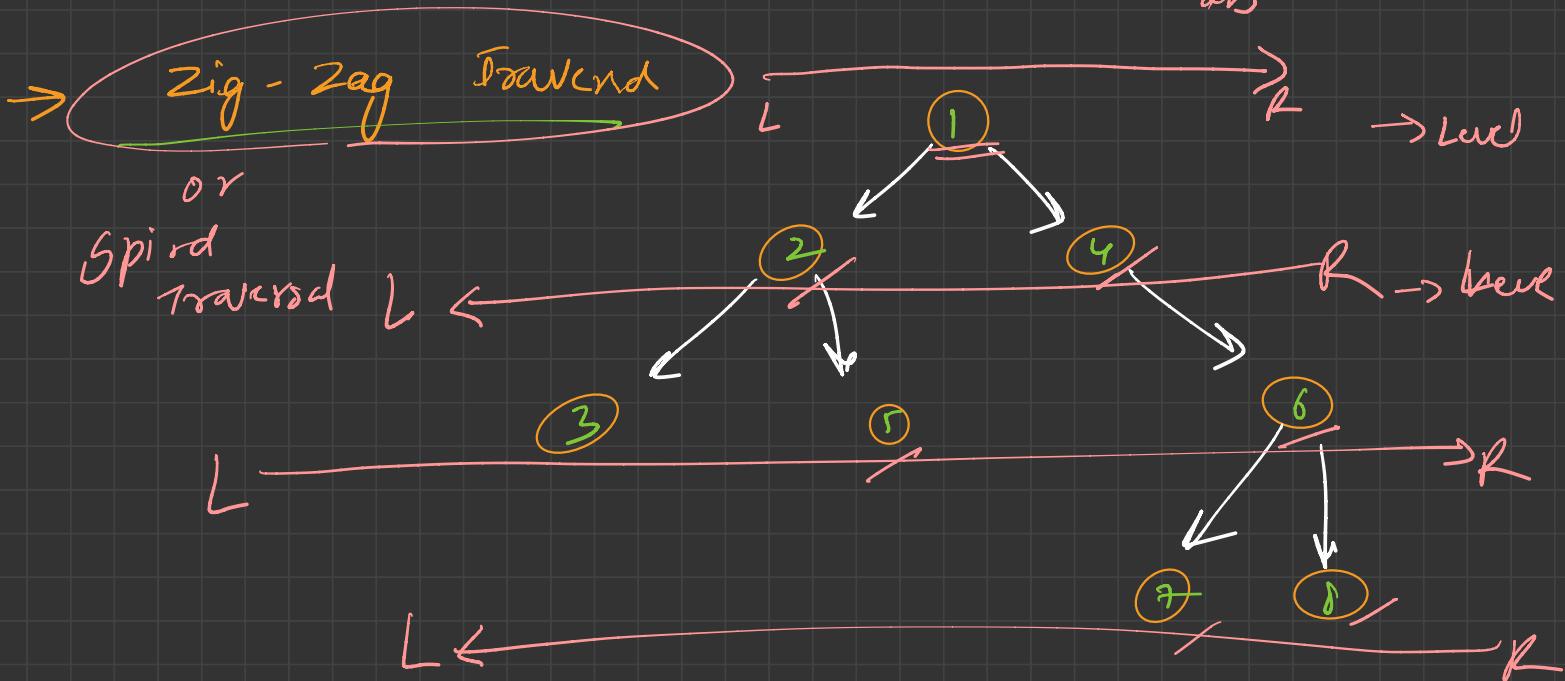
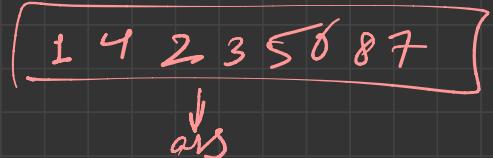
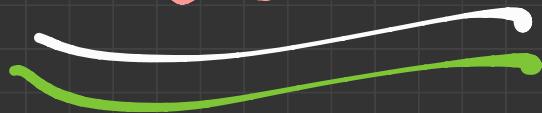
