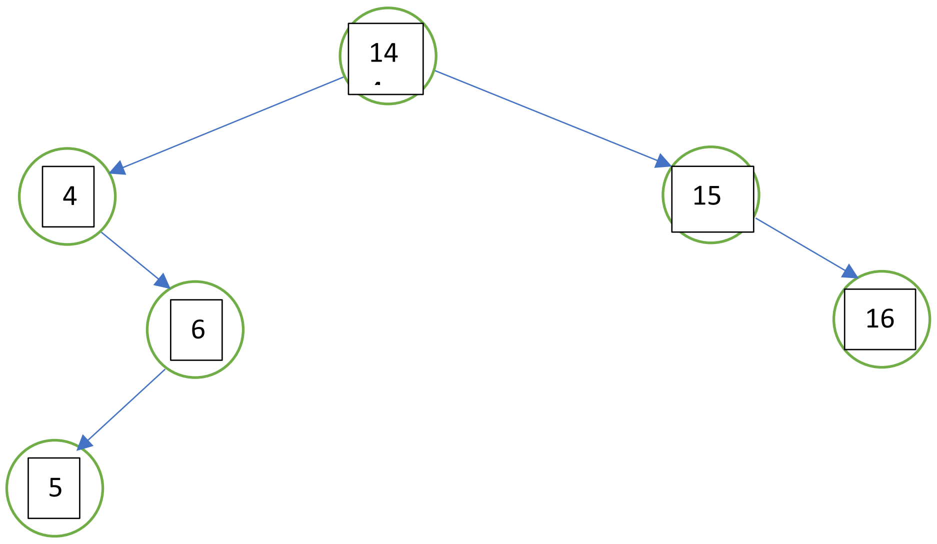
delete 10:



delete 8:



delete 9:



delete 5:

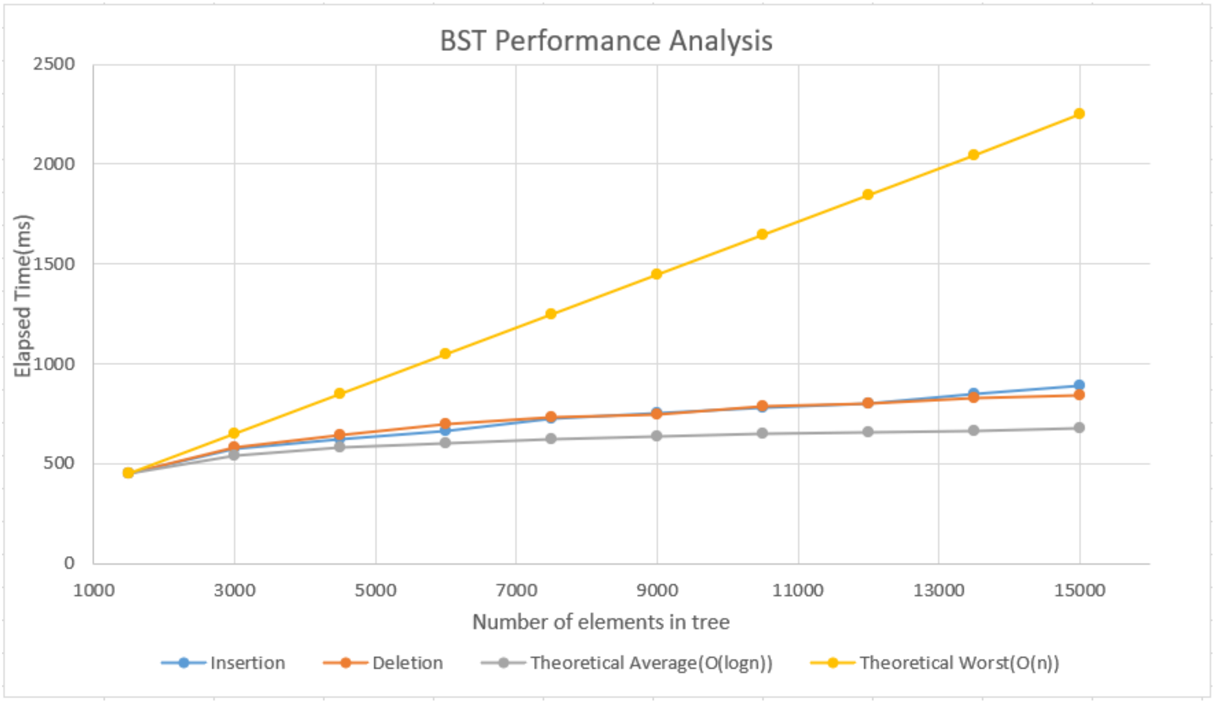
$$\begin{tikzcd}
 & & \boxed{14} & & \\
\boxed{4} & & & \boxed{15} & \\
 & \boxed{6} & & & \boxed{16}
\arrow[from=1-3, to=2-1]
\arrow[from=1-3, to=2-4]
\arrow[from=2-1, to=3-2]
\arrow[from=2-4, to=3-5]
\end{tikzcd}$$

d) findMin(node):

```
    if(node->left == nullptr)
                return node;
    else:
                findMin(node->left);
```

e) Maximum height: n Minimum height: $[\log_2 n]$

Question 3)



The empirical test results for insertion and deletion are more similar to O(log(n)) in terms of growth than O(n). This can be explained because of the randomization of the array before insertion and deletion. When we insert sorted elements in random order the height of the tree becomes similar to $\log_2 n$, so the traversal time for deletion and insertion is also similar to this order. If there was a perfectly balanced tree the growth of the insertion and deletion functions would be the same as the theoretical average however this isn't the case so the empirical results are somewhere in the middle of the worst and average cases. Also, the reason why the empirical results differ from the worst case is because the worst case occurs when we have a sorted array that is inserted and deleted. The traversal times grow in the order of O(n).

As stated before, if we tried to insert a sorted array, the results would look similar to the worst-case graph as the method would need to traverse the whole tree to find the right place to insert. However, the deletion would be in constant time. This is because the element that is to be deleted would be at the root, so there would not be any traversals happening. But if we tried to delete in the opposite order of insertion then the results for deletion and insertion would both be the worst case. The randomness of the array makes the empirical insertion and deletion act similar to the average case.