Trees Formulaes & Properties

No of different binary trees with n no of nodes

→ unlabelled nodes {Selectⁿ}

$$T(n) = \text{catalan}(n) = \frac{{}^{2n}C_n}{n+1}$$

→ labelled nodes {Selectⁿ & Permutatⁿ}

$$T(n) = \frac{{}^{2n}C_n}{n+1} \times n!$$

$$\text{catalan}(n) = \frac{{}^{2n}C_n}{n+1} = \sum_{i=1}^{n} \text{catalan}(i-1) \times \text{catalan}(n-i)$$

Given height of binary tree (h) {0-indexed} no of nodes $= n$?

$$n_{min} = h+1 \quad \{\text{skewed tree}\}$$

$$n_{max} = 1+2+2^2+\dots 2^n = 2^{h+1}-1$$

$$\Rightarrow \boxed{h+1 \leq n \leq 2^{h+1}-1}$$

Given no of nodes(n), height $h = ?$

$h_{max} = n-1$ {skewed tree}

$h_{min} = \log_2(n+1) - 1$

$\Rightarrow \boxed{\log_2(n+1)-1 \leq h \leq n-1}$

For binary tree

$$\boxed{n_{deg(0)} = n_{deg(2)} + 1}$$ ← Irrespective of value of $n_{deg(1)}$

$\boxed{\text{Internal Nodes} = \text{External Nodes} + 1}$

→ if $n_{deg(1)} = 0$ {All nodes have 0 or 2 children}
⇓
Strict binary tree

For $\boxed{\text{Strict / proper binary tree}}$ (0 or 2 children)

→ if height h is given $\boxed{2h+1 \leq n \leq 2^{h+1} - 1}$

min Nodes $n_{min} = 2h+1$

max Nodes $n_{max} = 2^{h+1} - 1$

→ if nodes 'n' are given $\boxed{\log_2(n+1) - 1 \leq h \leq (n-1)/2}$

min height $h_{min} = \log_2(n+1) - 1$

max height $h_{max} = \frac{n-1}{2}$

For strict m-ary trees { 0 or m children }

→ if h is given

$$n_{min} = mh+1$$

$$n_{max} = \frac{m^{h+1}-1}{m-1} \quad (= 1+m+m^2+ \dots m^h)$$

$$\Rightarrow \boxed{mh+1 \leq n \leq \frac{m^{h+1}-1}{m-1}}$$

→ if n is given

$$h_{min} = \log_m [n(m-1)+1] - 1$$

$$h_{max} = \frac{n-1}{m}$$

$$\boxed{\log_m [n(m-1)+1] - 1 \leq h \leq \frac{n-1}{m}}$$

→ External nodes (0 children) vs Internal nodes (m children)

$$\boxed{e = (m-1)i + 1}$$

Full Binary Tree

→ Binary tree of given height with maximum nodes possible

$$\Rightarrow n = 2^{h+1} - 1$$

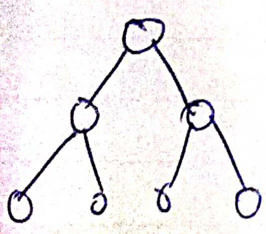
{ One extra node will increase height of tree }

Complete Binary Tree

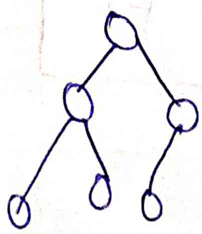
→ In array representation, there will be no blank spaces until last node

→ A complete binary tree of height h is either a full binary tree of height h or a full binary tree of height (h−1) + some nodes in left on h^{th} level without skipping any space.

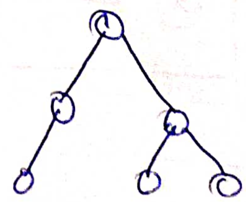
All full binary trees are complete but complete binary tree may or may not be full.



Both full & complete



Complete but not full



Neither complete nor full