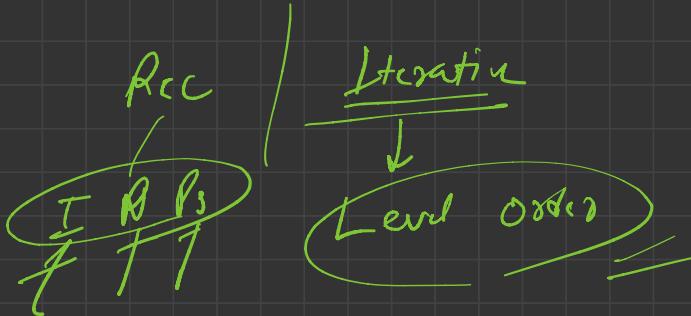
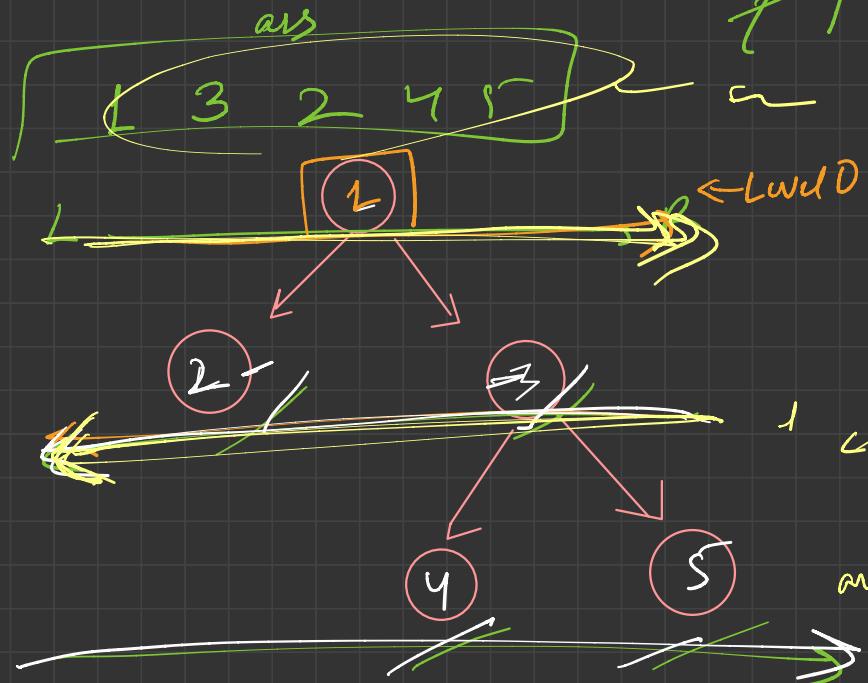




