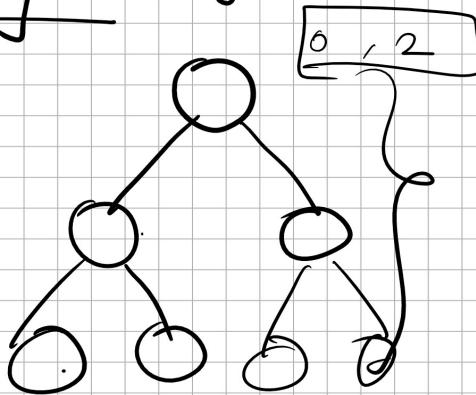


⇒ Maximum Depth of a Binary Tree. {.

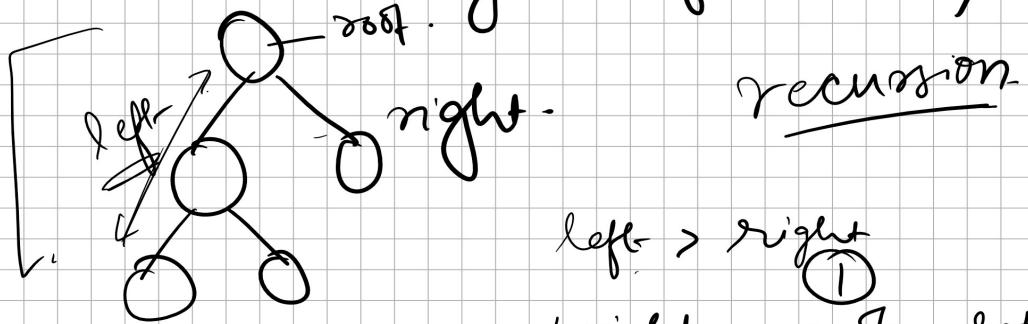
A generic tree which

has at most 2 child nodes for each parent node.



max Depth = height of a tree.

How to calc height of a Binary Tree?



base condition
if root is null, 0
height

if (root == null)
return 0;

$$\text{height} = \boxed{\text{root}} + \text{right}$$

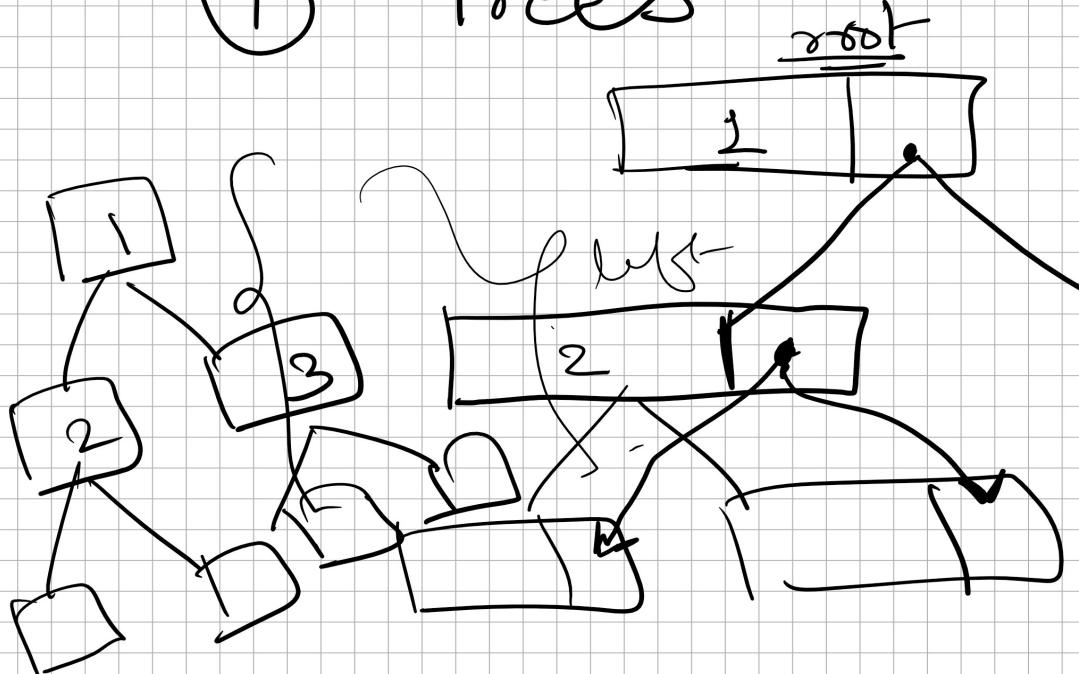
$$\text{height} = \boxed{\text{root}} + \text{left}$$



