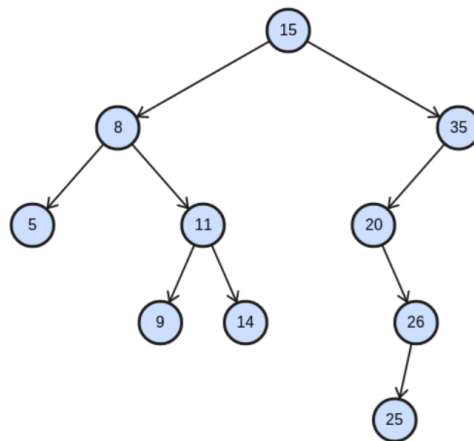


# DSA

February 2018

## 1 Binary Search Tree



The following questions are available on various websites. Please search them on Google and submit the code to be sure about your logic and implementation.

- Insert a node with given data in BST. The node should be inserted such that BST structure properties are maintained.
- Search a node with given data in the BST, return true if present else return false.
- Delete a node with given data such that the BST structure properties are maintained.
- Find the Lowest Common Ancestor of a group of nodes. (LCA of 5, 11, 14 is 8)
- Find Inorder Successor of a node in BST. (The Inorder Successor of the node 15 is 20)

- Find Inorder Predecessor of a node in BST. (The Inorder Predecessor of the node 9 is 8)
- Count BST subtrees that lie in given range. For example- let's take the range as  $[8,15]$ , then the subtree rooted at leaf 9, 14 are in the range, and the subtree rooted at node 11 is also in the range, because the minimum value and the maximum value in that subtree (9 and 11 respectively) are within the mentioned range. The subtree rooted at 8 is not in the range, because the minimum value 5 in this subtree is out of the given range. Hence, answer=3 (3 subtrees possible)
- Two of the nodes of a Binary Search Tree (BST) are swapped. Fix (or correct) the BST
- Given a sorted array. Write a function that creates a Balanced Binary Search Tree using array elements.
- Find the total number of possible Binary Search Trees with  $n$  distinct keys.
- Construct Tree from given Inorder and Preorder traversals.