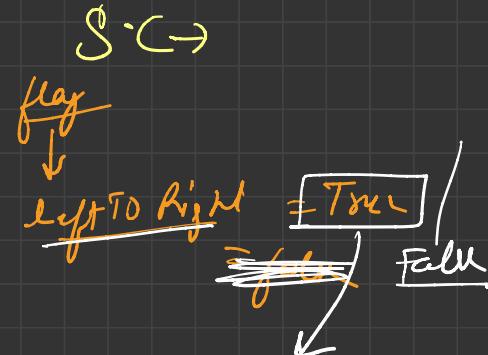

Trees



approach :-



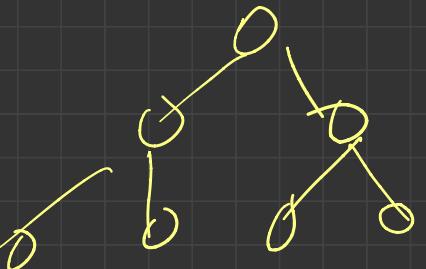
$$T.C \rightarrow O(n)$$



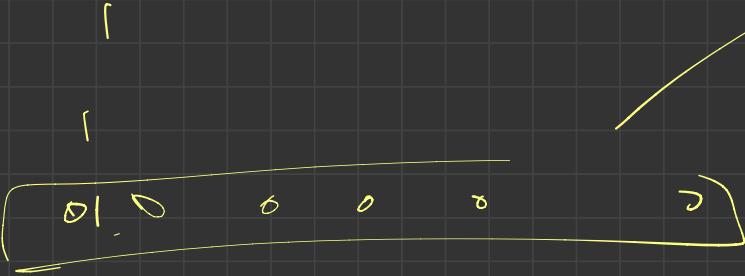
$$S.C \rightarrow$$

Left TO Right

Right TO Left



$S \subset \mathcal{O}(n)$



②