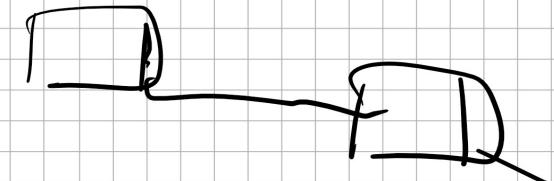
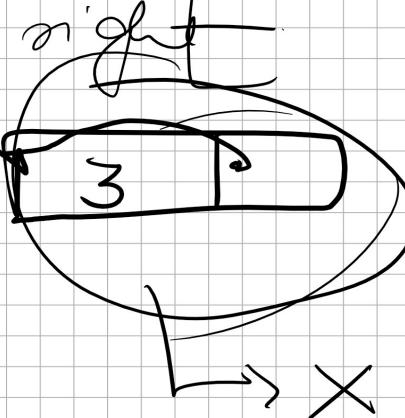
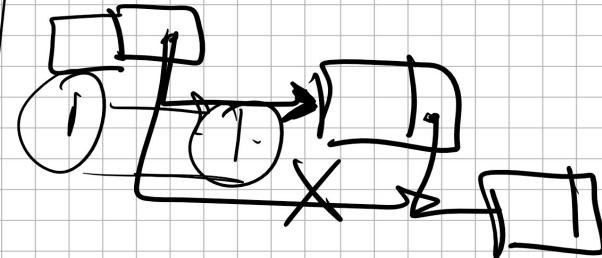
Max Depth.

