St. Francis Institute of Technology, Mumbai-400 103
Department of Information Technology

A.Y. 2023-24
Class: SE-ITA/B, Semester: III
Subject: DATA STRUCTURE LAB

Experiment 3 : Binary search tree operations

1. **Aim:** Write a C program to implement a binary search tree using operations like insertion, deletion, counting of nodes, counting of leaf nodes etc.

2. **Objectives:** After study of this experiment, the student will be able to

- To use basic principles of programming as applied to complex data structures

- To implement the tree through programming

3. **Outcomes:** After study of this experiment, the student will be able to

- Formulate and solve problems of binary search tress and its operations
- Understand the concepts and apply the methods in basic trees

4. **Prerequisite:** Types of trees, Binary tree.

5. **Requirements:** PC and Turbo C compiler version 3.0

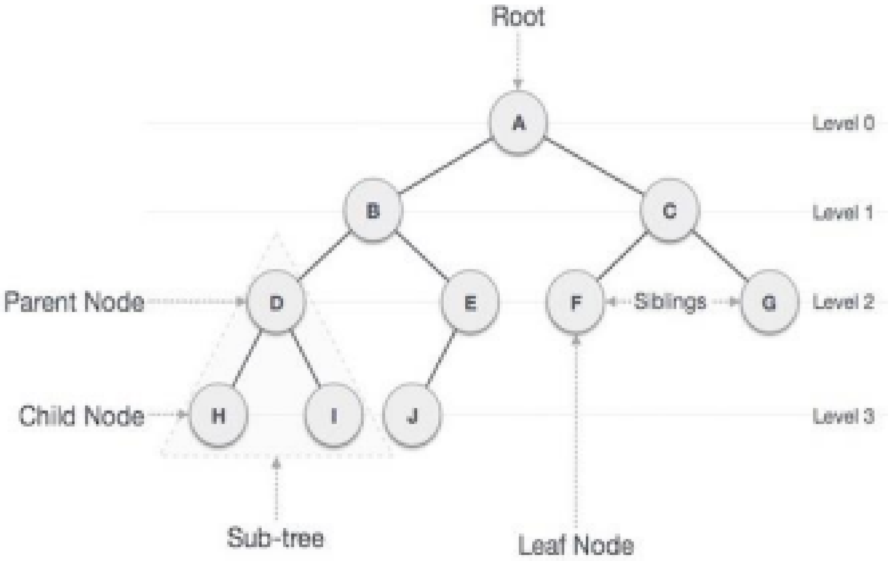
6. **Pre-Experiment Exercise:**

Brief Theory:

Explain the following terms

A. Tree

- Tree represents the nodes connected by edges. We will discuss binary tree or binary search tree specifically.
- Binary Tree is a special datastructure used for data storage purposes. A binary tree has a special condition that each node can have a maximum of two children. A binary tree has the benefits of both an ordered array and a linked list as search is as quick as in a sorted array and insertion or deletion operation are as fast as in linked list.

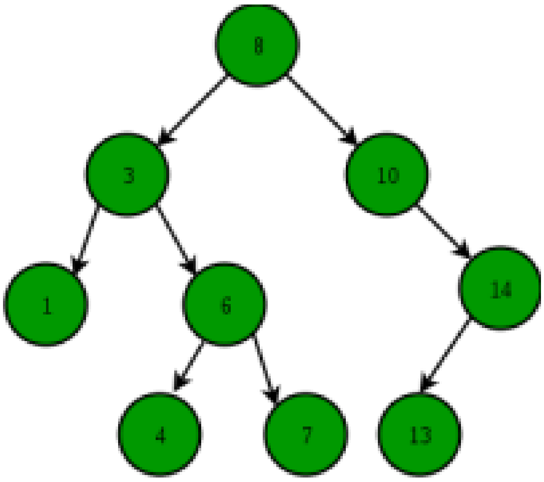


Following are the important terms with respect to tree.

- **Path** – Path refers to the sequence of nodes along the edges of a tree.
- **Root** – The node at the top of the tree is called root. There is only one root per tree and one path from the root node to any node.
- **Parent** – Any node except the root node has one edge upward to a node called parent.
- **Child** – The node below a given node connected by its edge downward is called its child node.
- **Leaf** – The node which does not have any child node is called the leaf node.
- **Subtree** – Subtree represents the descendants of a node.
- **Visiting** – Visiting refers to checking the value of a node when control is on the node.
- **Traversing** – Traversing means passing through nodes in a specific order.
- **Levels** – Level of a node represents the generation of a node. If the root node is at level 0, then its next child node is at level 1, its grandchild is at level 2, and so on.
- **keys** – Key represents a value of a node based on which a search operation is to be carried out for a node.

B. Binary Search Tree

- Every element has a unique key. The keys in a nonempty left subtree (right subtree) are smaller (larger) than the key in the root of subtree. The left and right subtrees are also binary search trees.



C. Operations on Binary Search Tree

Explain each of the following in details with suitable diagram.

- Traversing and displaying of Binary Search Tree
- Searching a node in Binary Search Tree
- Insert Node in Binary Search Tree
- Deleting a Node from a Binary Search Tree

7. Laboratory Exercise

A. Procedure

Write a C program to implement a binary search tree and show all the following operations in switch case,

i) Insertion
ii) Deletion
iii) Display Tree
iv) counting of nodes
v) counting of leaf nodes

B. Result/Observation/Program code:

Observe the output for the above code and print it.

8. Post-Experiments Exercise

A. Questions:

Construct the binary search tree corresponding to given numbers, 65,98,23,45,10,89,62,85,49,33.

B. Write In-order, postorder and preorder traversal of the the above tree

C. Explain the procedure to insert an element 87 into the binary tree.

D. Explain the Procedure of deleting element 62 from the tree.

9. Conclusion:

1. Summary of Experiment
2. Importance of Experiment

10. References:

1. S. K Srivastava, Deepali Srivastava; Data Structures through C in Depth; BPB Publications; 2011.
2. Reema Thareja; Data Structures using C; Oxford.
3. Data Structures A Pseudocode Approach with C, Richard F. Gilberg & Behrouz A. Forouzan, second edition, CENGAGE Learning.
