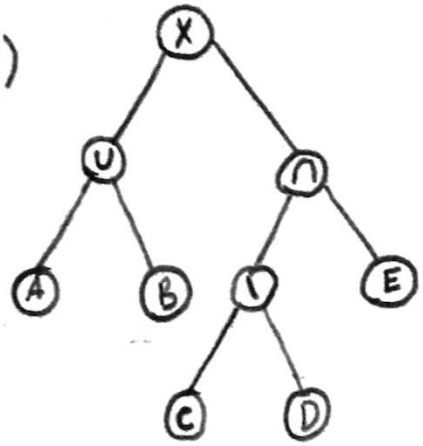
Berk Gakar — 22003021 — CS202-3

# Homework 2 - Trees

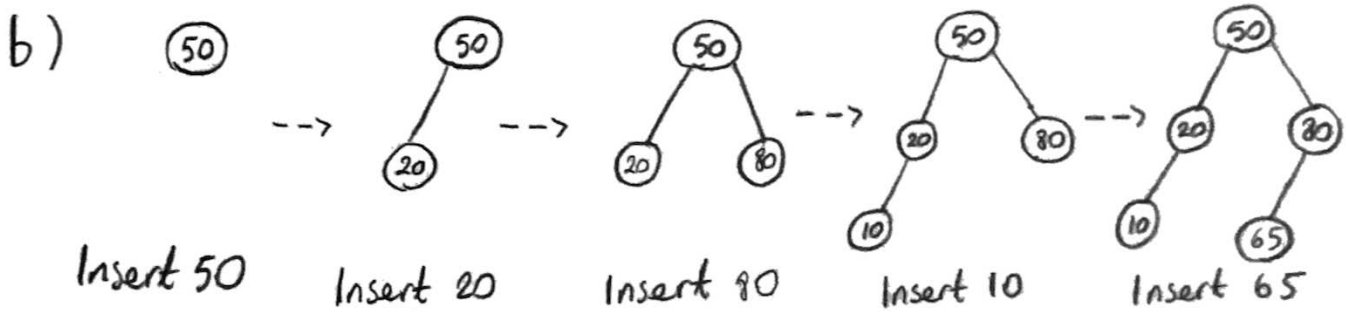
## Question 1

a)

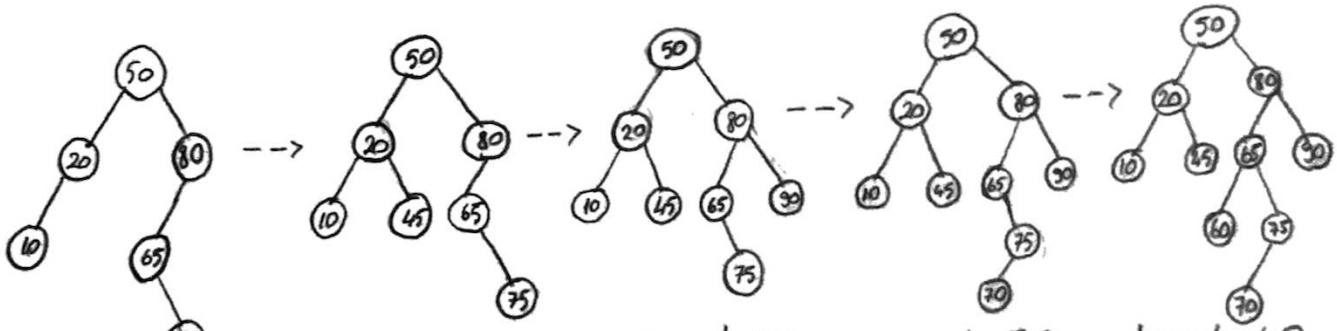



Prefix expression for the given tree:
X U A B ∩ \ C D E
Infix expression for the given tree:
((A U B) X ((C \ D) ∩ E))
Postfix expression for the given tree:
A B U C D \ E ∩ X

