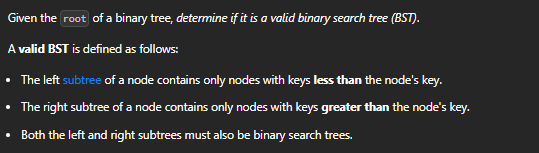
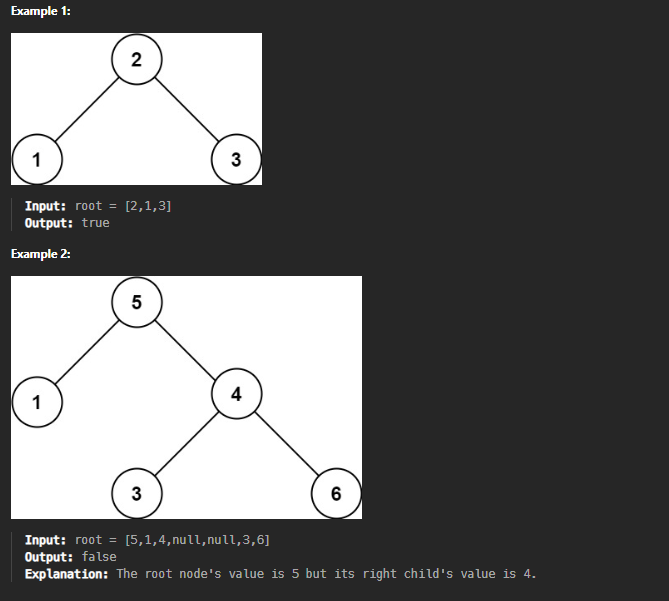
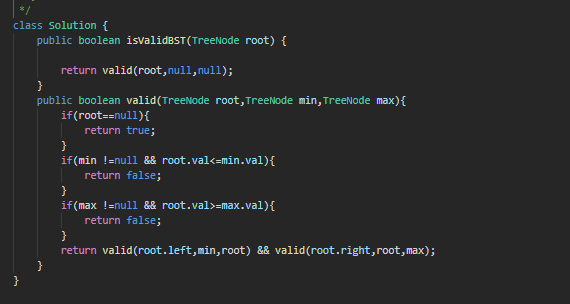
**Binary Search Tree**

