



## Agenda

- ① Diameter of tree
- ② is Balanced BT
- ③ level order traversal
- ④ left view
- ⑤ right view
- ⑥ zigzag traversal

# Diameter of a Binary Tree

- Max<sup>m</sup> dist. b/w any two leaf Nodes

$$\{40, 80\} \rightarrow 4$$

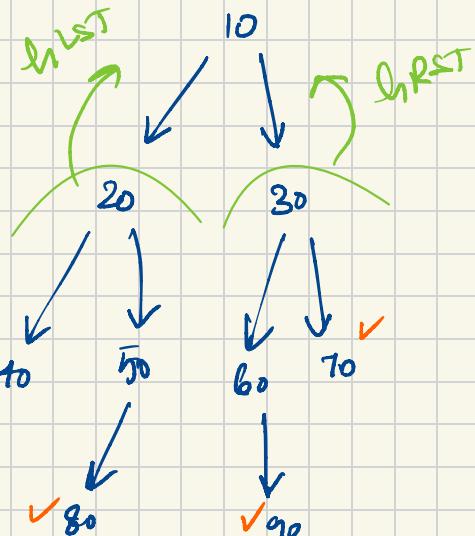
$$\{40, 90\} \rightarrow 6$$

$$\{40, 70\} \rightarrow 5$$

$$\{80, 90\} \rightarrow 7 \checkmark$$

$$\{80, 70\} \rightarrow 6 \quad \checkmark \text{ to}$$

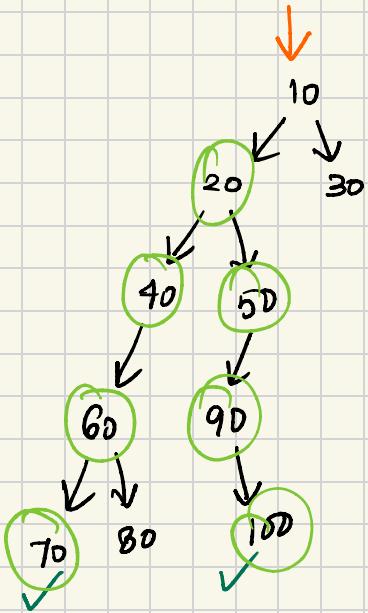
$$\{90, 70\} \rightarrow 4$$



diameter = 7 units

~~diameter = hLST + 1 + hRST~~

WRONG!



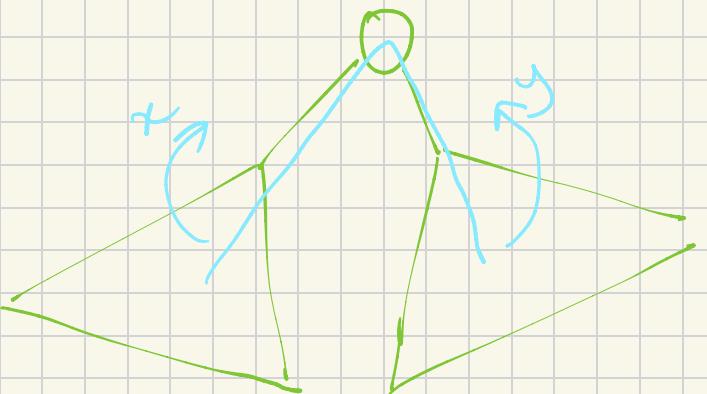
WHoNG!

$$\begin{aligned}
 \text{diameter} &= \cancel{\text{lhList} + \text{rhList}} \\
 &= 4 + 1 + 1 \\
 &\checkmark = 6
 \end{aligned}$$

{ NOTE: diameter doesn't pass through root.

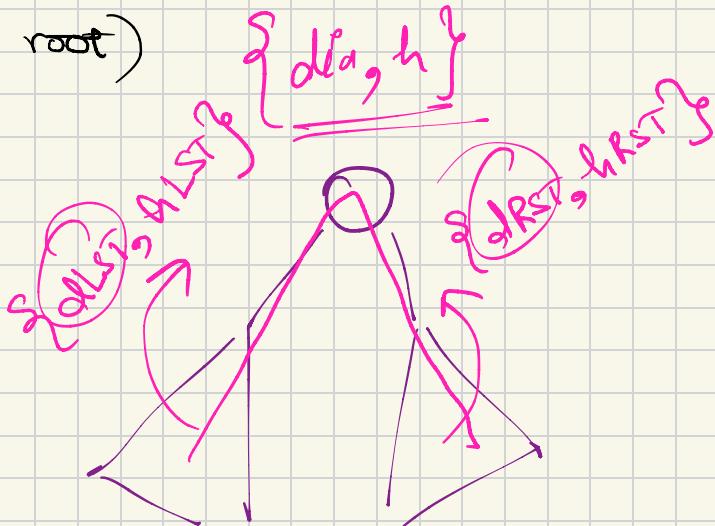
dirometer = Max { diameter through each node }