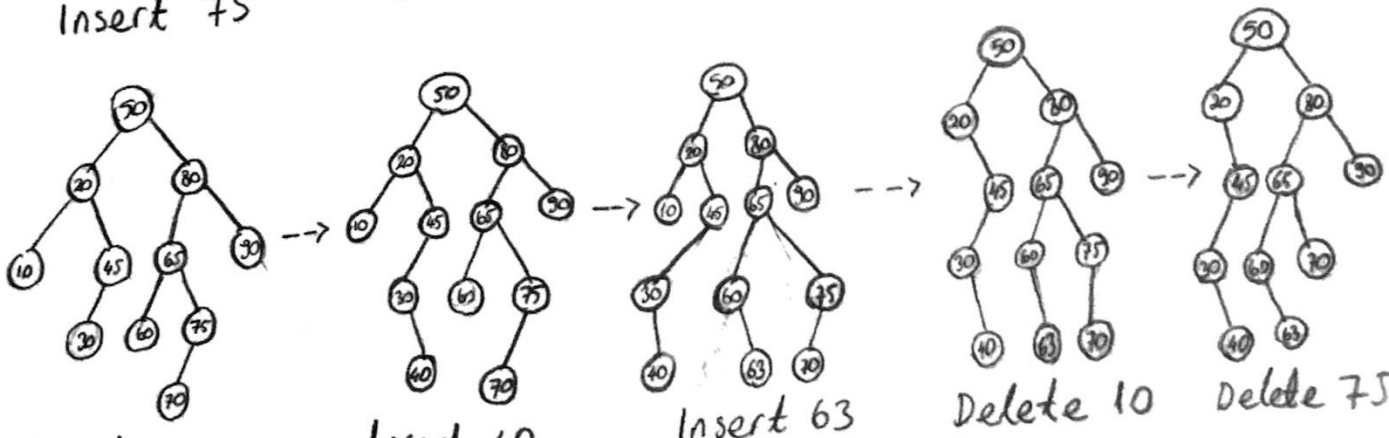
b)




Insert 50 — Insert 20 — Insert 80 — Insert 10 — Insert 65

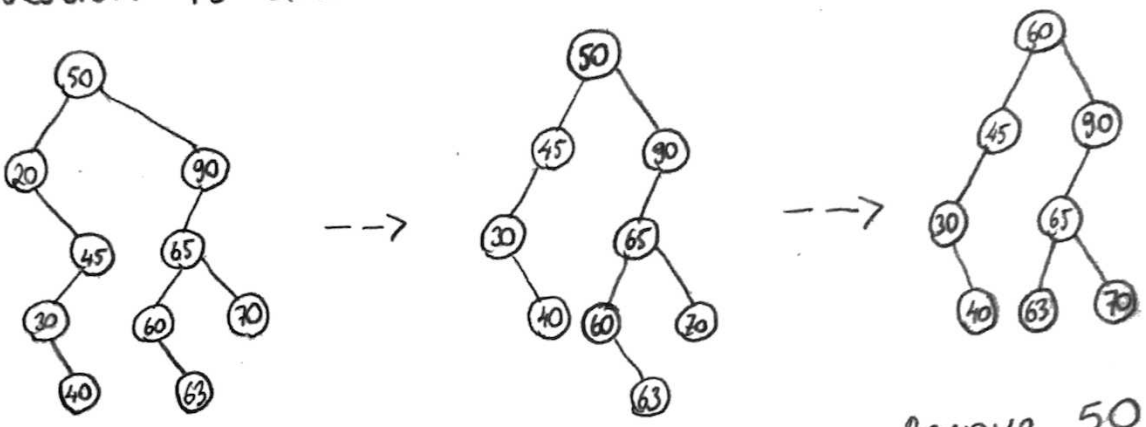



Insert 75 — Insert 45 — Insert 90 — Insert 70 — Insert 60



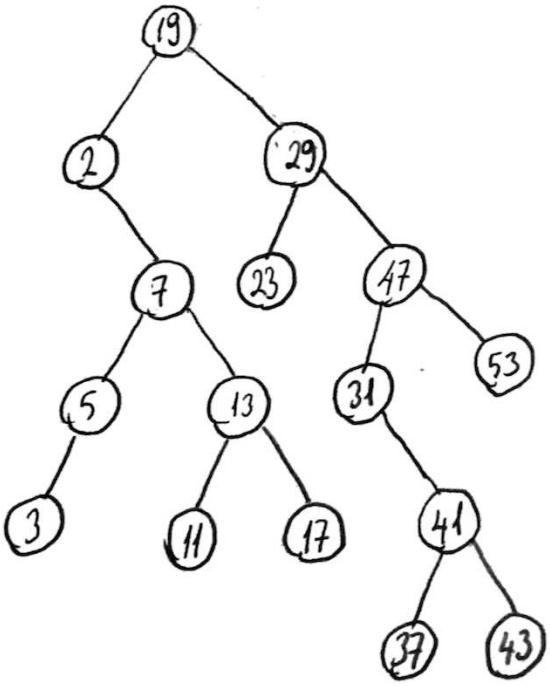

Insert 30 — Insert 40 — Insert 63 — Delete 10 — Delete 75

Question 1b continues from here:




Remove 80 Remove 20 Remove 50

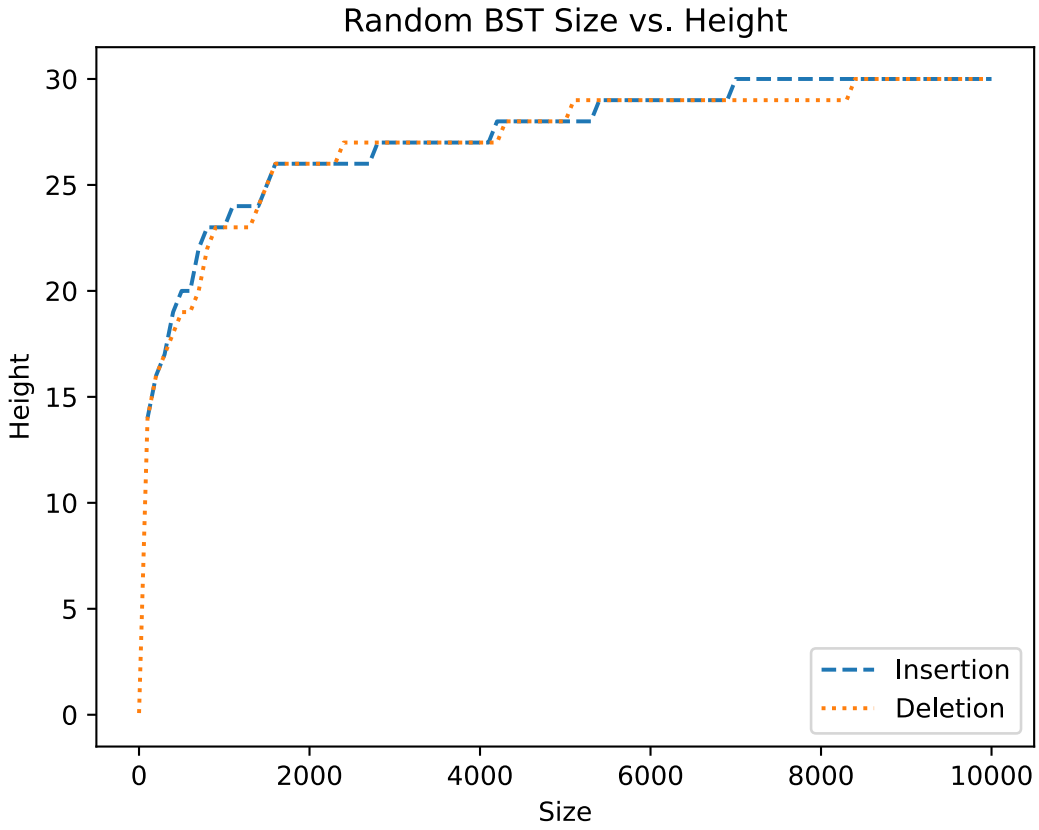
c)




Postorder Traversal:
3, 5, 11, 17, 13, 7, 2, 23, 37, 43, 41, 31, 53, 47, 29, 19

Preorder Traversal:
19, 2, 7, 5, 3, 13, 11, 17, 29, 23, 47, 31, 41, 37, 43, 53

**Question 3**




As stated in the lecture slides, the average height of an n-node binary search tree is close to $\lceil \log_2(n+1) \rceil$. On the other hand, for the worst case, height is $n$. Therefore, expected average heights and the obtained heights for the ranges 0-2000, 2000-4000, 4000-6000, 6000-8000, 8000-10000 are the following:

| *Number of nodes* | *Expected average height* | *Obtained height* |
|---|---|---|
| $(0, 2000]$ | $0 < h \leq 11$ | $0 < h \leq 26$ |
| $(2000, 4000]$ | $11 < h \leq 12$ | $26 < h \leq 27$ |
| $(4000, 6000]$ | $12 < h \leq 13$ | $27 < h \leq 29$ |
| $(6000, 8000]$ | $13 \leq h \leq 13$ | $29 < h \leq 30$ |
| $(8000, 10000]$ | $13 < h \leq 14$ | $29 < h \leq 30$ |

Due to the randomness of the numbers inserted, the plotted binary search tree has approximately two and a half times the expected height for 0-2000 nodes, which can be called a starting range. Apart from that, at equally increasing node number intervals, the height increased by 1 or 2 as expected. By considering this fact and the logarithmic shape of the lines in the graph, it is possible to say that the data obtained meets the expectations.

Nevertheless, slight differences were seen between the insertion and deletion lines in the graph. This is because the array holding the numbers to be inserted in BST is shuffled before deletion operation. In other words, the numbers were not deleted from BST in the order they were inserted. If the data were deleted in the order in which they were inserted, the two lines would be expected to be the same.