# SNS COLLEGE OF TECHNOLOGY

**Coimbatore-35**
**An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

## DEPARTMENT OF ARTIFICIAL INTELLIGENCE

## 19ITT101-PROGRAMMING IN C AND DATA STRUCTURES

I YEAR - II SEM

UNIT 5 – Trees

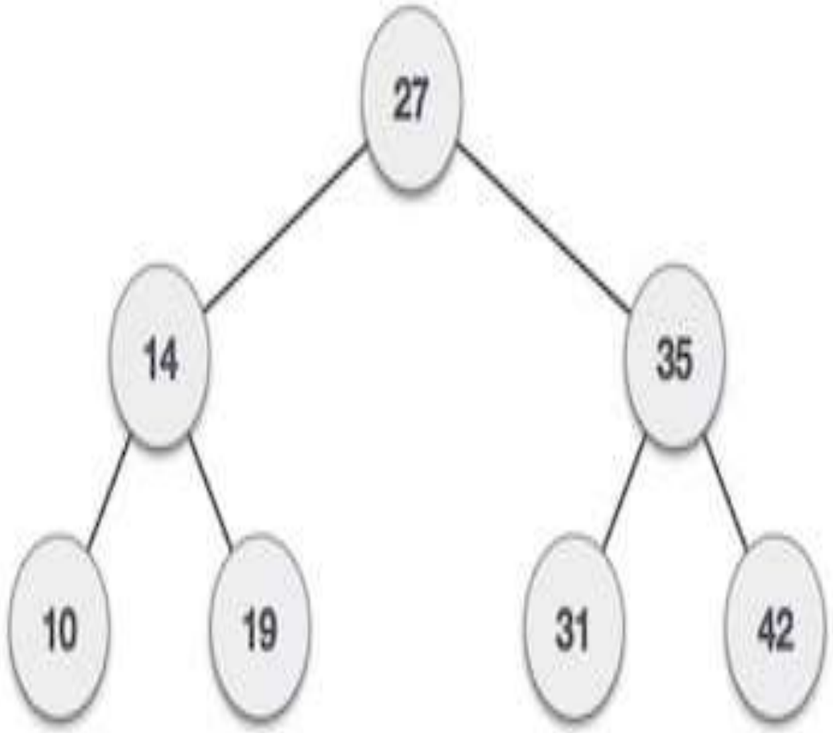
TOPIC 2 – Binary Search Tree

# Binary Search Trees

- A Binary Search Tree (BST) is a tree in which all the nodes follow the below-mentioned properties −
  - The value of the key of the left sub-tree is less than the value of its parent (root) node's key.
  - The value of the key of the right sub-tree is greater than or equal to the value of its parent (root) node's key

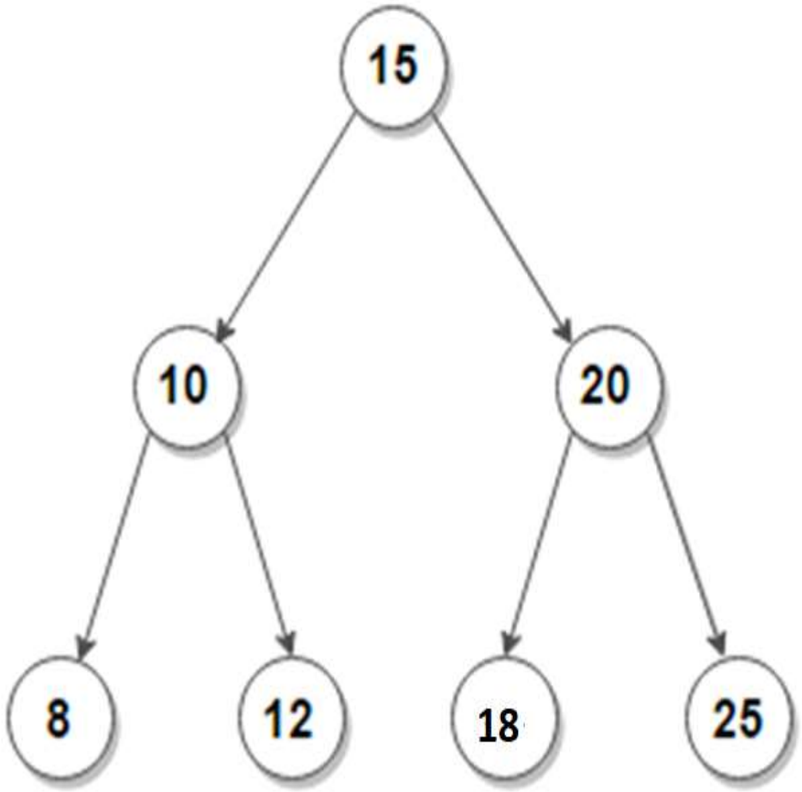
$$left\_subtree\ (keys) < node\ (key) \leq right\_subtree\ (keys)$$

