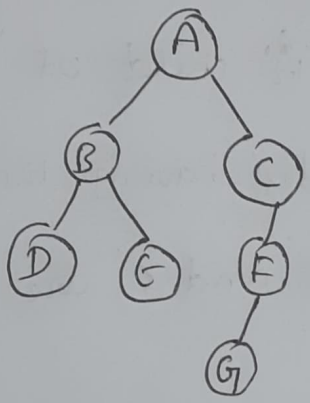
Trees

A tree is a set of one or more nodes that shows Parent-child relation such that:

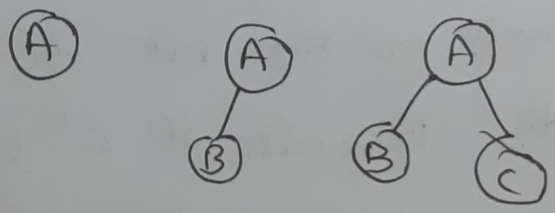
- there is a special node called the root node.
- the remaining nodes are partioned into disjoint subsets T1, T2, T3 ... Tn, $n \geq 0$ where T1, T2, T3 ... Tn which are all children of root node are themselves trees called subtrees.



A is the root node and B, C are children of A.

Binary tree :- A tree in which each node has either zero, one or two subtrees is called a binary tree.

Ex:-



Examples of Binary tree

Ex:-

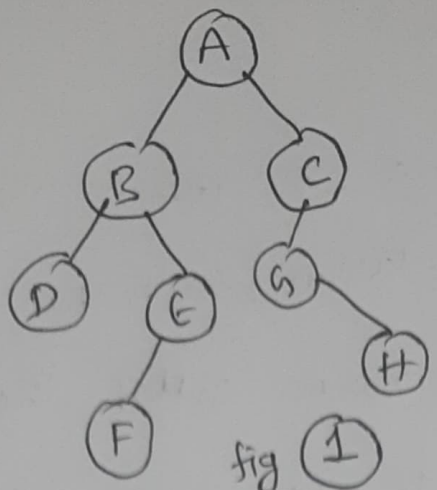


fig (1)

The following definitions are explained with respect to fig (1)

Root node:- A first node written at the top is root node. The root node does not have the Parent. 'A' is the root node.

Child:- The node obtained from the Parent node is called child node. 'D' is child of 'B'.

Siblings:- Two or more nodes having the same Parent are called siblings. 'D' and 'E' are siblings of same Parent 'B'.

Ancestors:- The nodes obtained in the Path from the specified node 'x' while moving upwards the root node are called ancestors. The Ancestors of node 'F' is A, B, E.

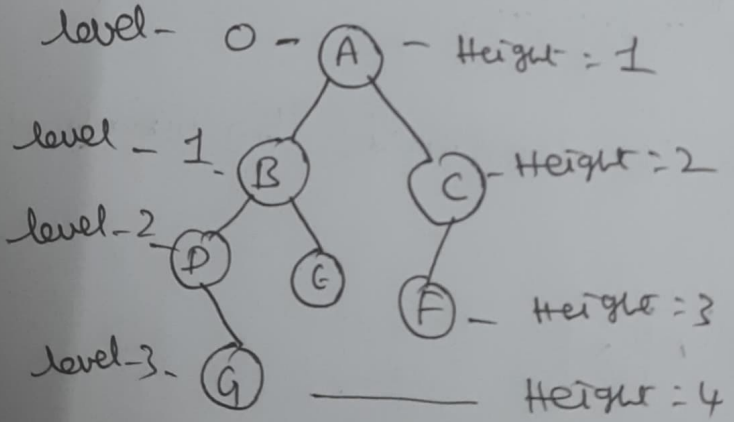
Descendents:- The nodes in the path below the Parent are called descendents. Descendents of 'C' is G and H

Degree:- The number of subtrees of a node is called its degree. The degree of 'B' is 2. The degree of 'C' is 1. The degree of 'E' is 0.

leaf:- A node in a tree that has a degree of zero is called a leaf node. D, F and H are leaf nodes.

