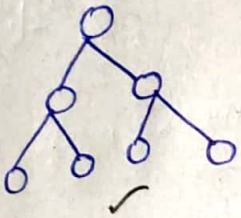
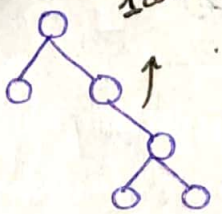
Strict Binary trees

A binary tree is a tree which has atmost 2 children. {0, 1, 2}

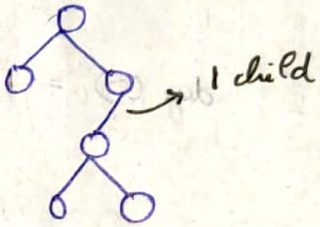
A strict binary tree is a tree which can have either 0 or 2 children and should not have ϕ child. } {0, 1, 2}



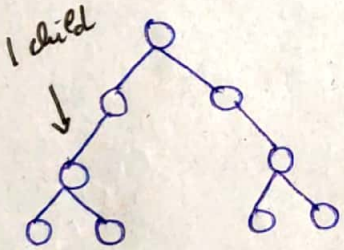
✓



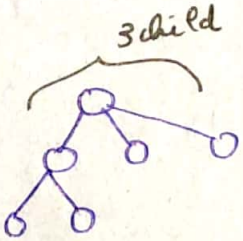
×



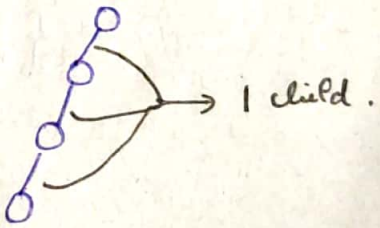
×



×



×

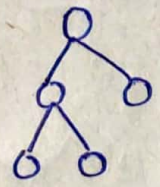


×

Height (vs) Nodes of strict Binary trees.

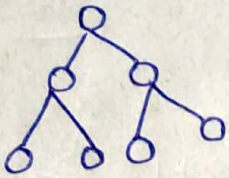
Height h=2

Min



n=5

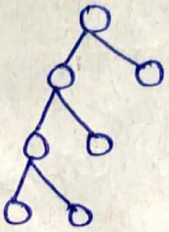
Max



n=7

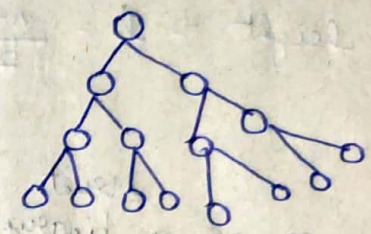
Height h=3

Min



n=7

Max

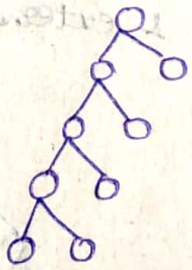


n=15

To calculate max & min number of nodes with a given height:

Height h=4

Min



n=9

Max

n=31

Max height of strict binary tree is similar to max height of binary tree.

Min number of nodes: $n = 2h+1$

Max number of nodes: $n = 2^{h+1} - 1$

To calculate min & max height with given number of nodes:

Min height $\Rightarrow h = \log_2(n+1) - 1$

Max height $\Rightarrow h = \frac{n-1}{2}$

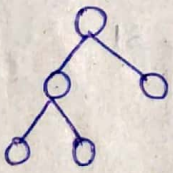
Height of a strict binary tree:

$$\log_2(n+1) - 1 \leq h \leq \frac{n-1}{2}$$

Nodes of a strict binary tree:

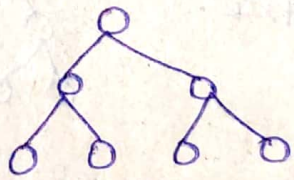
$$2h+1 \leq n \leq 2^{h+1} - 1$$

Rt bt internal & external nodes



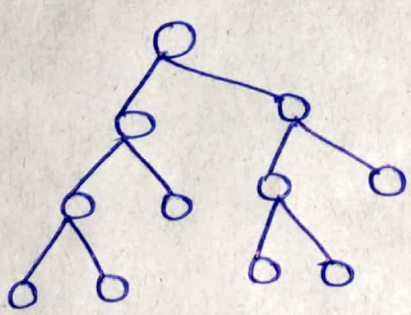
$i = 2$ (non leaf nodes)

$e = 3$ (leaf nodes)



$i = 3$

$e = 4$



$i = 5$

$e = 6$

∴ For strict Binary tree

External node = Internal node + 1.