

# TREE

## *Intermediate Level Questions:*

### Binary Tree:

- Print top view, bottom view , left view and right view of a binary tree.

[Follow here: <https://www.geeksforgeeks.org/print-nodes-top-view-binary-tree/> ]

[Practice here: <https://practice.geeksforgeeks.org/problems/top-view-of-binary-tree/1> ]

[Practice here: <https://practice.geeksforgeeks.org/problems/bottom-view-of-binary-tree/1> ]

[Practice here: <https://practice.geeksforgeeks.org/problems/left-view-of-binary-tree/1> ]

[Practice here: <https://practice.geeksforgeeks.org/problems/right-view-of-binary-tree/1> ]

- Find N<sup>th</sup> node of Inorder Traversal

[Follow here: <https://www.geeksforgeeks.org/find-n-th-node-inorder-traversal/> ]

- Print Level Order Traversal in Spiral Form

[Practice here: <https://practice.geeksforgeeks.org/problems/level-order-traversal-in-spiral-form/1> ]

- Print Diagonal Traversal of a Binary Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/diagonal-traversal-of-binary-tree/1> ]

- Print Boundary Traversal of Binary Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/boundary-traversal-of-binary-tree/1> ]

- Construct a Binary Tree from given Inorder and Preorder traversal

[Practice here: <https://practice.geeksforgeeks.org/problems/construct-tree-1/1> ]

- Construct a Binary Tree from Inorder and Level order traversal

[Practice here: <https://practice.geeksforgeeks.org/problems/construct-tree-from-inorder-and-levelorder/1> ]

- Construct Binary Tree from String with Bracket Representation

[Follow here: <https://www.geeksforgeeks.org/construct-binary-tree-string-bracket-representation/> ]

- Convert a Binary Tree into Doubly Linked List(DLL)

[Practice here: <https://practice.geeksforgeeks.org/problems/binary-tree-to-dll/1> ]

- Convert a Given Binary Tree into a Sum Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/transform-to-sum-tree/1> ]

- Find minimum swaps required to convert a Binary tree into Binary Search Tree

[Follow here: <https://www.geeksforgeeks.org/minimum-swap-required-convert-binary-tree-binary-search-tree/> ]

- Check if Binary Tree is Sum tree or not

[Practice here: <https://practice.geeksforgeeks.org/problems/sum-tree/1> ]

- Check if All leaf node are at same level or not

[Practice here: <https://practice.geeksforgeeks.org/problems/leaf-at-same-level/1> ]

- Check if a Binary Tree contains duplicate subtrees of size 2 or more.

[Practice here: <https://practice.geeksforgeeks.org/problems/duplicate-subtree-in-binary-tree/1> ]

- Check if two trees are mirror

[Practice here: <https://practice.geeksforgeeks.org/problems/check-mirror-in-n-ary-tree/0> ]

- Check if given graph is tree or not

[Follow here: <https://www.geeksforgeeks.org/check-given-graph-tree/> ]

- Sum of Nodes on the longest path from root to leaf node

[Practice here: <https://practice.geeksforgeeks.org/problems/sum-of-the-longest-bloodline-of-a-tree/1> ]

- Find Largest subtree sum in a tree

[Follow here: <https://www.geeksforgeeks.org/find-largest-subtree-sum-tree/> ]

- Maximum sum of nodes in Binary Tree such that no two are adjacent

[Practice here: <https://www.geeksforgeeks.org/maximum-sum-nodes-binary-tree-no-two-adjacent/> ]

- Print all k-sum paths in a Binary Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/k-sum-paths/1> ]

- Find Lowest Common Ancestor in a Binary Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/lowest-common-ancestor-in-a-binary-tree/1> ]

- Find distance between two nodes in a Binary Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/min-distance-between-two-given-nodes-of-a-binary-tree/1> ]

- K<sup>th</sup> Ancestor of a node in a Binary tree

[Follow here: <https://www.geeksforgeeks.org/kth-ancestor-node-binary-tree-set-2/> ]

- Find All Duplicate Subtrees in a Binary Tree

[Practice here: <https://practice.geeksforgeeks.org/problems/duplicate-subtrees/1>]

- Tree Isomorphism Problem

[Practice here: <https://practice.geeksforgeeks.org/problems/check-if-tree-is-isomorphic/1> ]

### Binary Search Tree:

- Construct BST from inorder and preorder traversal
- Construct BST from inorder and postorder traversal

- Construct BST from Preorder Traversal

[Follow here: <https://www.geeksforgeeks.org/construct-bst-from-given-preorder-traversal/> ]

- Convert Binary Tree into BST

[Practice here: <https://practice.geeksforgeeks.org/problems/binary-tree-to-bst/1>]

- Convert a normal BST into balanced BST

[Follow here: <https://www.geeksforgeeks.org/convert-normal-bst-balanced-bst/> ]

- Merge two BST [Very Important]

[Practice here: <https://practice.geeksforgeeks.org/problems/merge-two-bst-s/1> ]

- Find Lowest Common Ancestor (LCA) of BST

[Practice here: <https://practice.geeksforgeeks.org/problems/lowest-common-ancestor-in-a-bst/1> ]

- Find K<sup>th</sup> Largest Element in a BST

[Practice here: <https://practice.geeksforgeeks.org/problems/kth-largest-element-in-bst/1>]

- Count pairs from Two BSTs whose sum is equal to given value x.

[Practice here: <https://practice.geeksforgeeks.org/problems/brothers-from-different-root/1>]

- Find the median of BST in  $O(n)$  time and  $O(1)$  space

[Follow here: <https://www.geeksforgeeks.org/find-median-bst-time-o1-space/>]

- Count BST nodes that lies in the given range

[Practice here: <https://practice.geeksforgeeks.org/problems/count-bst-nodes-that-lie-in-a-given-range/1> ]

- Replace every element with the least greater element on its right

[Practice here: <https://www.geeksforgeeks.org/replace-every-element-with-the-least-greater-element-on-its-right/> ]

- Given “n” appointments, find the conflicting appointments

[Practice here: <https://www.geeksforgeeks.org/given-n-appointments-find-conflicting-appointments/> ]

- Populate inorder successor of all nodes.

[Practice here: <https://practice.geeksforgeeks.org/problems/populate-inorder-successor-for-all-nodes/1> ]

- Check Dead in a BST

[Practice here: <https://practice.geeksforgeeks.org/problems/check-whether-bst-contains-dead-end/1>]

- Check preorder is valid or not

[Practice here: <https://practice.geeksforgeeks.org/problems/preorder-to-postorder/0> ]

## Expression tree:

- Evaluate Expression tree.

[Practice here: <https://practice.geeksforgeeks.org/problems/expression-tree/1> ]

## AVL Tree:

- Insertion and Deletion only

Follow here:

[ Insertion: <https://www.geeksforgeeks.org/avl-tree-set-1-insertion/> ]

[ Deletion: <https://www.geeksforgeeks.org/avl-tree-set-2-deletion/> ]

## **RBL Tree:**

- Insertion and Deletion only

Follow here:

[ Intro: <https://www.geeksforgeeks.org/red-black-tree-set-1-introduction-2/> ]

[ Insertion: <https://www.geeksforgeeks.org/red-black-tree-set-2-insert/> ]

[ Deletion: <https://www.geeksforgeeks.org/red-black-tree-set-3-delete-2/> ]

## **B Tree and B<sup>+</sup> Tree:**

- Go through theory only

[ B tree: <https://www.geeksforgeeks.org/introduction-of-b-tree-2/> ]

[ B<sup>+</sup> Tree: <https://www.geeksforgeeks.org/introduction-of-b-tree/> ]