Internal nodes:- The nodes except leaf nodes in a tree are called internal nodes.

ex:- B, C are internal nodes.



level:- The distance of a node from the root is called level of the node.

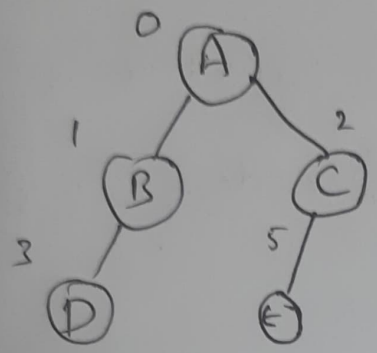
The root node is at level 0 and remaing nodes are at level + 1. The level of above tree is 3.

Height:- The height of the tree is defined as the maximum level of any leaf in the tree.

The height of above tree is 4.

Representation of Binary tree

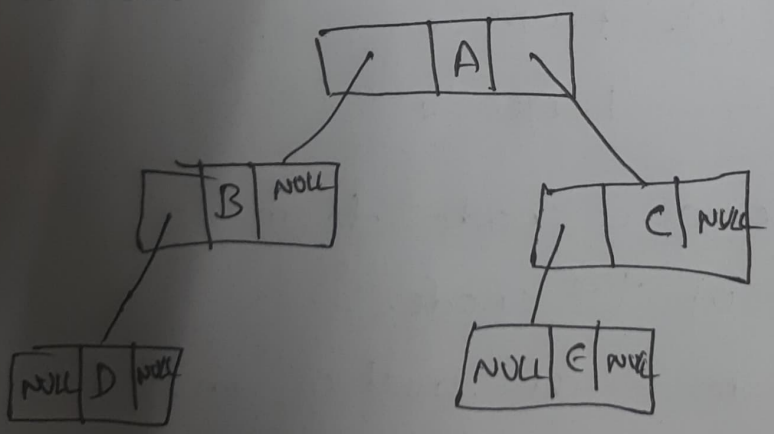
The Binary tree can be represented in memory by : 1. Array 2. linked list



The above tree can be represented by using array.

A	B	C	D	-	E	-
T(0)	T(1)	T(2)	T(3)	T(4)	T(5)	T(6)

The above tree can be represented by using doubly linked list.

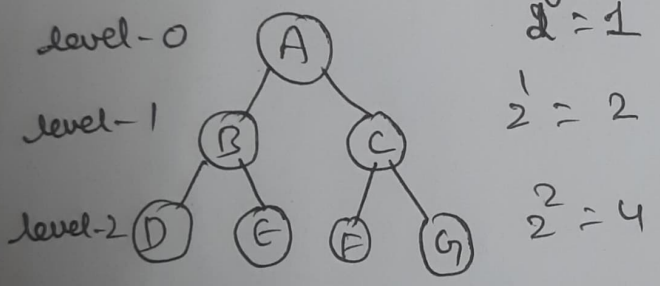


Types of Binary tree

There are different types of Binary trees:

1. Strictly binary tree (Full binary tree)
2. Skewed tree
3. Complete binary trees
4. An expression tree
5. Binary search tree (BST)

1. **Strictly Binary tree:-** A Binary tree having 2^i nodes in any given level 'i' is called Strictly Binary tree.

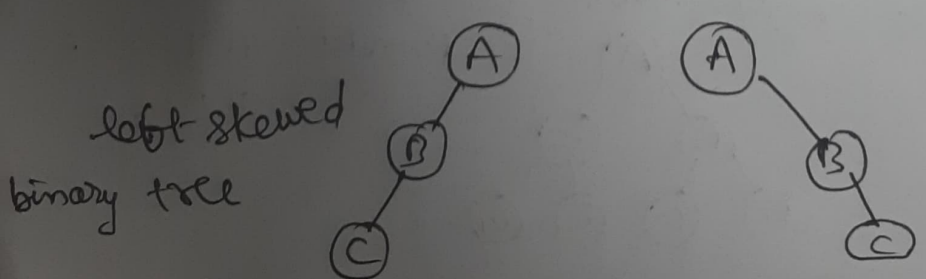


$2^0 = 1$

$2^1 = 2$

$2^2 = 4$

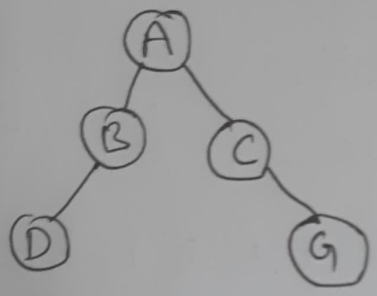
2. **Skewed tree:-** A skewed tree is a tree consisting of only left subtree or only right subtree. A tree with only left subtree is called left skewed binary tree and a tree with only right subtree is called Right skewed binary tree.