①

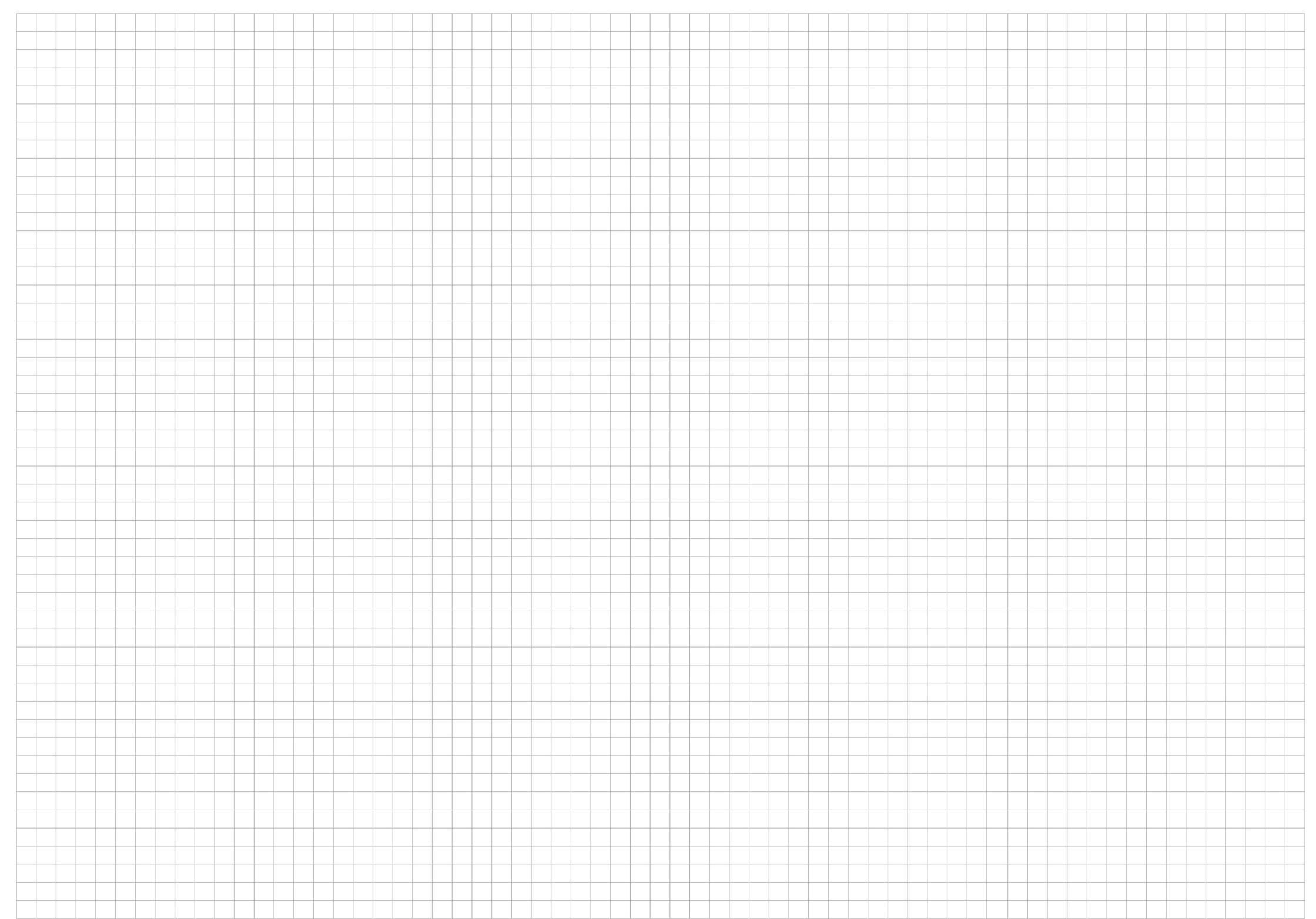
Trees