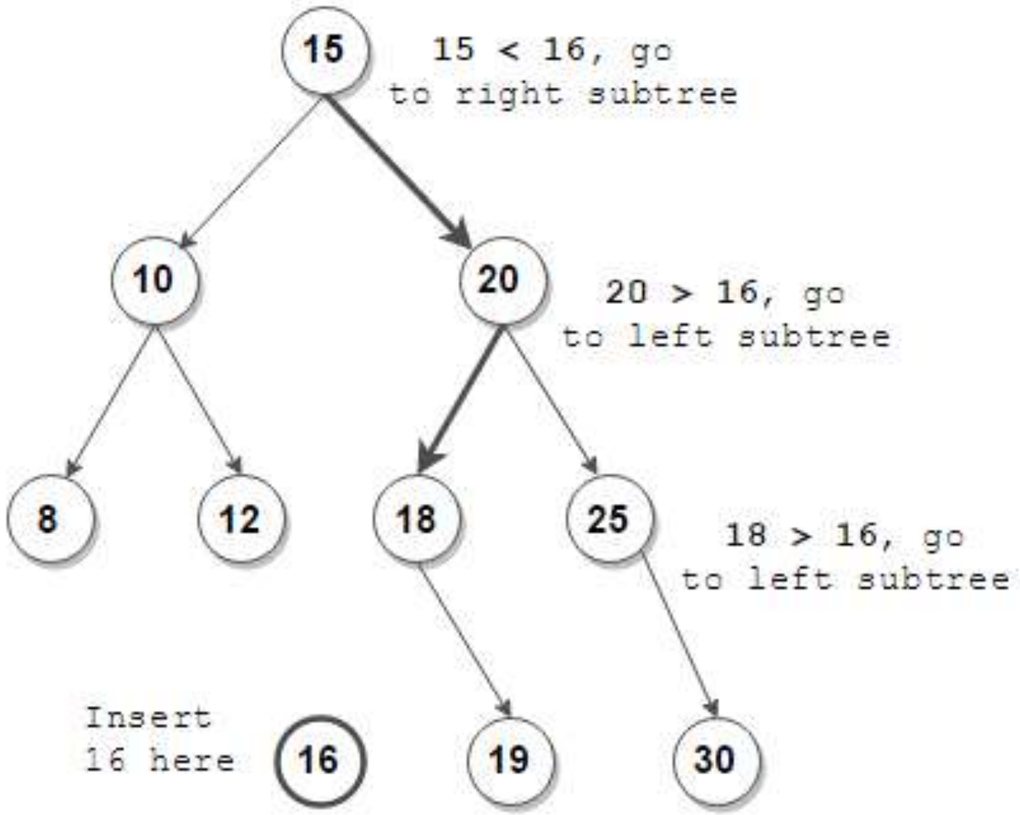
# Operations of Binary Search Tree




- **Search** − Searches an element in a tree.
- **Insert** − Inserts an element in a tree.
- **Delete** −Delete an element in a tree.
- **Pre-order Traversal** − Traverses a tree in a pre-order manner.
- **In-order Traversal** − Traverses a tree in an in-order manner.
- **Post-order Traversal** − Traverses a tree in a post-order manner.