Right skewed
binary tree.

Complete binary tree

A tree in which every level, except possibly the last level is completely filled.



Tree travelsal techniques

Traversing is a method of visiting each node of a tree exactly once in a systematic order.

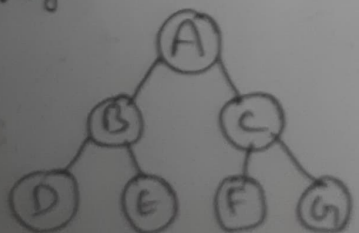
There are three tree travelsal techniques

1. Preorder
2. Inorder
3. postorder

Preorder

The steps (algorithm) for preorder

1. Process the root node
2. process the left subtree in preorder
3. process the right subtree in preorder

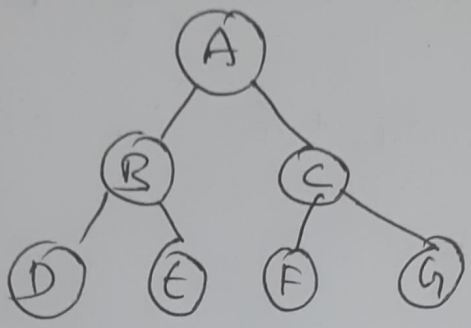


Preorder :- A B D E C F G

Inorder tree traversal

Steps for Inorder tree traversal:

1. Process the left subtree in inorder
2. Process the root node
3. Process the right subtree in inorder



In-order: D B E A F C G

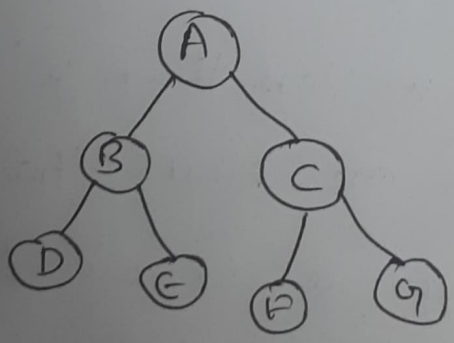
Postorder tree traversal

Steps for postorder tree traversal:

1: Process the left subtree in postorder

2: Process the right subtree in postorder

3: Process the root node.



Pst-order: D E B F G C A

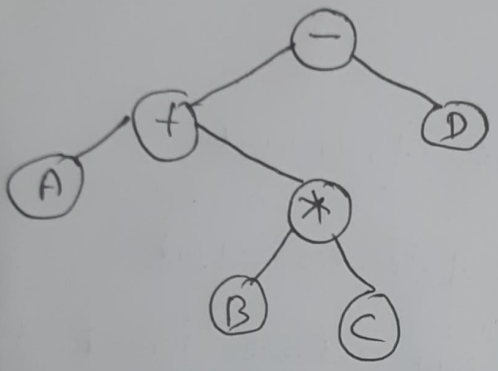
Expression tree

An expression tree is a binary tree that satisfy the following properties:

1. All leaf nodes are operands
2. All parent nodes are operators.

Construct a binary tree for the following expression.

A + B * C - D

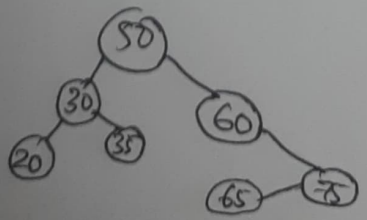


Binary search tree

A binary search tree is a binary tree in which for each node say 'x' in the tree, elements in the left-subtree are less than 'x' and elements in the right subtree are greater than x.

