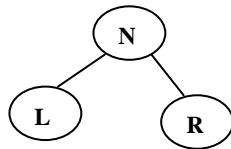


Binary Tree Traversals:

Traversal of a binary tree means to visit each node in the tree exactly once. The tree traversal is used in all tree concepts.

In a linear list nodes are visited from first to last, but a tree being a non linear one we need definite rules. There are 3 ways to traverse a tree. All of them differ only in the order in which they visit the nodes. Naming has been done on the basis of position of node(N), then left child(L) and right child(R).



The three main methods of traversing a tree are:

- Inorder Traversal (LNR)
- Preorder Traversal(NLR)
- Postorder Traversal(LRN)

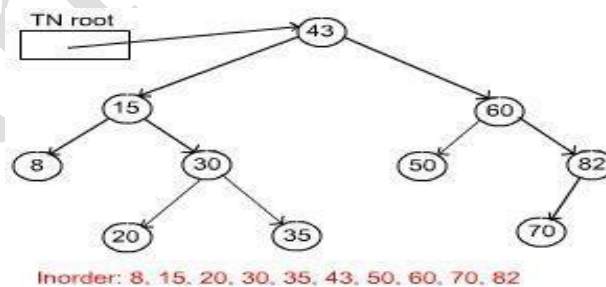
In all of them we do not require to do anything to traverse an empty tree. All the traversal methods are base functions since a binary tree is itself recursive as every child of a node in a binary tree is itself a binary tree.

Inorder Traversal:

To traverse a non empty tree in inorder the following steps are followed recursively.

- Visit the Root
- Traverse the left subtree
- Traverse the right subtree

The inorder traversal of the tree shown below is as follows.



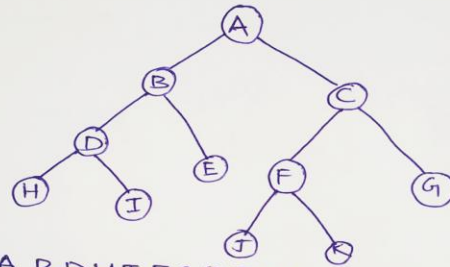
Preorder Traversal:

1. Visit the root.
2. Traverse the left sub-tree, i.e., call Pre-order(left-sub-tree)
3. Traverse the right sub-tree, i.e., call Pre-order(right-sub-tree)

Post-order Traversal:

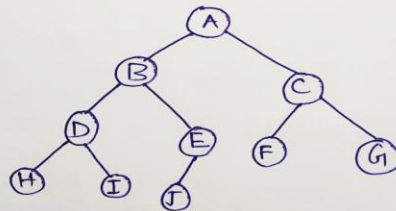
1. Traverse the left sub-tree, i.e., call Post-order(left-sub-tree)
2. Traverse the right sub-tree, i.e., call Post-order(right-sub-tree)
3. Visit the root.

Example:- Traverse the tree.



Preorder:- A B D H I E C F J K G
Inorder:- H D I B E A J F K C G
Postorder:- H I D E B J K F G C A

Example:- Traverse the tree.



Preorder:- A B D H I E J C F G
Inorder:- H D I B J E A F C G
Postorder:- H I D J E B F G C A

Construction of Binary Tree

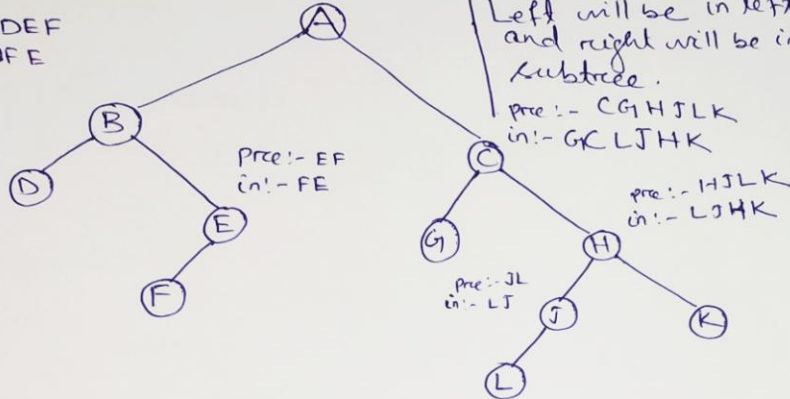
A binary tree can be constructed by using any two traversal mechanisms. The examples are described below where inorder traversal is given to make construction easier one. Also if inorder traversal is not given still we can construct binary tree.

Construction of binary tree

Example:-

Preorder:- \rightarrow ABDEFCHJLK
Inorder:- DBFEAGCLJHK

pre:- BDEF
in:- DBFE



The first node of preorder will be the root node due to its traversal mechanism (NLR). Find its position in inorder. Left will be in left subtree and right will be in right subtree.

pre:- CHJLK
in:- GCLJHK

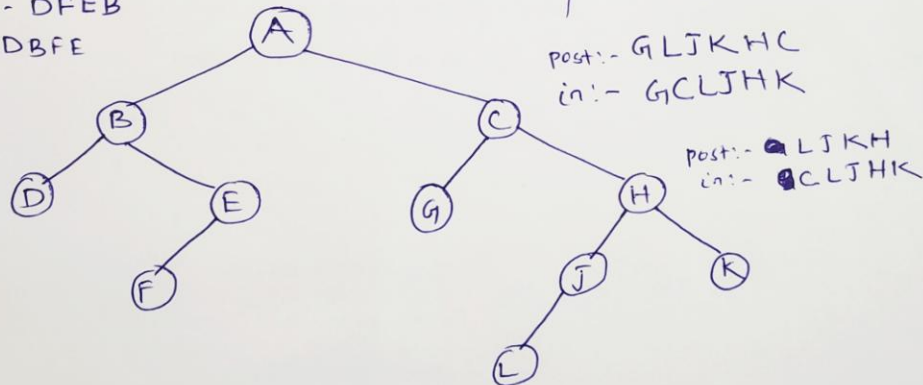
pre:- HJLK
in:- LJHK

pre:- JL
in:- LJ

Example:-

Postorder:- DFEBGLJHKCA
inorder:- DBFEAGCLJHK

post:- DFEB
in:- DBFE



post:- GLJHK
in:- GCLJHK

post:- LJKH
in:- GCLJHK

The last node of postorder will be the root node due to its traversal mechanism (LRN) and rest same as before.

Everybody should learn programming language and write programs, because it makes you think for some time and utilize your brain.

**With Regards
Yours Sam Sir**