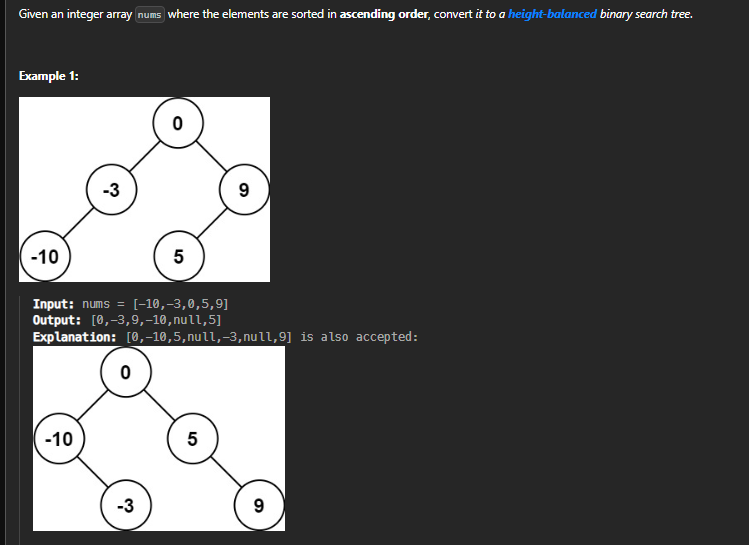
1.Validate Binary Search Tree

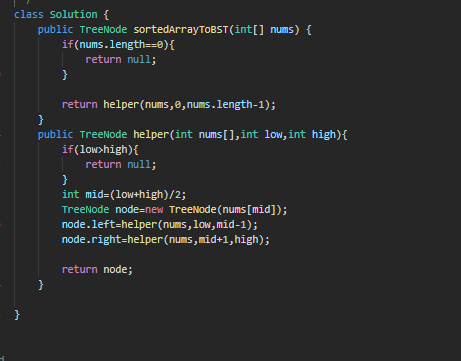




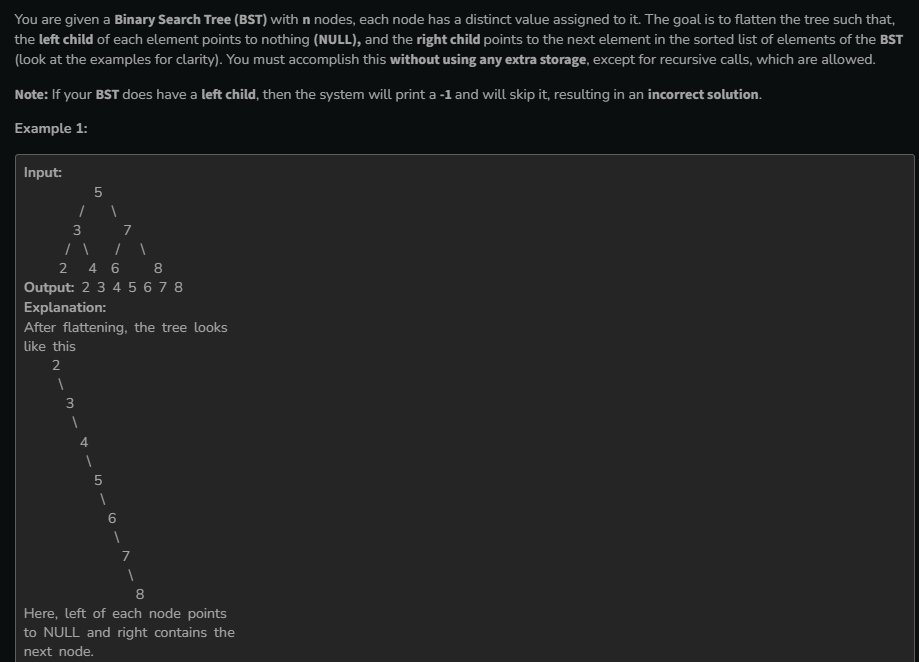


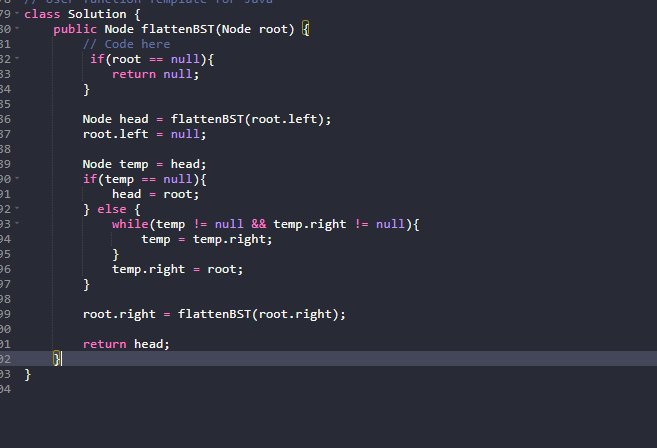
2.Converted sorted array to BST



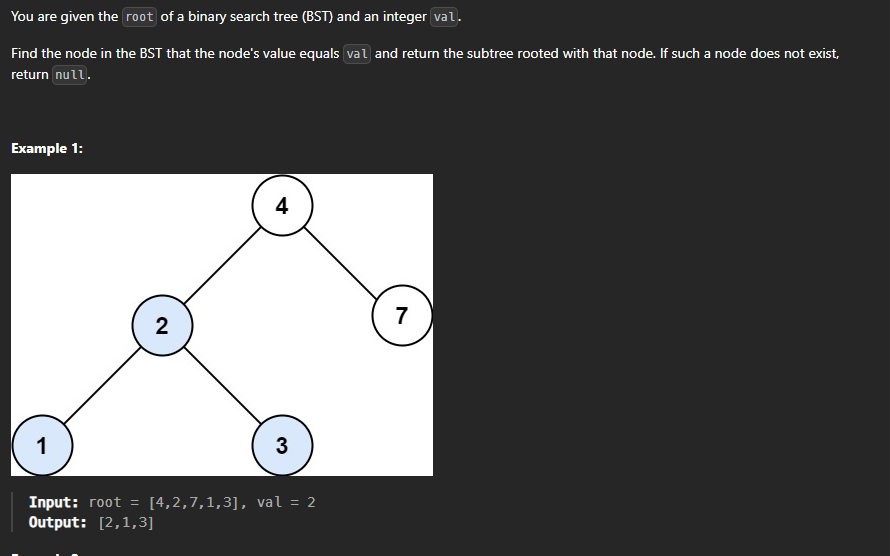