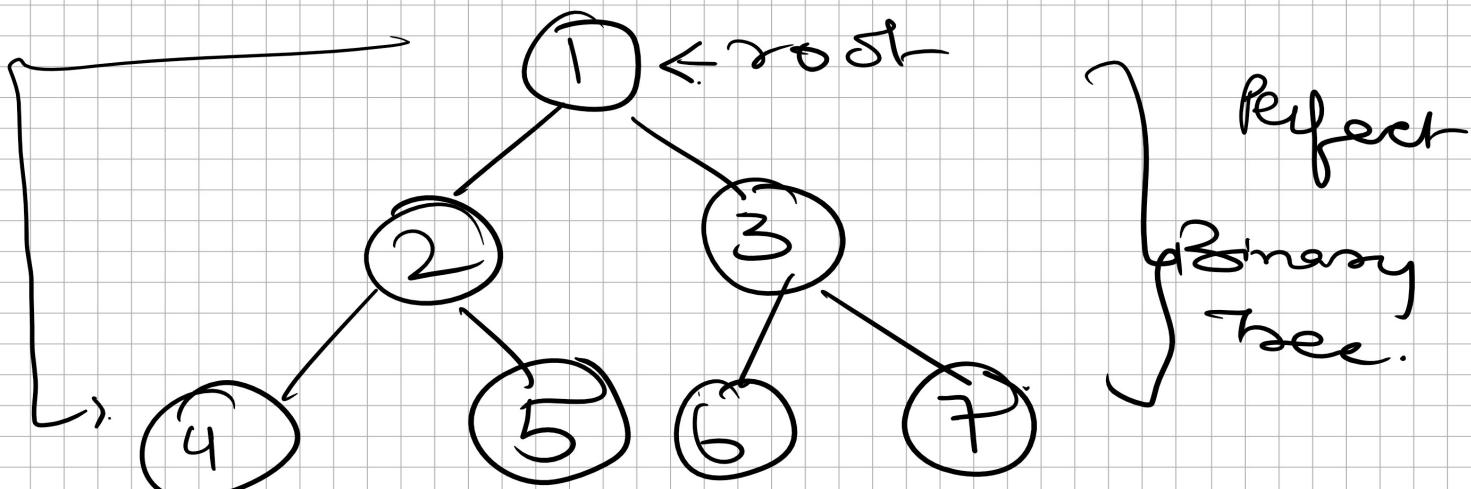
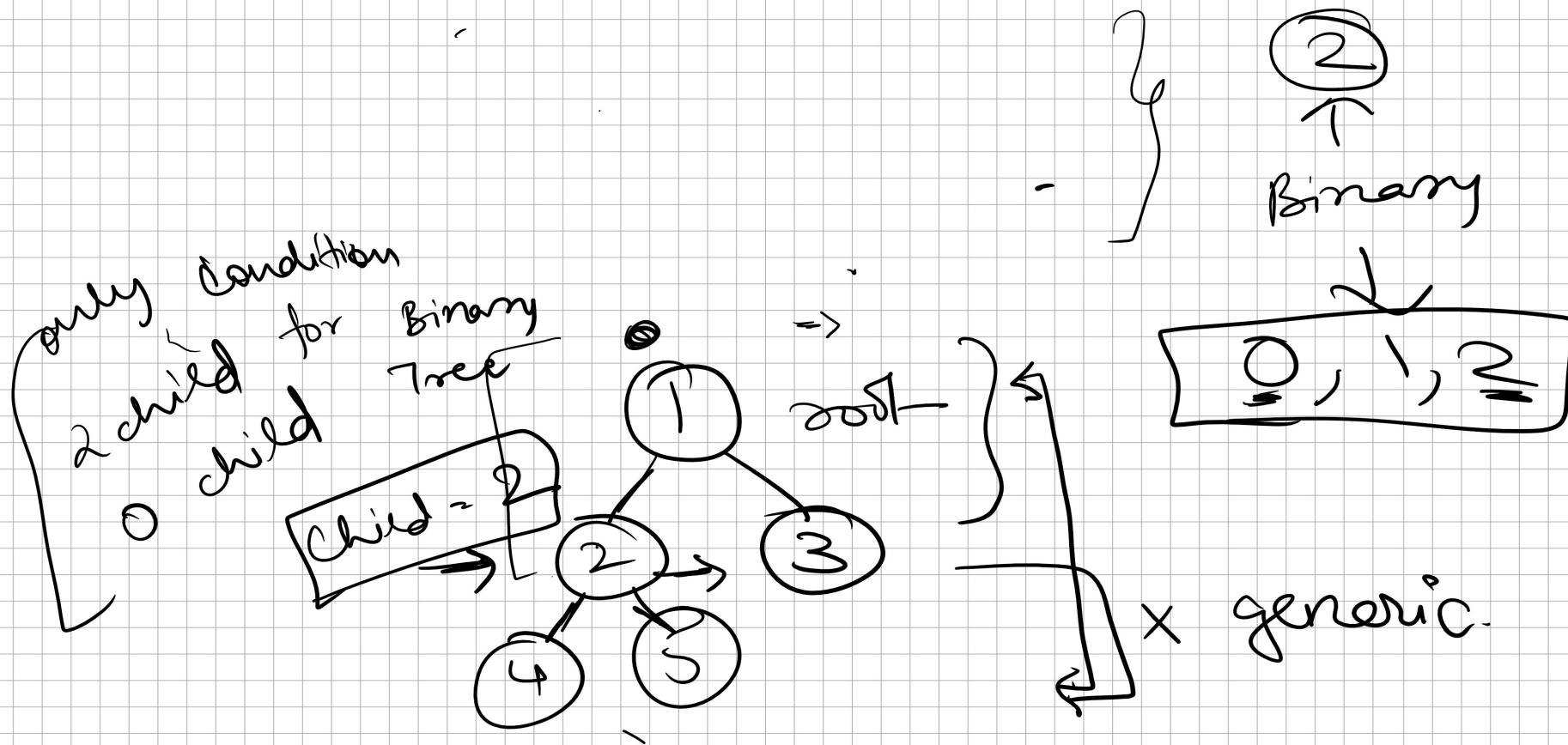


