Department of Computer Science and Engineering,
Anna University, Chennai- 600025
CS23302 – Data Structures and Algorithms
Practical
July – November 2026
Year/Sem/Batch : II/III/ N

Exercise: 06	Binary Search Tree	07 – Aug - 2026

Observation (5 Marks)

1. A company's reporting hierarchy is structured as a binary tree, where each manager has at most two direct reportees. The CEO is the root.

   (a) Represent the given organizational hierarchy as a binary tree (node = employee name).

   (b) Write a program to construct this tree using linked representation (struct/class with left and right pointers).

   (c) Write a function to count the total number of employees (nodes) in the organization.

   (d) Write a function to find the "depth" of a specific employee (i.e., how many levels below the CEO they are).

2. A family tree is to be modeled where each person has at most two recorded children.

   (a) Implement the tree using array representation and linked representation. Compare the two in terms of memory usage.

   (b) Write a function to find the height of the family tree (i.e., the number of generations).

   (c) Write a function to count the number of leaf nodes (members with no recorded children).

   (d) Write a function to determine if the tree is a full binary tree (every person has either 0 or 2 children).

3. A file system directory structure is represented as a tree, where each folder can contain sub-folders (children).

   (a) Perform a preorder traversal to print the directory structure the way it would appear in a typical file explorer (parent listed before children).

   (b) Perform a postorder traversal to simulate deleting a directory — all sub-folders must be removed before the parent folder itself can be deleted. Explain why postorder is suitable here.

   (c) Perform a level-order (BFS) traversal to list all folders "level by level," as needed for a storage-usage report grouped by directory depth.

Execution (15 Marks)

1. A school wants to store student roll numbers in a BST for fast lookup, insertion, and deletion as students join or leave.

(a) Write a program to insert a sequence of roll numbers into a BST: {50, 30, 70, 20, 40, 60, 80, 10}.

(b) Write a function to search for a specific roll number and report the number of comparisons made — explain why this is O(log n) on average but O(n) in the worst case.

(c) Perform an inorder traversal of the resulting BST. What property do you observe about the output? Why is this always true for a BST?

(d) A student with roll number 40 leaves the school. Implement deletion and show the tree before and after, handling all three cases (leaf node, one child, two children).

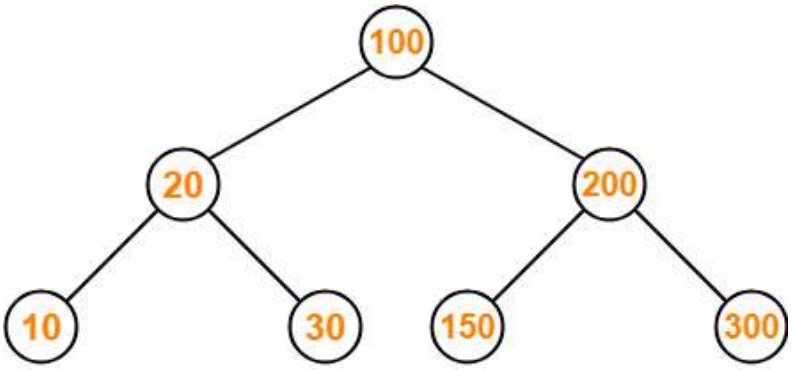
2. (a) Construct a Binary Search Tree (BST) for the following sequence of numbers-

50, 15, 62, 5, 20, 58, 91, 3, 8, 37, 60, 24

(b) Write the number of nodes in left sub tree and right subtree

(c) How many distinct binary search trees can be constructed out of 4 distinct keys?

(d) Write all the traversal sequences of the given BST



Binary Search Tree

(e) Discuss about the time complexity of BST operations.

3. Construct a Binary Search Tree with the following alphabets

M,R,I,L,E,K,O,U,P,R,T,G

and do the basic operations insert , delete and search

(i) Insert the alphabets Q and V

(ii) Delete the alphabets G,O,M

(iii) Search an alphabet E