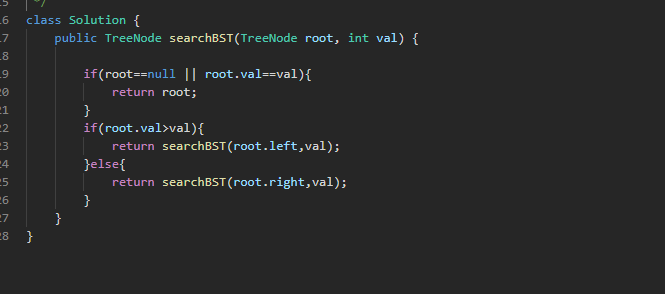
3.Flatten BST to sorted List



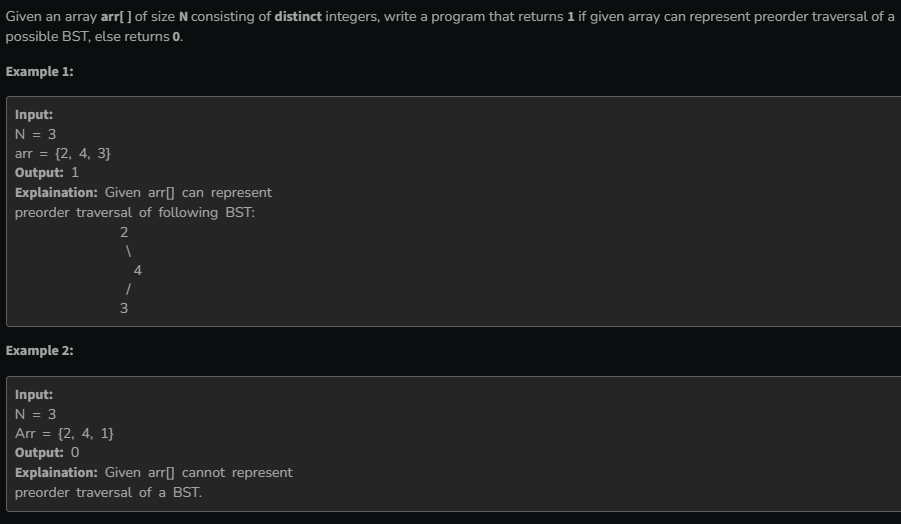


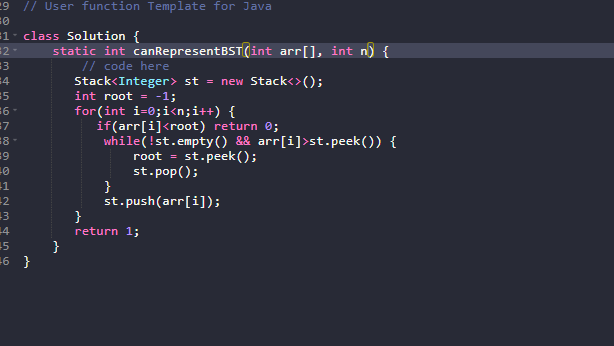
4.Search in Binary Search Tree





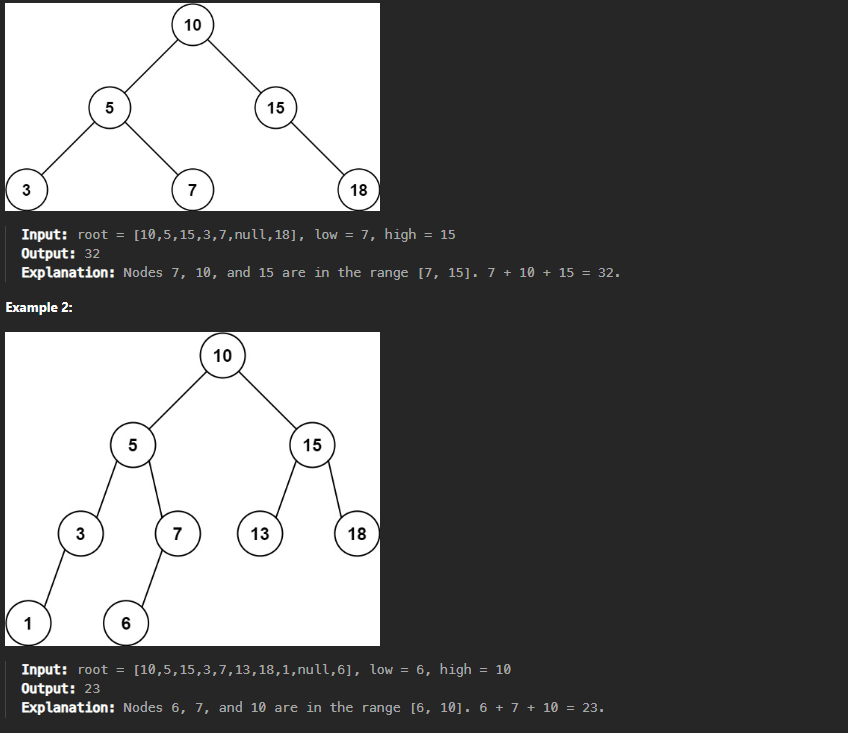
5.Preorder Traversal and Bst

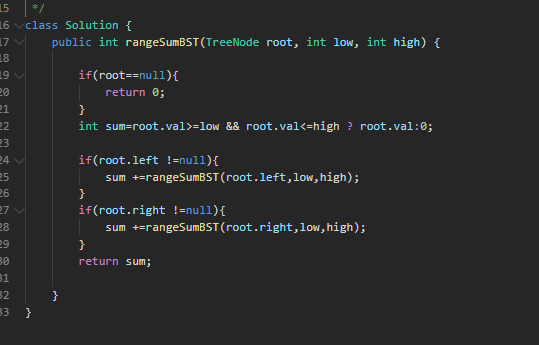