Linked List



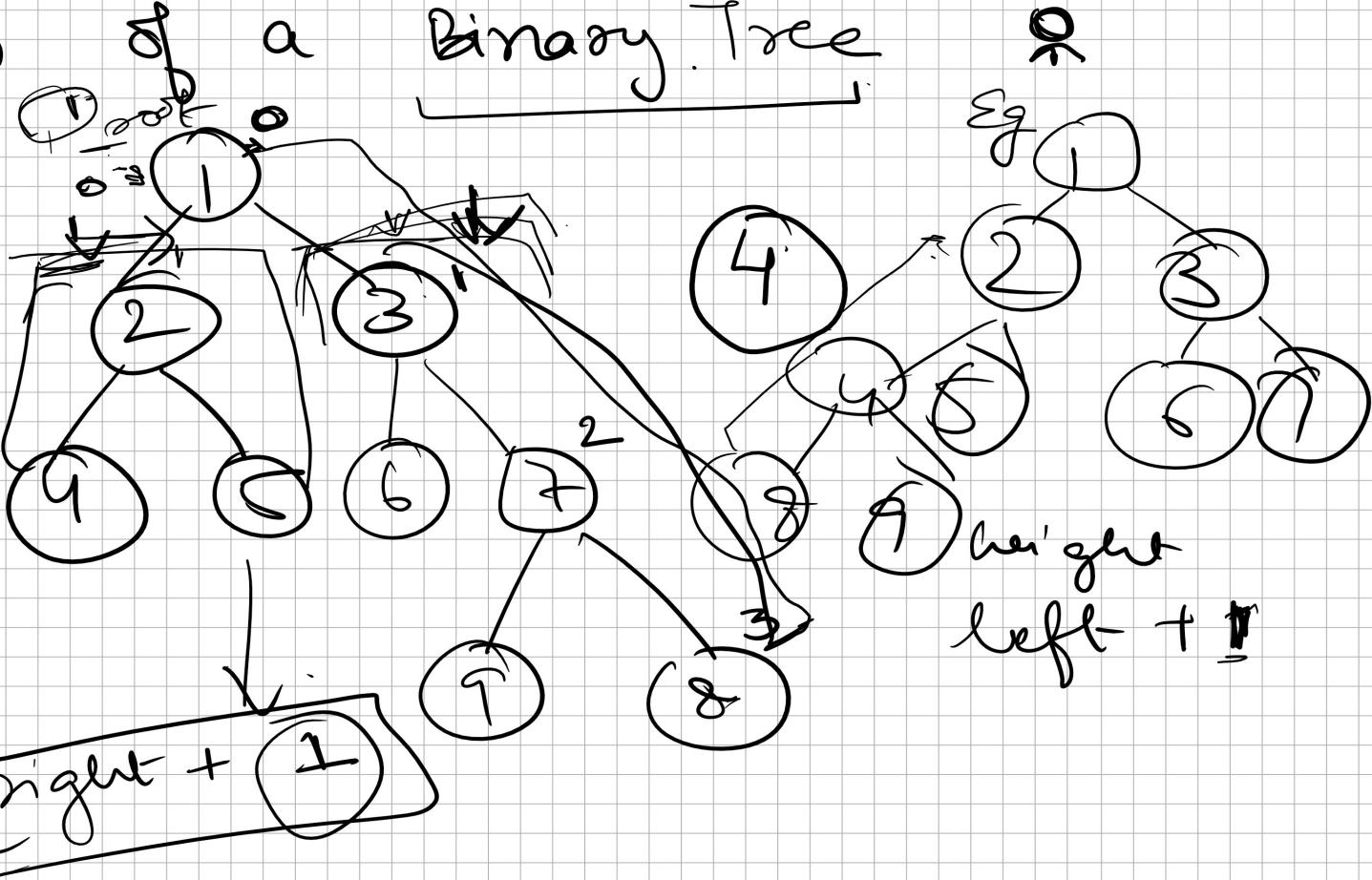
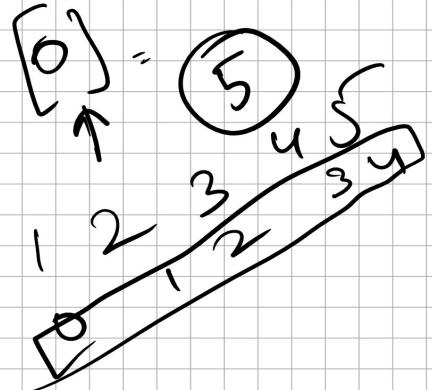
→ similar to linked list but instead of each node pointing to next node in a linear manner, each node points to several other nodes.





Max depth of a Binary Tree

arr. length



maxDepth (root):

{
1. 2 base condition check

2. leftHeight = maxDepth (left)

rightHeight = maxDepth (right)

root . left
root . right

base condition [recursion root]

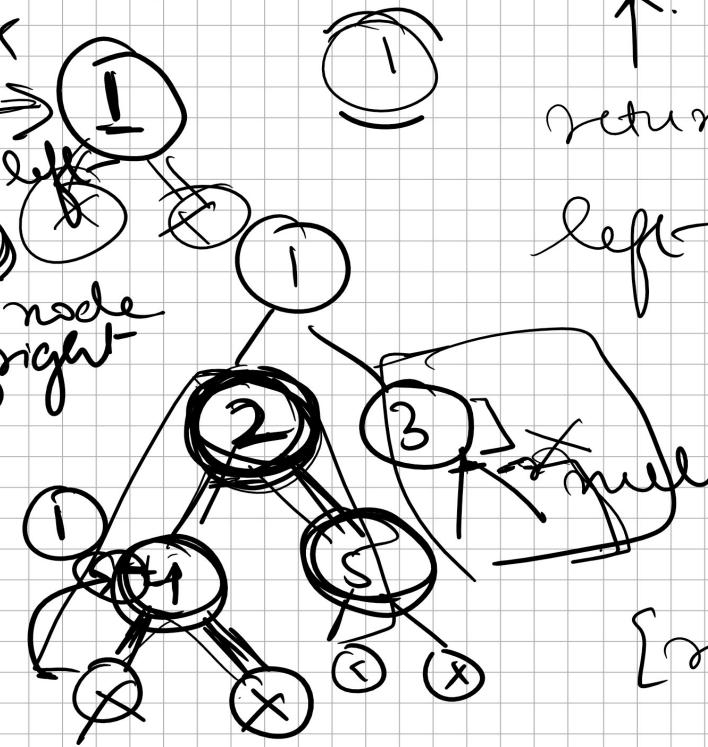
↑
return 0;

left

right

root = null

\$ root
= null
—————
[return 0]



if (leftHeight > rightHeight)

 return leftHeight + 1 ;

else return rightHeight + 1 ;