**National University of Computer & Emerging Sciences (NUCES) Islamabad,**

Department of Computer Science

**DATA STRUCTURES – FALL 2023**

**LAB 09**

**Learning Outcomes**

In this lab, you will implement the Expression Tree and Tree Traversal Techniques.

**Tree ADT**

**Tree** is a widely used nonlinear ADT. **Trees** structure data **hierarchically** based on relationships between the data elements. This means we can access the elements of the tree **non-sequentially**, it also allows trees to use memory more efficiently than linear types.

**Structure of Tree**

The data elements that make up a tree are called nodes and each node contains some data and pointers that connect the current node to the next nodes. Nodes can have none, one, or many children nodes.

5

3

8

**Terminologies in Tree ADT**

• Root - the top most node in a tree.

• Parent - A node that is the predecessor of any node.

• Siblings - nodes with the same parent.

• Leaf - a node with no children.

• Internal node - a node with at least one child.

**Binary Tree**

A binary tree is a tree which has at most two children at each node.

-

4

\*

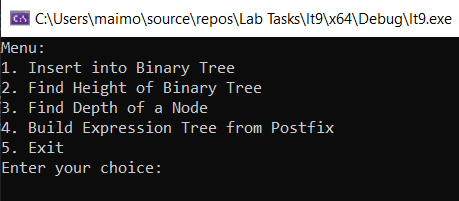
3

+

6

2

**Note: Your program should be menu-based program that allows the user to select and perform each task independently and provide the necessary input and output as specified in the tasks.**



# **TASK 1**

Create a function **insert** that will insert a node in the binary tree. (30 minutes)

# **TASK 2**

Write a function that finds the height of a binary tree. (30 minutes)

# **TASK 3**

Write a function that finds the depth of a particular node in a binary tree. (30 minutes)

# **TASK 4**

**Binary Expression Tree** (30 minutes)

A binary expression tree is a specific kind of a binary tree used to represent expressions. Each internal node of binary expression tree corresponds to an operator and each leaf node corresponds to an operand.

Expression = 4 - ((6+2) \*3)

**Expression Tree**

-

4

\*

3

+

6

2

Given a postfix expression, create an expression tree of that expression.