Lab Experiment: 09 **Batch: 1 & 2**
Subject: Data Structures Lab **MCA**
Semester: 1st

Objective:

To understand the structure and implementation of binary trees using arrays and linked lists.

- To perform various tree traversal techniques (in-order, pre-order, post-order, and level-order).
- To implement heap sorting using a binary tree structure.

Instructions:

Implement the following tasks in C. Use appropriate data structures (array or linked list) to create the binary tree and demonstrate traversal methods and heap sorting.

1st Assignment: Binary Tree Creation

Using Arrays:

- Represent a complete binary tree using an array.
  - Note that for a node at index i:
    - The left child is at 2 * i + 1
    - The right child is at 2 * i + 2

Using Linked Lists:

- Represent a binary tree where each node contains data and pointers to its left and right children.
- Include functions to create and insert nodes in the binary tree.

2nd Assignment: Tree Traversal Methods

Implement the following traversal methods:

In-order Traversal:

- Traverse the left subtree, visit the root node, then traverse the right subtree.

Pre-order Traversal:

- Visit the root node, traverse the left subtree, then traverse the right subtree.

Post-order Traversal:

- Traverse the left subtree, traverse the right subtree, then visit the root node.

Level-order Traversal:

- Traverse the nodes level by level, starting from the root.

Implement each traversal function and test them with the binary tree created above.

Instructions for Submission

1. Implement the above tasks in C, ensuring each function works as expected.
2. Capture the output for each function (tree traversal and heap sort).
3. Document each step and observation.
4. Submit a PDF containing the following:
    - C Code: Include all implemented code sections.
    - Output Screenshots: Attach screenshots of the code output for each function.
    - Explanation: Provide explanations for each step of the code, including observations and results.