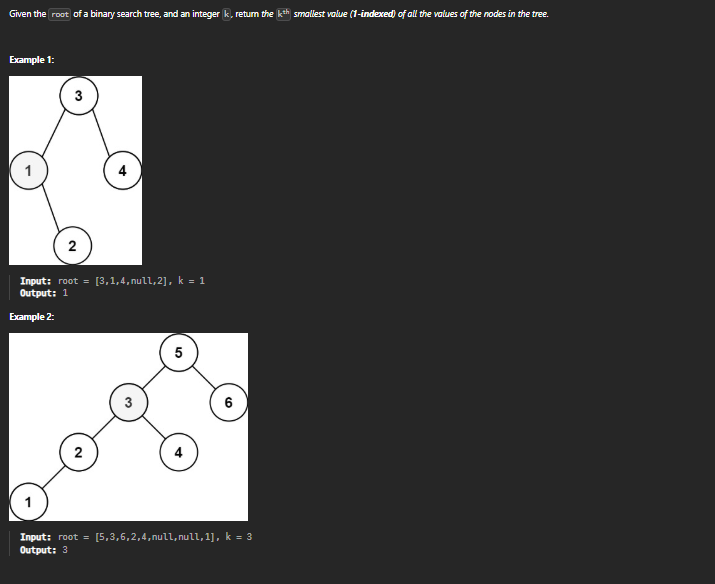
6.Range Sum of BST

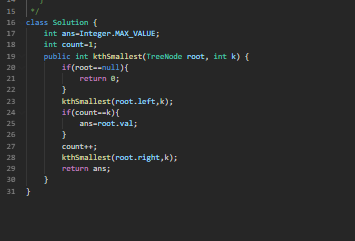
Given the root node of a binary search tree and two integers low and high, return the sum of values of all nodes with a value in the inclusive range [low, high].



