07

Tree Data Structure

DATA STRUCTURE LAB

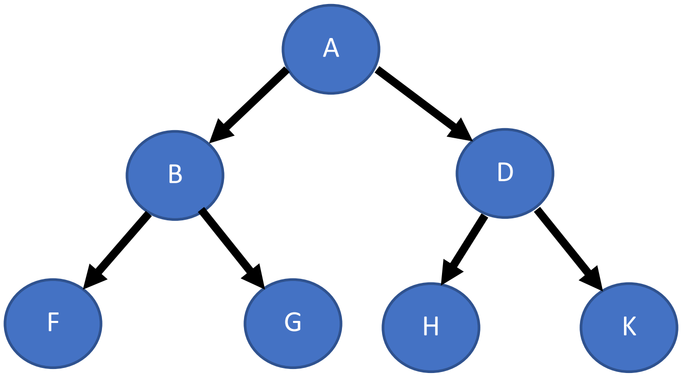
SESSION - 07

CSE135 | DAFFODIL INTERNATIONAL UNIVERSITY

TREE

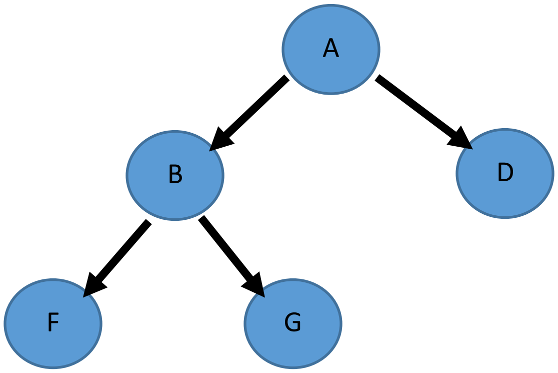
Full Binary Tree:

Binary tree with required number of child as per level



Complete Binary Tree:

Binary tree but childs from left to right



Array to Tree:

0	1	2	3	4
7	5	4	6	9

→ X[i] = root;   i=0

→ Left will be = x[2i+1]

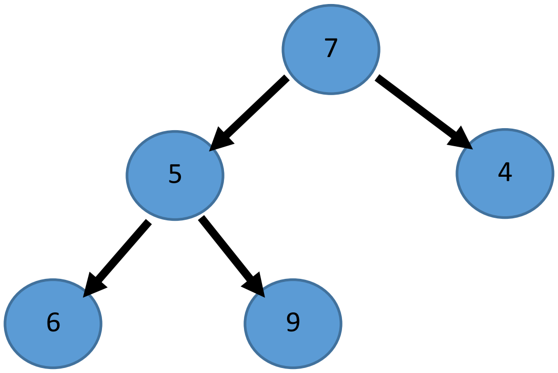
When i=0   x[1]=left

When i=1 x[3]=left

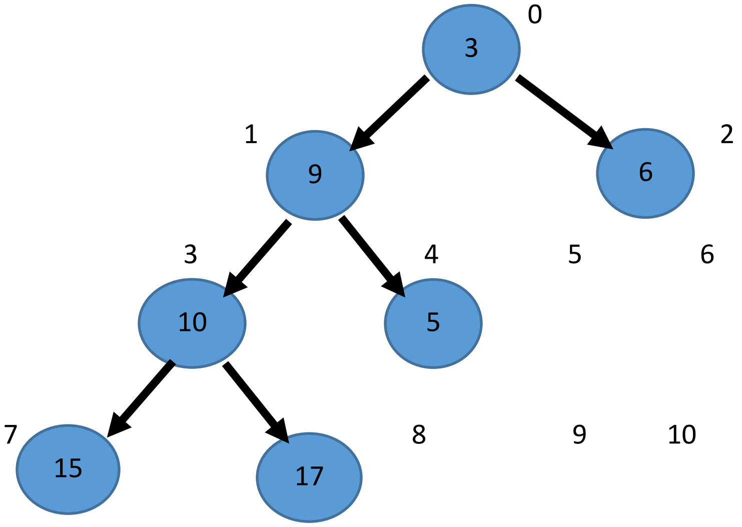
→ Right will be = x[2i+2]

When i=0 x[2]=right

When i=1 x[4]=right



3	9	6	10	5			15	17	
0	1	2	3	4	5	6	7	8	9



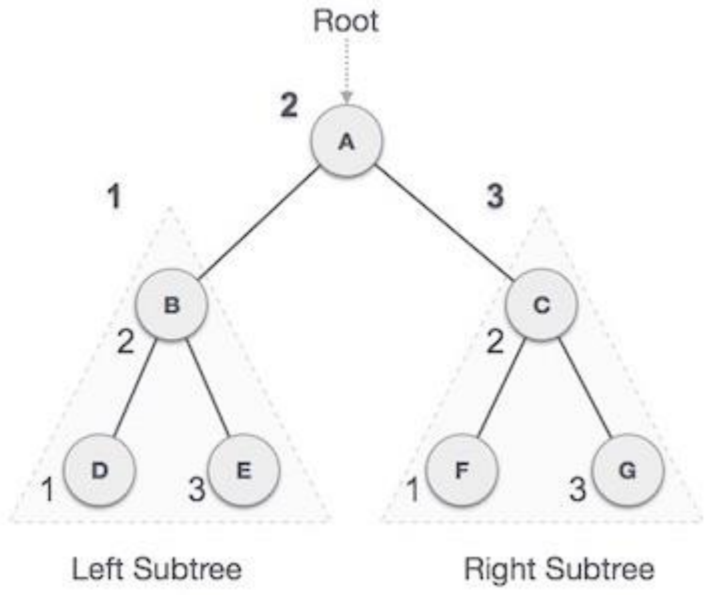
Tree Traversal:

Traversal is a process to visit all the nodes of a tree.