{  
o TC; SCN<sup>2</sup>)  
o SC; OCH)



form:  $\{ \text{dia}, \text{height} \}$  returns of a tree

Pair  
fun (Node root)  
{  
}



Class pair

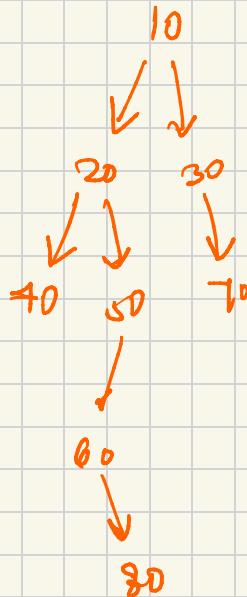
{ int dia;  
int height;

# Is Balanced Binary Tree

- Each node is balanced

Balanced Node

$$|\text{hLst} - \text{hRst}| \leq 1$$



boolean isBalanced (Node root)

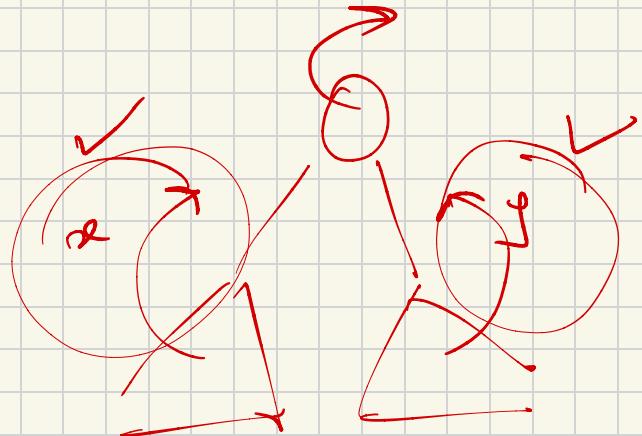
{

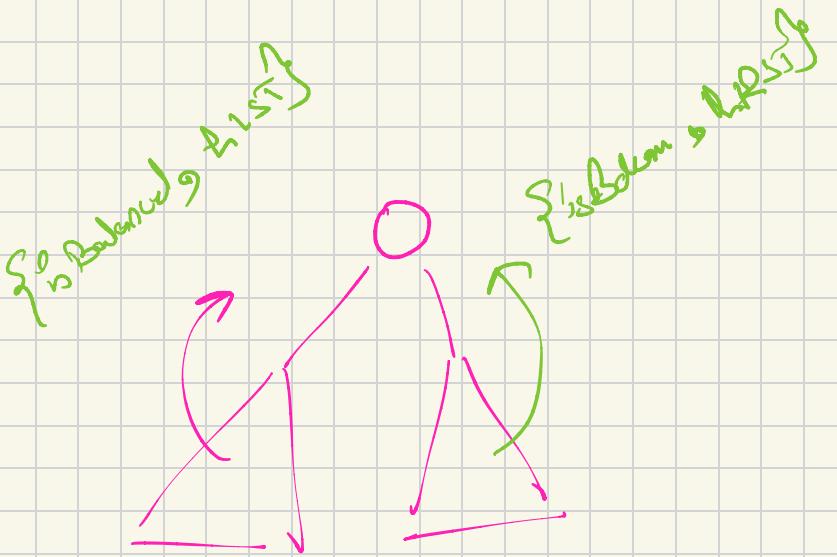
boolean  $\alpha =$  \_\_\_\_\_  
y  $\beta =$  \_\_\_\_\_