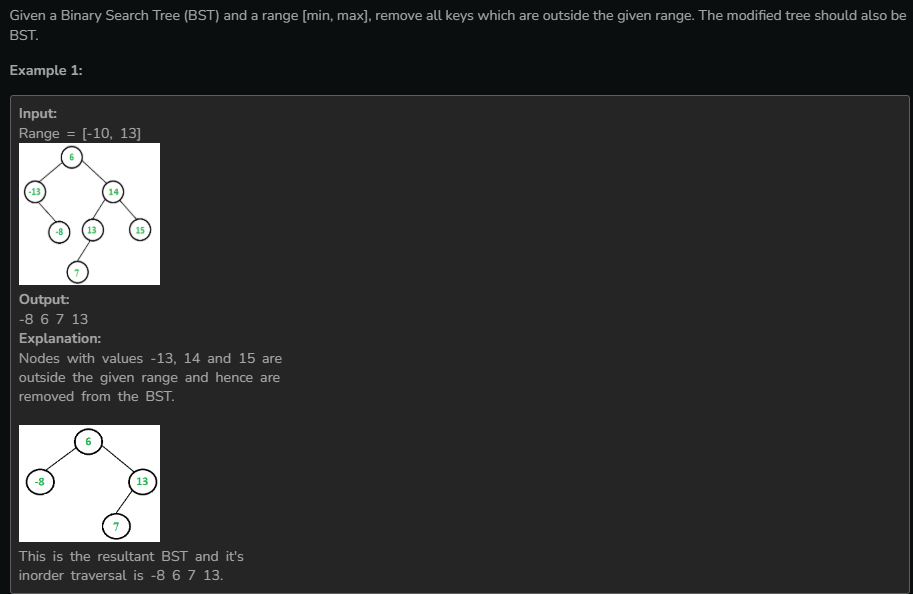


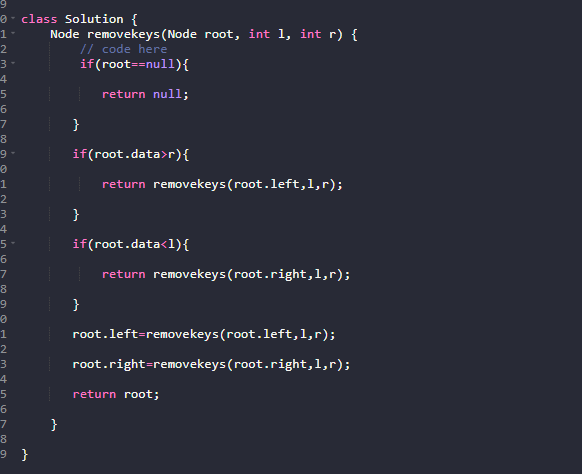
7.Kth Smallest Element in BST



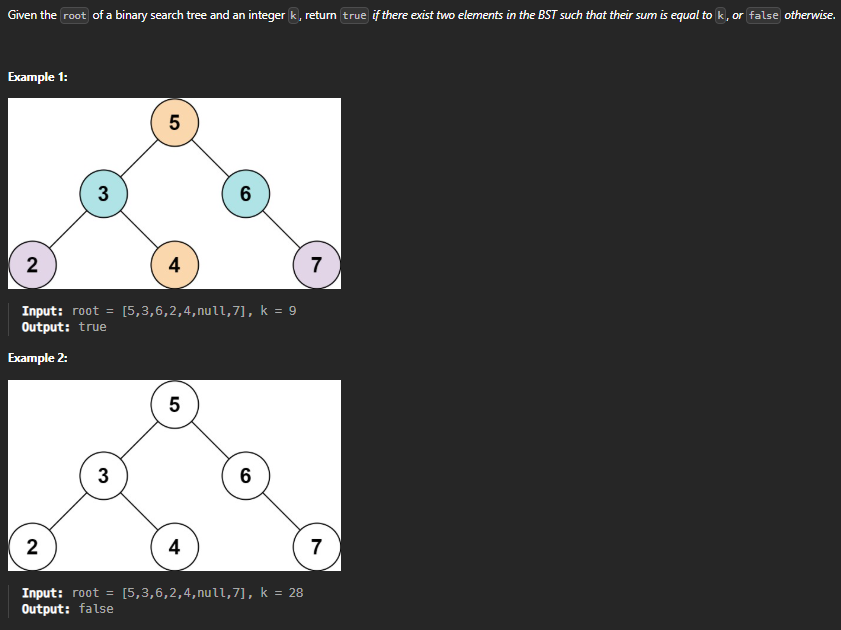


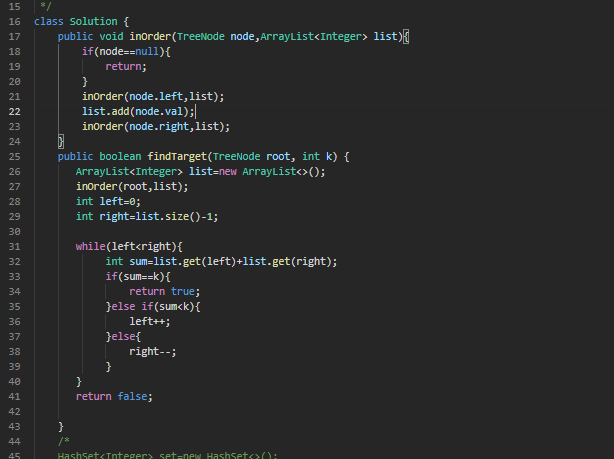
8.Remove BST key outside given range