- → In-order Traversal
- → Pre-order Traversal
- → Post-order Traversal

In-order Traversal:

In this traversal method, the left subtree is visited first, then the root and later the right sub-tree.



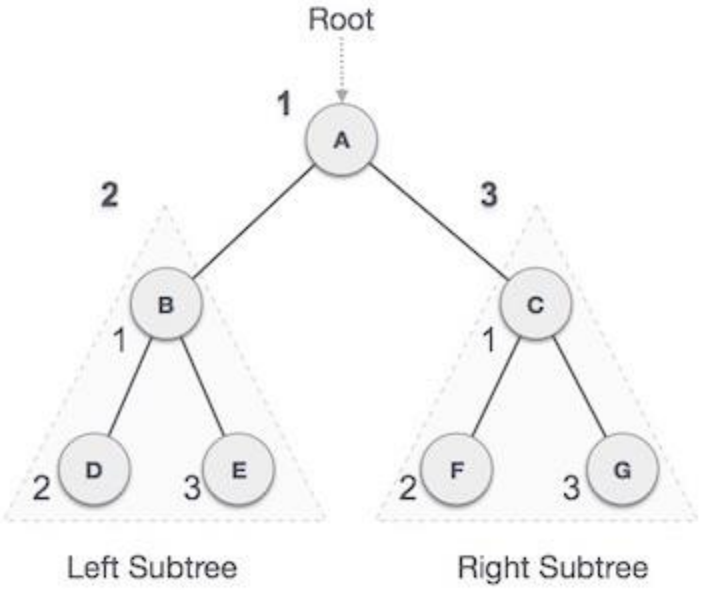
D → B → E → A → F → C → G

In-order Implementation:

```
void inorder(tree *t)
{
        if(t)
        {
                        inorder(t->left);
                        printf("%d\n",t->data);
                        inorder(t->right);
        }
}
```

Pre-order Traversal:

In this traversal method, the root node is visited first, then the left subtree and finally the right subtree.



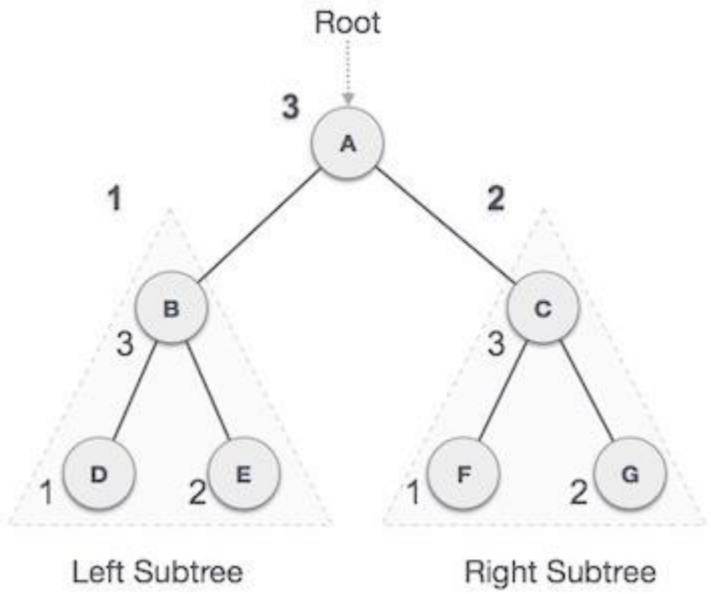
A → B → D → E → C → F → G

Pre-order Implementation:

```
void preorder(tree *t)
{
        if(t)
        {
                        printf("%d\n",t->data);
                        preorder(t->left);
                        preorder(t->right);
        }
}
```

Post-order Traversal:

In this traversal method, the root node is visited last, hence the name. First we traverse the left subtree, then the right subtree and finally the root node.



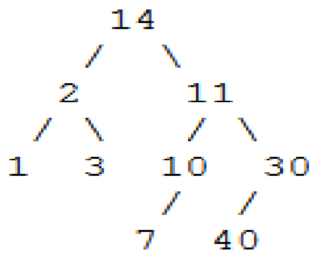
D → E → B → F → G → C → A

Post-order Implementation:

```
void postorder(tree *t)
{
        if(t)
        {
                        postorder(t->left);
                        postorder(t->right);
                        printf("%d\n",t->data);
        }
}
```

##EXERCISE :

1. For the following binary tree answer the following questions:



```
        14
       /  \
     2     11
    / \    / \
   1   3  10  30
          /   /
         7   40
```

a. Find height, depth, size of the tree
b. Write the pre-order, in-order and post-order traversal of the tree.
c. Make a
d. Convert the given tree into max heap & min heap.

Find sum of all left leaves in a given Binary Tree