1. **Finding how many binary tree generated if unlabeled nodes are given formula**

Catalan Number: T(n) = 2nCn/n+1

Ex: 5 node T(n) = 2x5 C5/5+1 = T(n) = ~~10~~ x ~~9~~(3) x ~~8~~(2) x 7 x ~~6~~ / ~~5~~ x ~~4~~ x ~~3~~ x ~~2~~ x 1 / ~~6~~ = T(n) = 3 x 2 x 7 = 42

42 binary trees are generated if 5 nodes are given

2. **Finding how many permutation binary tree generated if labeled nodes are given formula**

Catalan Number: T(n) = 2nCn/n+1 x n!

Ex: 3 node T(n) = 2x3 C3/3+1 x 3! = T(n) = ~~6~~ x 5 x ~~4~~ / ~~3~~ x ~~2~~ x 1 / ~~4~~ x 3 x 2 x 1 = 5 x 3 x 2 x 1 = 30

30 binary trees are generated if 3 nodes are given

3. **Finding how many binary tree generated with maximum height if nodes are given**

Formula: 2^(n-1)

Ex. 5 nodes, 2^(5-1) = 2^4 = 2 x 2 x 2 x 2 = 16 binary tree generated with maximum height

4. **Height Vs Nodes**

Finding height of binary tree if nodes are given

Formula = logbase2 (n+1)-1 (for min nodes) <= h <= n-1 (for max nodes)

Finding nodes of binary tree if height is given

Formula = h+1(for min nodes) <= n <= (2^h+1) - 1 (for max nodes)

5. **Strict Binary Tree (proper binary tree / complete binary tree)**

For general binary tree node having 0, 1 and 2 children

For strict binary tree node having 0 and 2 children

Finding nodes of binary tree if height is given

Formula = 2h+1(for min nodes) <= n <= (2^h+1) - 1 (for max nodes)

Finding height of binary tree if nodes are given

Formula = logbase2 (n+1)-1 (for min height) <= h <= n-1 / 2 (for max height)

Number of external nodes/leaf nodes = internal nodes/non leaf nodes + 1

6. **N-ary Trees**

In 3-ary tree node having only {0, 1, 2, 3} degree / child nodes (based on capacity)

In 4-ary tree node having only {0, 1, 2, 3, 4} degree / child nodes (based on capacity)

And so on ……

7. **Strict N-ary Trees**

In 3-ary tree node having only {0, 3} degree / child nodes (based on capacity)

In 3-ary tree node having only {0, 4} degree / child nodes (based on capacity)

And so on ……

Finding nodes of n-ary tree if height is given

Formula = mh+1 (for min nodes) <= n <= (m^h+1) -1 / m - 1 (for max nodes)

Finding height of binary tree if nodes are given

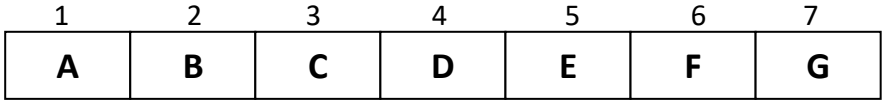
Formula = logbase2 [n (m + 1) + 1] - 1 (for min height) <= h <= n-1 / m (for max height)

Number of external nodes/leaf nodes = (m - 1) internal nodes + 1

8. **Representation of Binary Tree using Array**

Finding left and right child of binary tree in array

Array starts from 1 index



Element = i

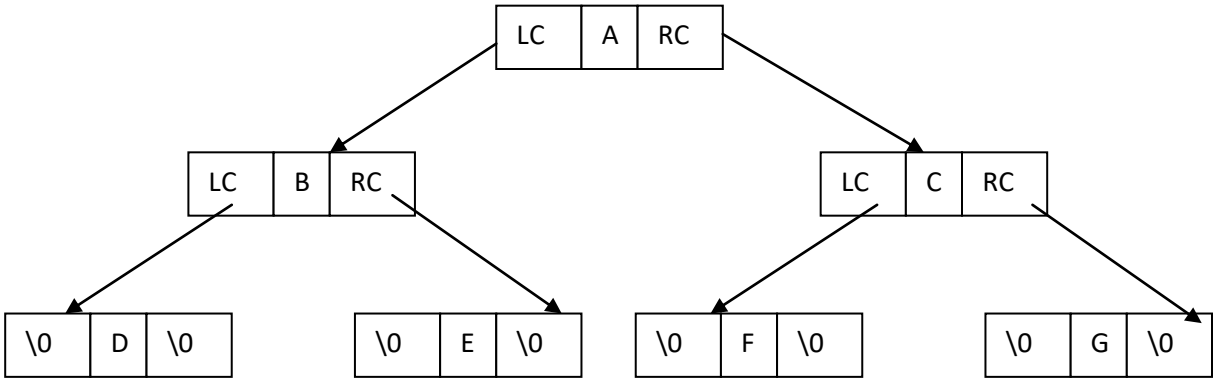
Left Child = 2 x i

Right Child = 2 x i + 1

Finding Parent of left and right child of binary tree in array

Parent = [i / 2] floor value

9. **Representation of Binary Tree using Linked**



Always n + 1 = NULL pointer

10. **Full Binary Tree Vs Complete Binary Tree**

Full binary tree height have maximum number of nodes (h = (2^h+1) - 1)

Complete binary tree of height h will be a full binary tree h – 1 and in the last level elements will must be filled from left to right without skipping any element.

Full binary tree is always a complete binary tree but complete binary tree need not be a full binary tree.