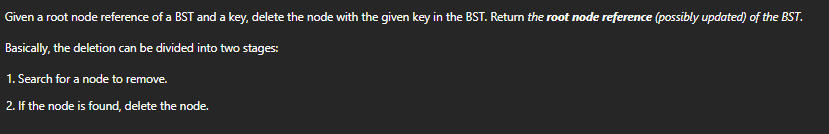


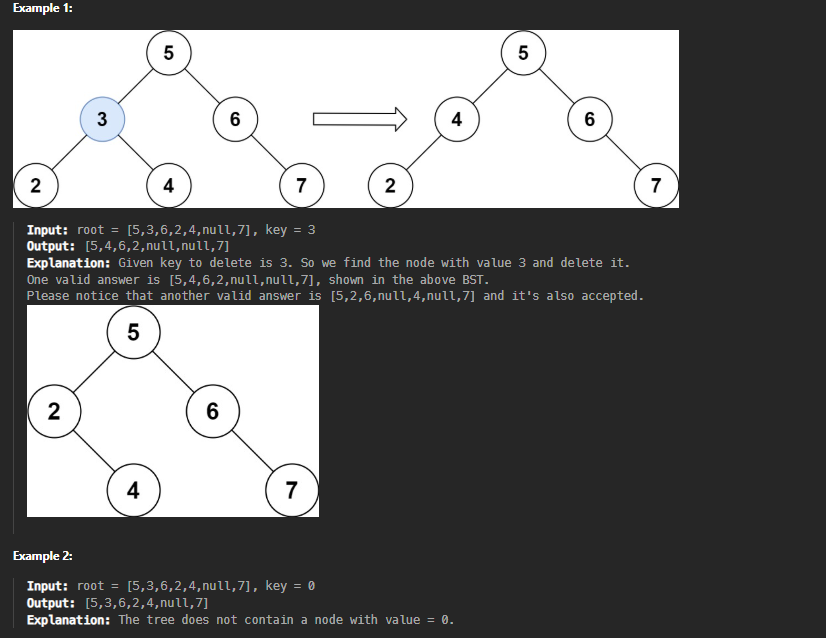
9.Two Sum IV – Input is a BST





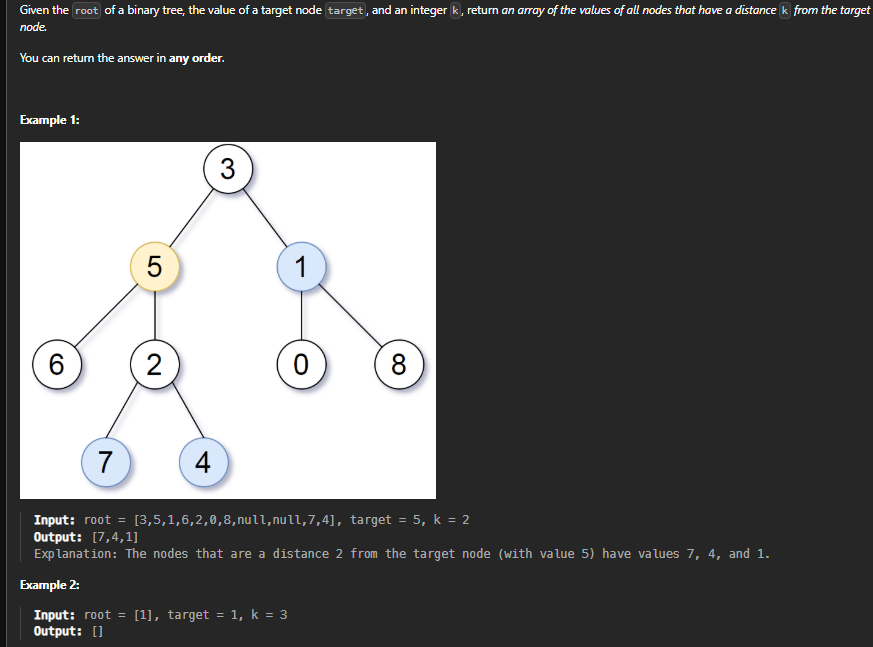
10.Delete node in a BST

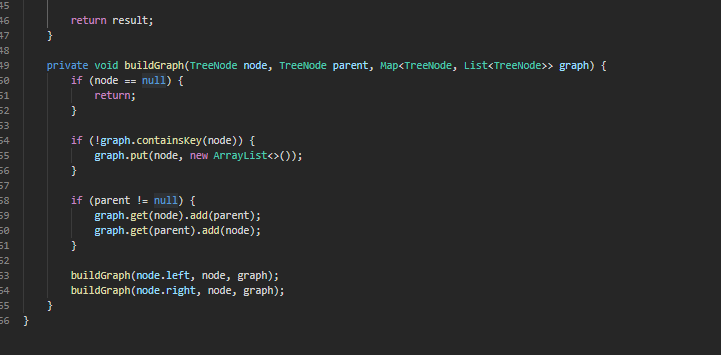
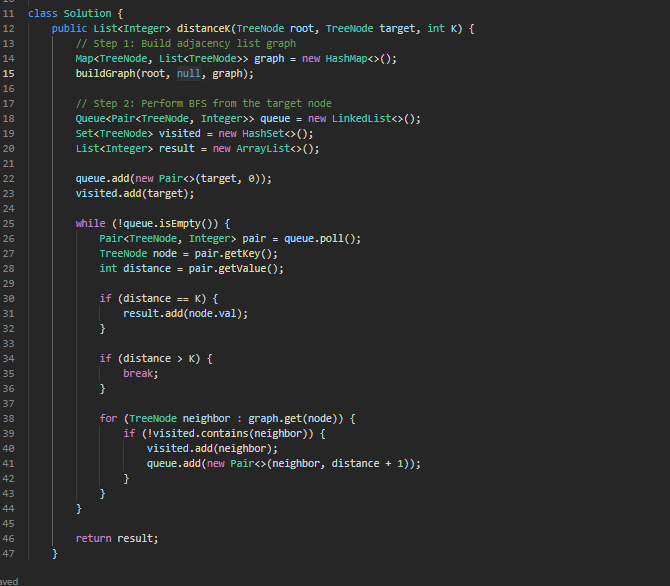