int hLST = height  
int hRST = height

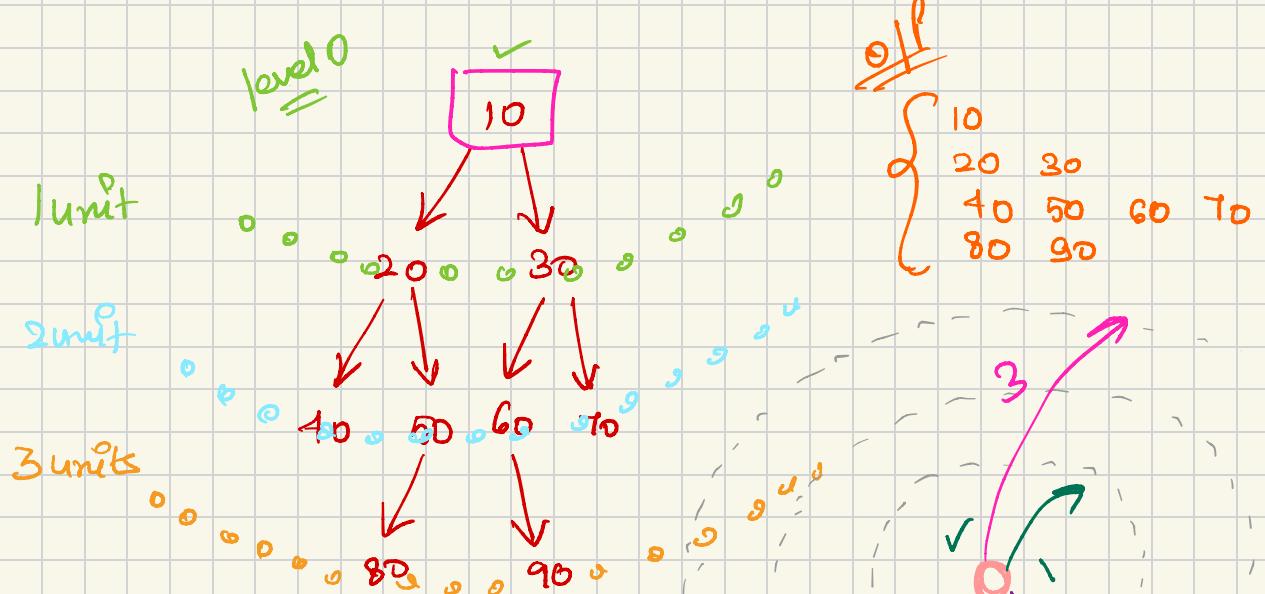
if ( $abs(hLST - hRST) \leq 1$ )

}



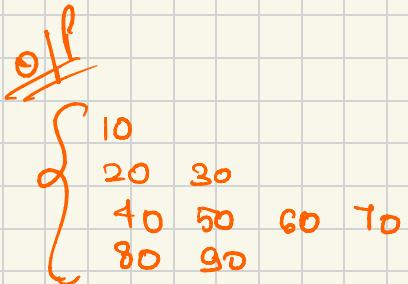


# Level Order Traversal { BFS }



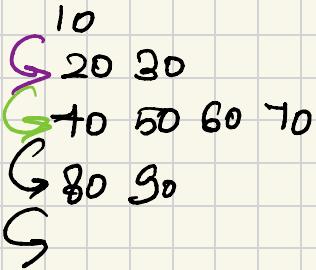
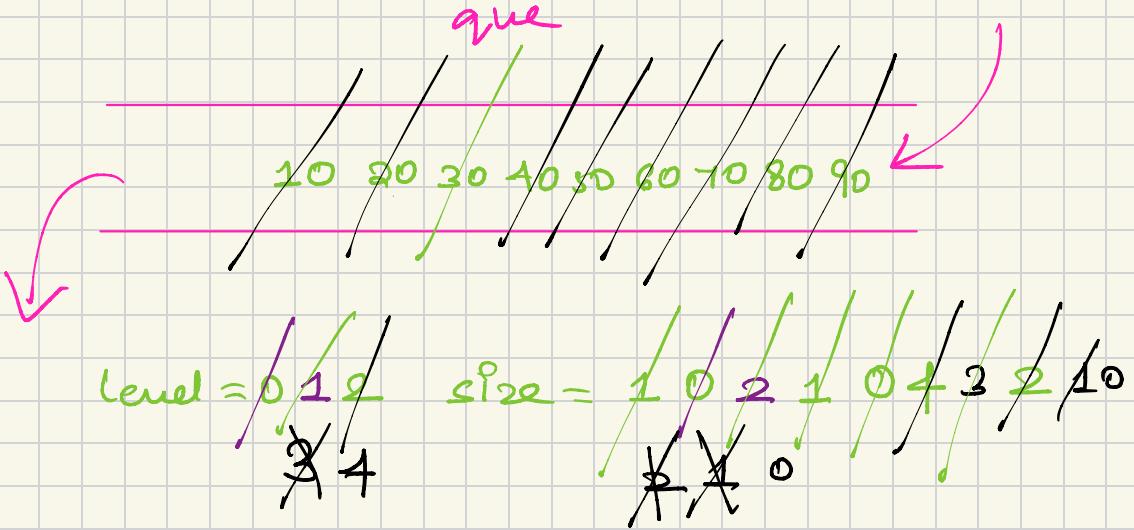
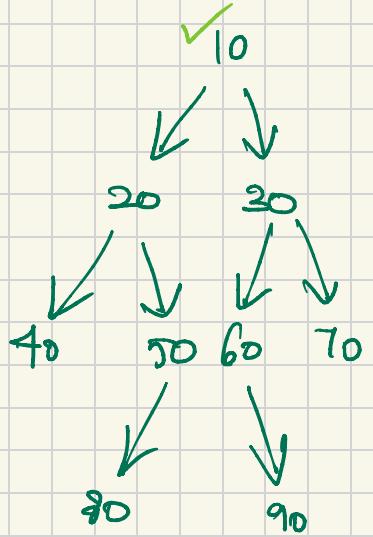
Over  
2

