Modern Education Society's Wadia College of Engineering, Pune

210256: DATA STRUCTURES and ALGORITHM LABORATORY (2019 C0URSE)

NAME OF STUDENT:	**CLASS:**
SEMESTER/YEAR:	**ROLL NO:**
DATE OF PERFORMANCE:	**DATE OF SUBMISSION:**
EXAMINED BY:	**EXPERIMENT NO: B7**

TITLE: To illustrate the various Binary Tree functions.

AIM/PROBLEM STATEMENT: For given expression eg. a-b*c-d/e+f construct inorder sequence and traverse it using postorder traversal (non-recursive) and delete the entire tree.

OBJECTIVES:

1. To understand concept of Tree & Binary Tree.

2. To understand concept & features of object oriented programming.

OUTCOMES:

1. To analyze the working of various Tree operations.

THEORY:

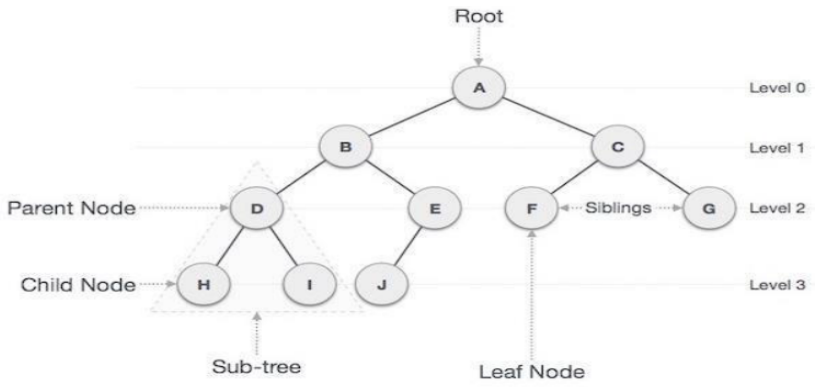
Tree

Tree represents the nodes connected by edges also a class of graphs that is acyclic is termed as trees. Let us now discuss an important class of graphs called trees and its associated terminology. Trees are useful in describing any structure that involves hierarchy. Familiar examples of such structures are family trees, the hierarchy of positions in an organization, and so on.

Binary Tree

A binary tree is made of nodes, where each node contains a "left" reference, a "right" reference, and a data element. The topmost node in the tree is called the root.

Every node (excluding a root) in a tree is connected by a directed edge from exactly one other node. This node is called a parent. On the other hand, each node can be connected to arbitrary number of nodes, called children. Nodes with no children are called leaves, or external nodes. Nodes which not leaves are called internal nodes. Nodes with the same parent are called siblings.



Insert Operation

The very first insertion creates the tree. Afterwards, whenever an element is to be inserted, first locate its proper location. Start searching from the root node, then if the data is less than the key value, search for the empty location in the left subtree and insert the data. Otherwise, search for the empty location in the right subtree and insert the data.

Traversals

A traversal is a process that visits all the nodes in the tree. Since a tree is a nonlinear data structure, there is no unique traversal. We will consider several traversal algorithms with we group in the following two kinds: Depth-first traversal , Breadth-first traversal

There are 3 different types of depth-first traversals:

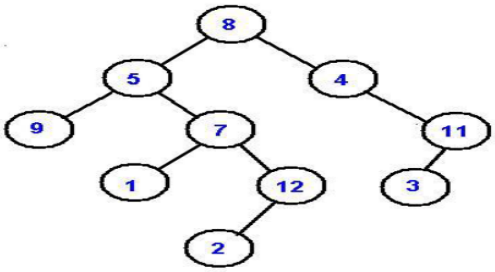
PreOrder traversal - visit the parent first and then left and right children

InOrder traversal - visit the left child, then the parent and the right child

PostOrder traversal - visit left child, then the right child and then the parent

There is only one kind of breadth-first traversal--the level order traversal. This traversal visits nodes by levels from top to bottom and from left to right.

As an example, consider the following tree and its four traversals:



PreOrder - 8, 5, 9, 7, 1, 12, 2, 4, 11, 3

InOrder - 9, 5, 1, 7, 2, 12, 8, 4, 3, 11

PostOrder - 9, 1, 2, 12, 7, 5, 3, 11, 4, 8

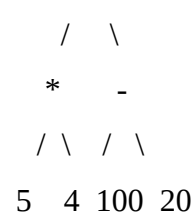
LevelOrder - 8, 5, 4, 9, 7, 11, 1, 12, 3, 2

QUESTIONS:

1. Consider the polynomial 5y 2 − 3y + 2.

(a) Write the polynomial as an expression tree that obeys the usual ordering of operations. Hint: to clarify the ordering for yourself, first write the expression with brackets indicating the order of operations. (b) Write the polynomial as postfix expression

2. Given a full binary expression tree consisting of basic binary operators (+ , – ,*, /) and some integers, Your task is to evaluate the expression tree.



3. (a) Draw the binary tree whose in-order traversal is DBEAFC and whose pre-order traversal is ABDECF.

(b) What is the post-order traversal of this tree?

(c) Draw all binary search trees of height 2 that can be made from all the letters ABCDEF, assuming the natural ordering.