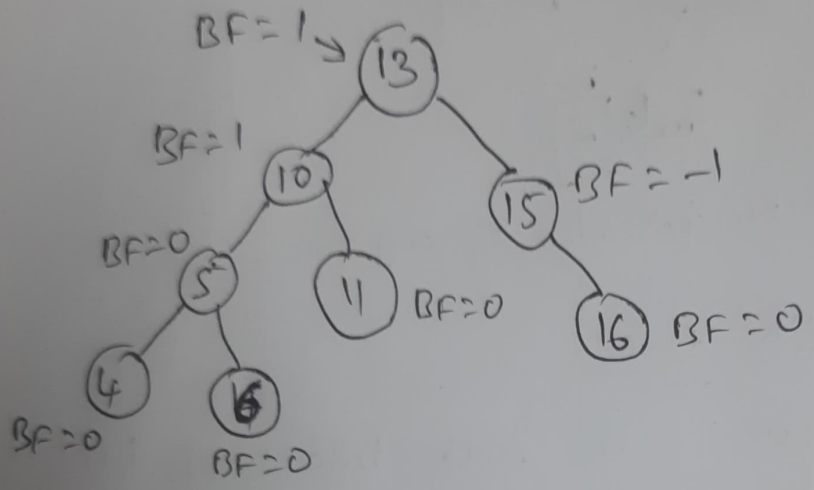
for ex:- Construct a binary tree for the following:

50, 60, 30, 35, 20, 75, 65



AVL tree

An AVL tree is a self-balancing binary search tree (BST) where, for any node, the heights of its left and right subtrees differ by at most one.



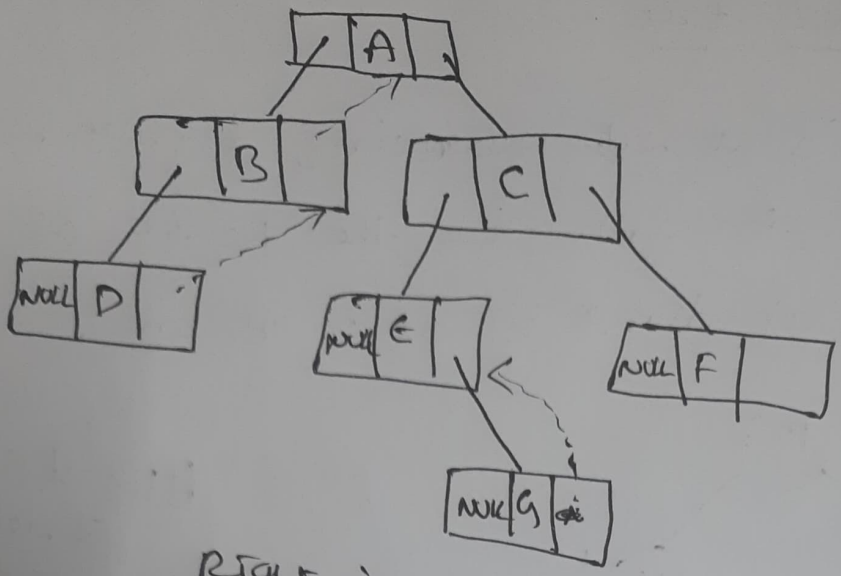
BF = height of left subtree − height of right subtree

Balance Factor (BF) is defined as difference between heights of left subtree and right subtree.

In AVL tree, the Balance factor (bf) of any node either 0, 1 or -1.

Threaded binary tree

In a binary tree, more than 50% of link fields have 'null' values and more space is wasted by presence of 'null' values. These link fields which contain null characters are replaced by address of some nodes in the tree which facilitate upward movement in the tree. These extra links which contains addresses of some nodes are called threads and the tree is termed as threaded binary tree.



Right in-threaded binary tree.

Red-Black tree

A Red-Black tree is a self-balancing binary search tree where each node has an additional attribute: a color, which can be either red or black.

Properties of Red-Black trees.

1. Each node is either red or black.
2. The root of the tree is always black.
3. Red nodes cannot have red children
4. Every path from a node to its descendent null nodes has the same number of black nodes
5. All leaves are black.