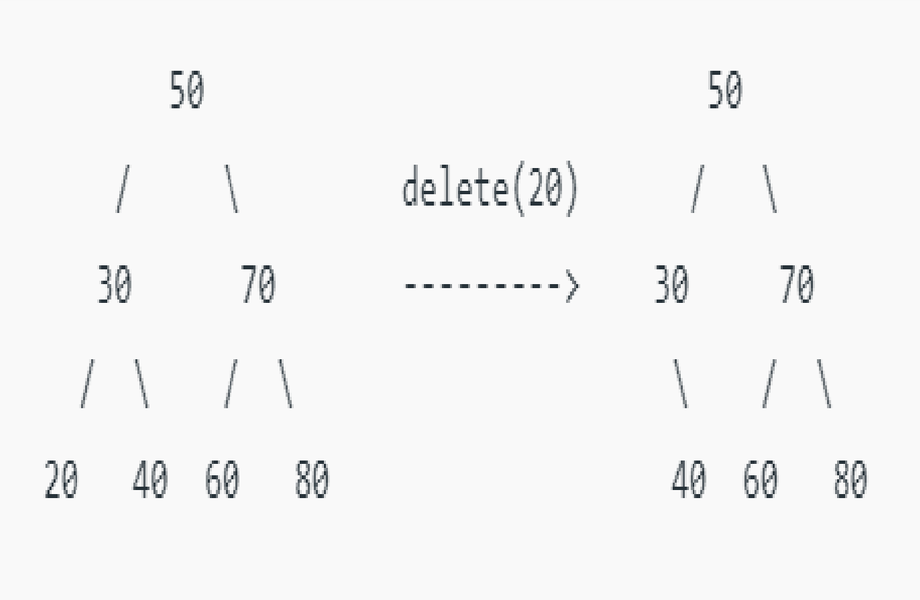
# Insertion




Insert (root, 16)

