
TREES

INDEX

- 1. Write a C program to find the depth or height of a tree.**
- 2. Write a C program to determine the number of elements (or size) in a tree.**
- 3. Write a C program to delete a tree (i.e, free up its nodes)**
- 4. Write C code to determine if two trees are identical**
- 5. Write a C program to find the minimum value in a binary search tree**
- 6. Write a C program to create a mirror copy of a tree (left nodes become right and right nodes become left)!**
- 7. Write C code to return a pointer to the nth node of an inorder traversal of a BST.**
- 8. Write C code to implement the preorder(), inorder() and postorder() traversals. What's their time complexities?**
- 9. Write a C program to create a copy of a tree**
- 10. Write C code to check if a given binary tree is a binary search tree**
- 11. Write a C program to delete a node from a Binary Search Tree?**
- 12. Write C code to search for a value in a binary search tree (BST).**
- 13. Write C code to count the number of leaves in a tree**
- 14. Construct a tree given its inorder and preorder traversal strings. Similarly construct a tree given its inorder and post order traversal strings.**

- 15. Given an expression tree, evaluate the expression and obtain a parenthesized form of the expression.**
- 16. How do you convert a tree into an array?**
- 17. Tree Traversals**
- 18. Lowest Common Ancestor in a Binary Search Tree.**
- 19. Level Order Tree Traversal**
- 20. Program to count leaf nodes in a binary tree**
- 21. Level order traversal in spiral form**
- 22. Check for Children Sum Property in a Binary Tree.**
- 23. Convert an arbitrary Binary Tree to a tree that holds Children Sum Property**
- 24. Diameter of a Binary Tree**
- 25. How to determine if a binary tree is height-balanced?**
- 26. Inorder Tree Traversal without Recursion**
- 27. Inorder Tree Traversal without recursion and without stack!**
- 28. Construct Tree from given Inorder and Preorder traversals**
- 29. Maximum width of a binary tree**
- 30. Foldable Binary Trees**
- 31. Print nodes at k distance from root**
- 32. Sorted order printing of a given array that represents a BST**
- 33. Applications of tree data structure**

- 34.Inorder Successor in Binary Search Tree**
- 35.Get Level of a node in a Binary Tree**
- 36.Print Ancestors of a given node in Binary Tree**
- 37.Print BST keys in the given range**
- 38.Tournament Tree (Winner Tree) and Binary Heap**
- 39.Check if a given Binary Tree is SumTree**
- 40.Decision Trees – Fake (Counterfeit) Coin Puzzle**
- 41.Check if a binary tree is subtree of another binary tree**
- 42.Trie | (Insert and Search)**
- 43.Trie | (Delete)**
- 44.Connect nodes at same level**
- 45.Connect nodes at same level using constant extra space**
- 46.Sorted Linked List to Balanced BST**