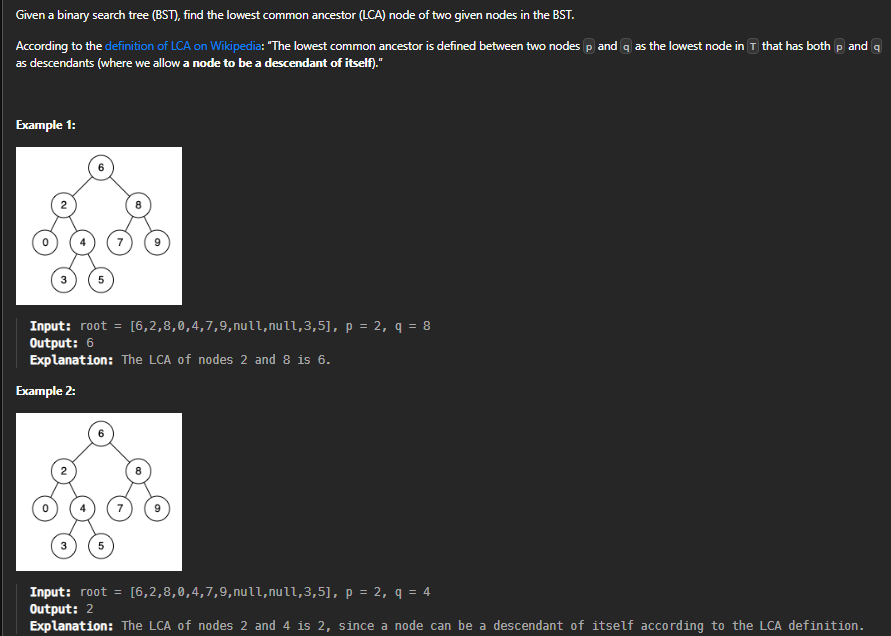


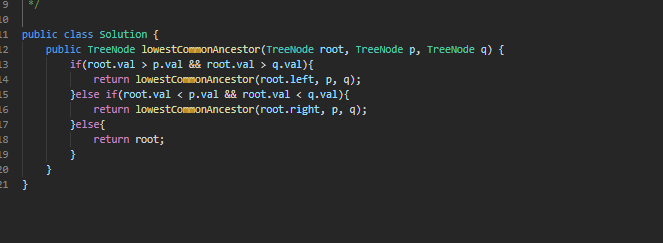
11.All Node Distance K in Binary Tree



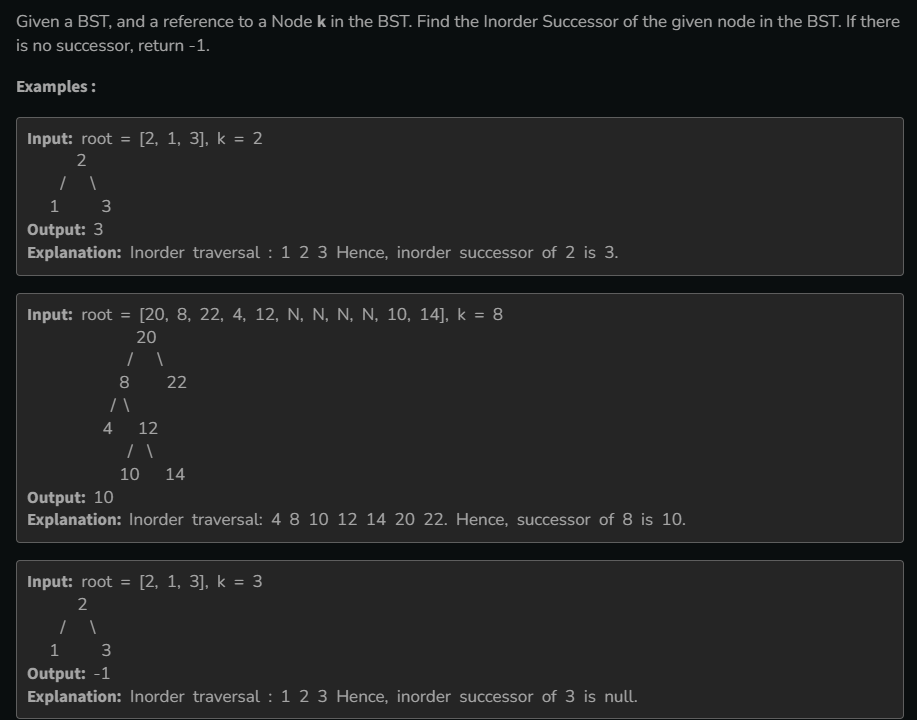


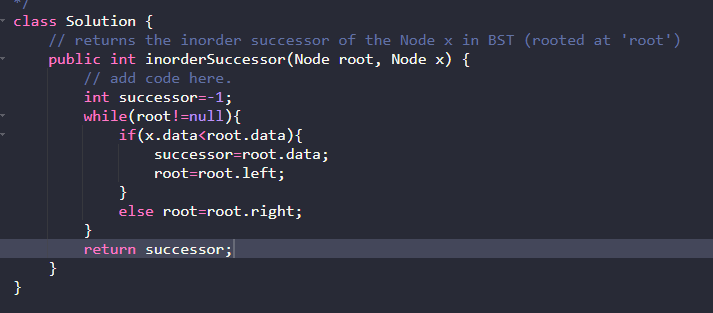
12.Lowest Common Ansector of a Binary Search Tree