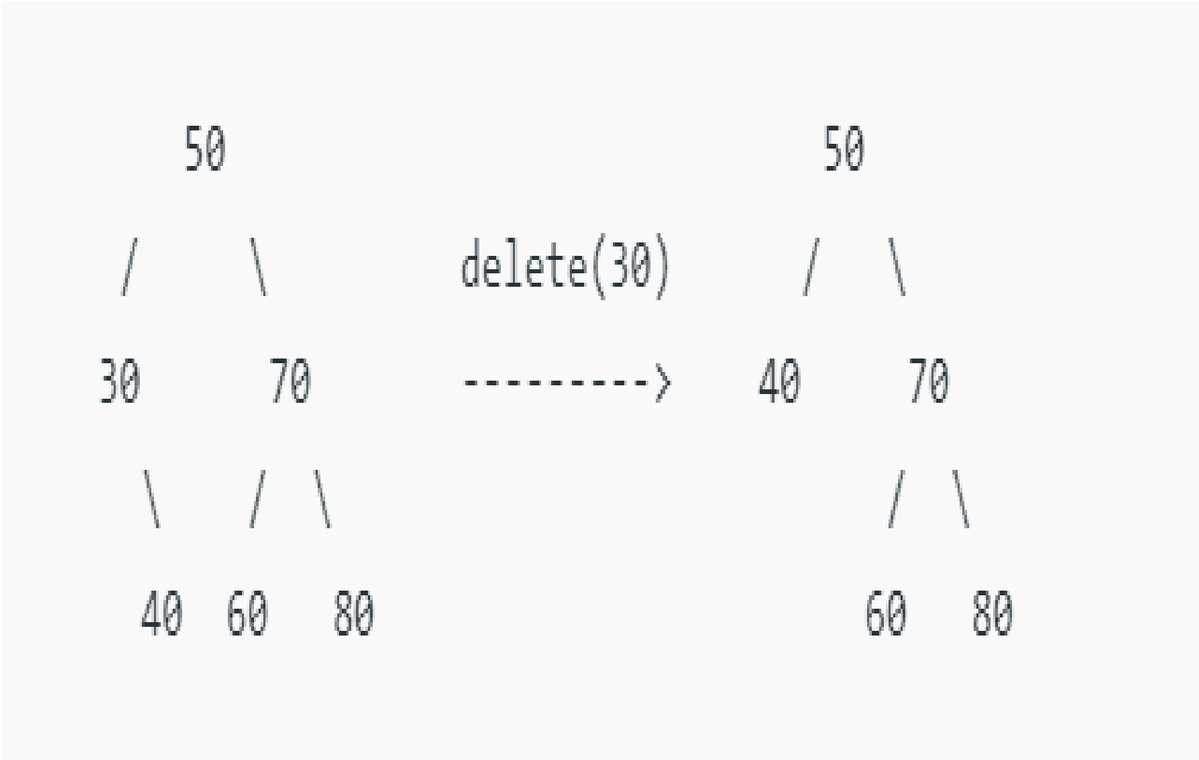
# Deletion




- **Node to be deleted is the leaf:**

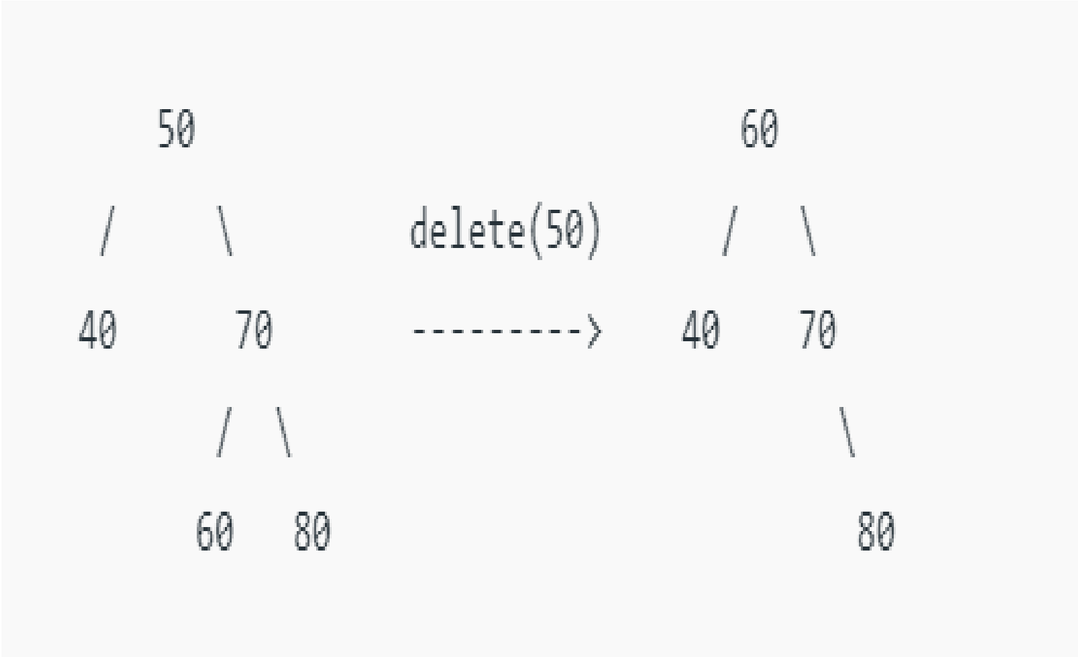
  Simply remove from the tree.

# Deletion




- **Node to be deleted has only one child:**
  Copy the child to the node and delete the child

# Deletion




- **Node to be deleted has two children:**

  Find inorder successor of the node. Copy contents of the inorder successor to the node and delete the inorder successor. Note that inorder predecessor can also be used.