0 Breadth first Search!

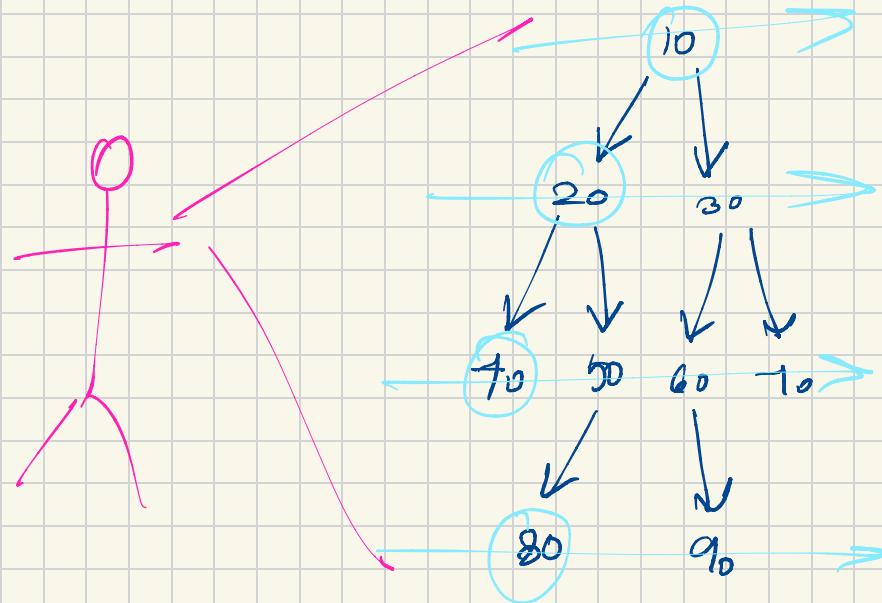


level 0  
level 1  
level 2  
level 3

moving  
radically!

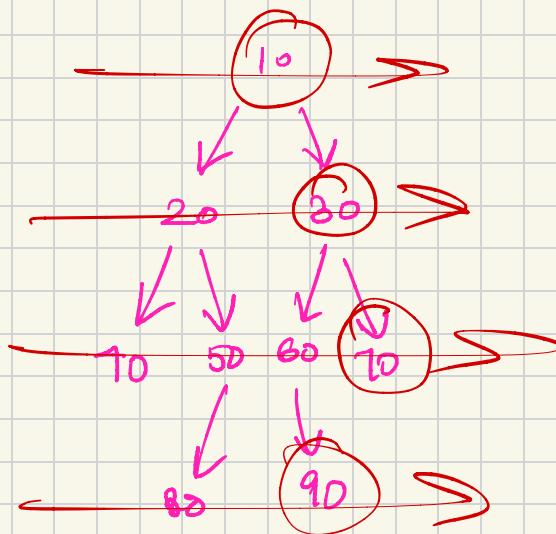


Left View



LV<sub>L</sub> { 10, 20, 40, 80 }

right view



10, 20, 10, 90

## Zigzag Traversal