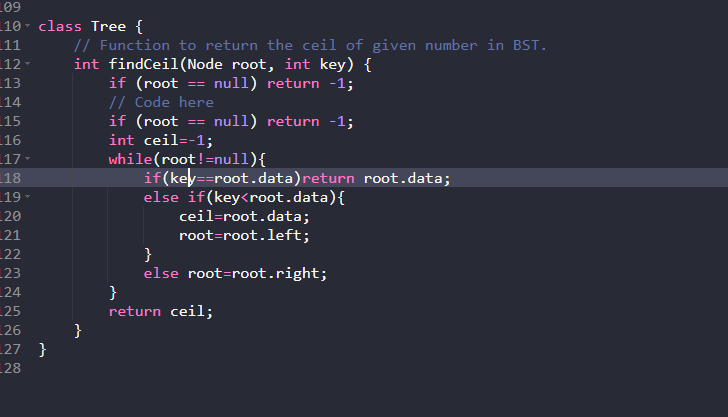
13.Inorder Succesor in BST



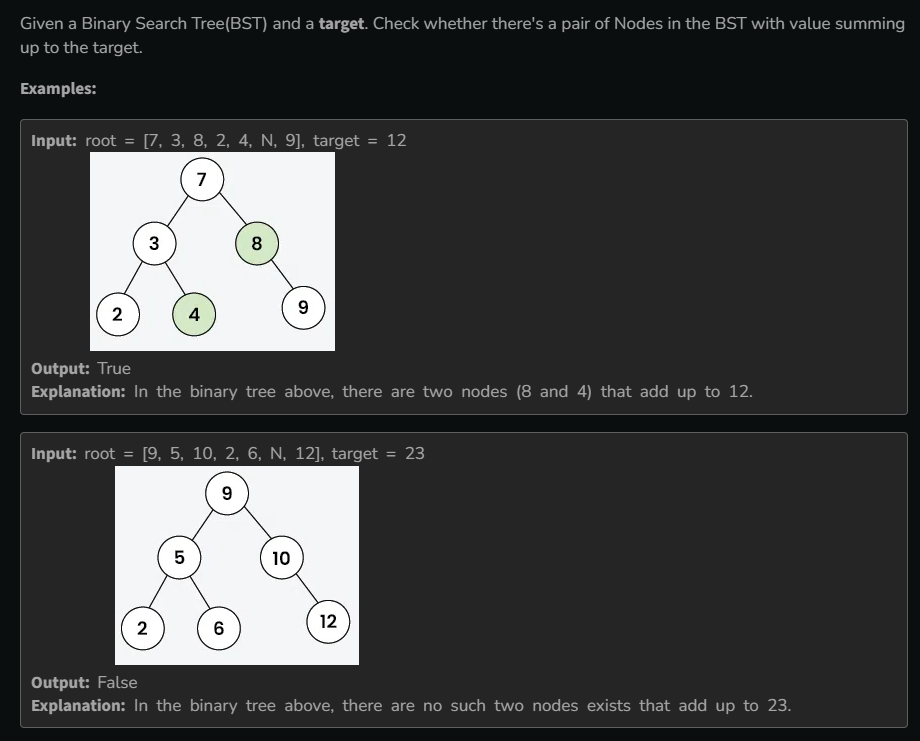


14.Ceil in BST



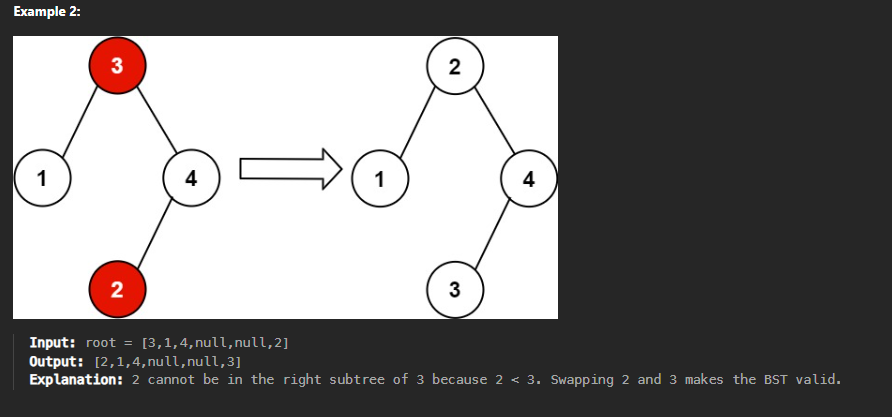
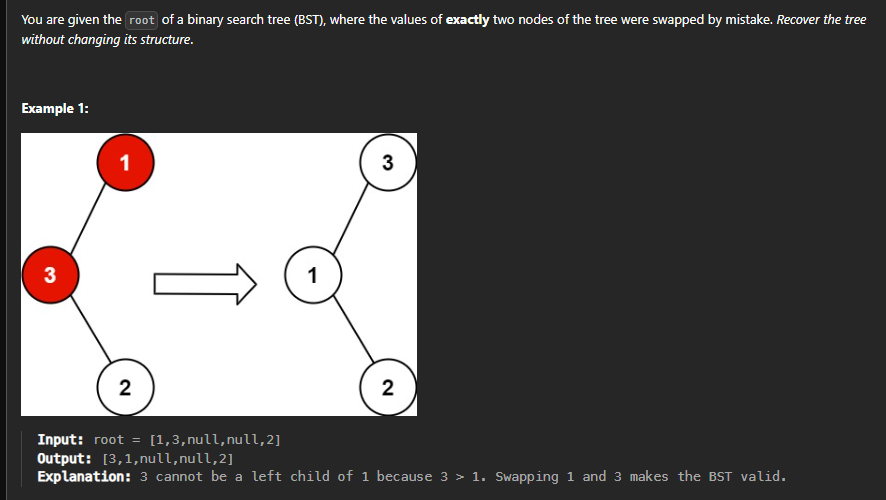


15.Pair Sum in BST





16.Recover Binary Search Tree





<https://leetcode.com/problems/serialize-and-deserialize-bst/https://leetcode.com/problems/serialize-and-deserialize-bst/>