Boundary traversal

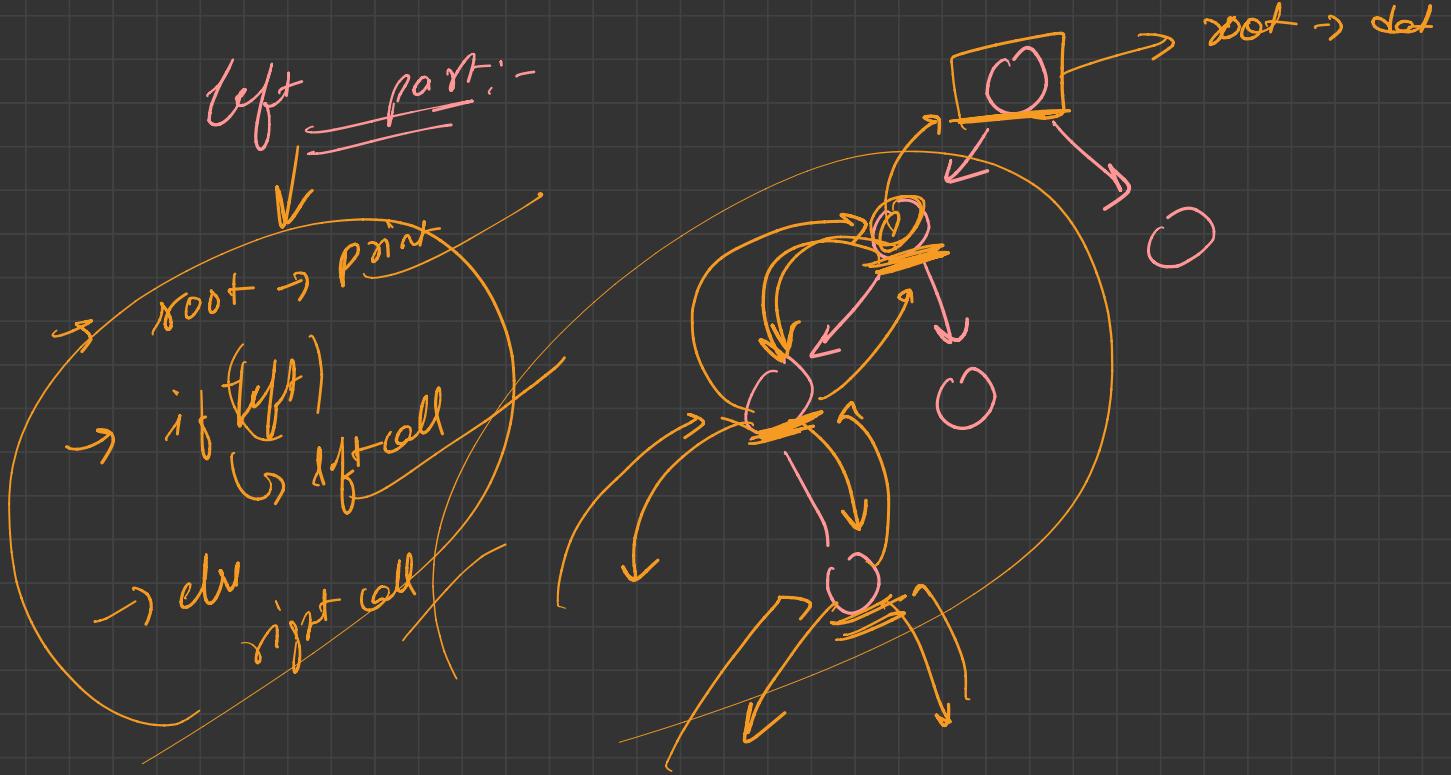


algo:-

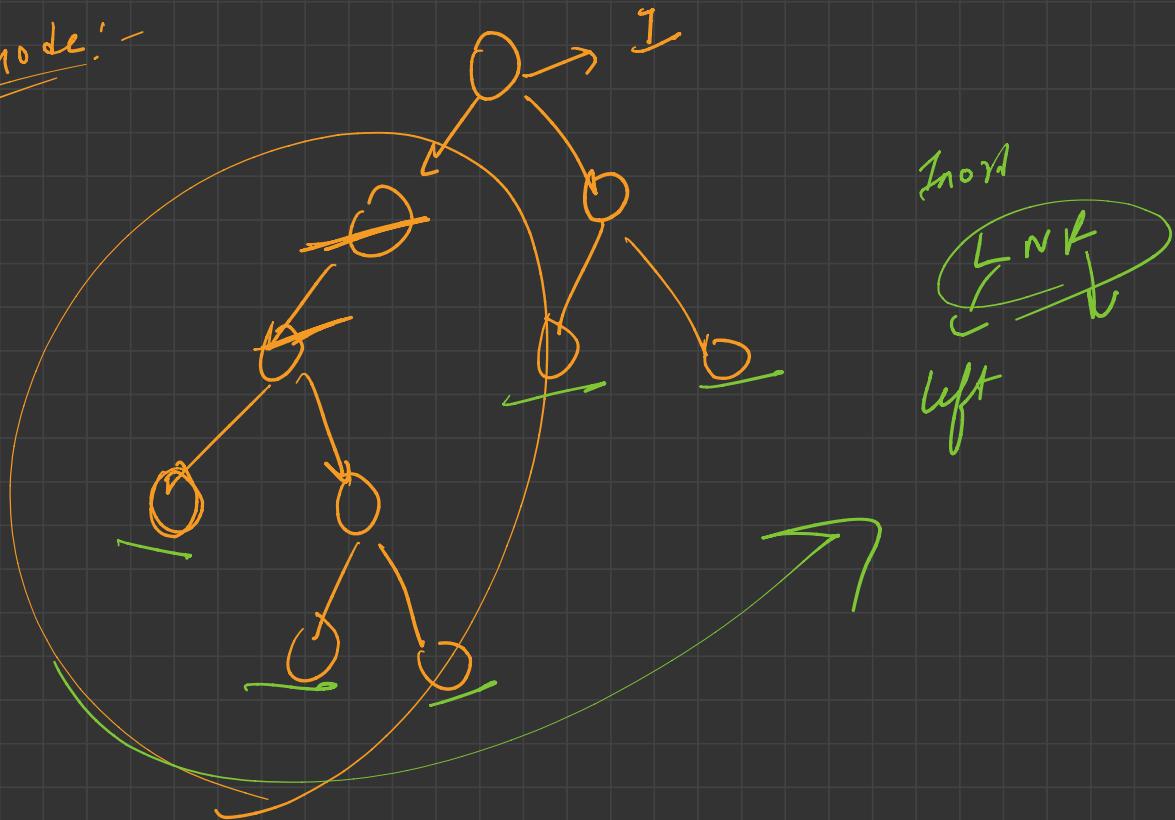
left - port → excl
Leaf node

Leaf node ($L \rightarrow f$)

right port
↓
(reverse cond)
excl
Leaf node



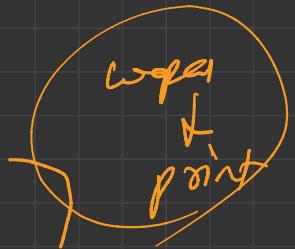
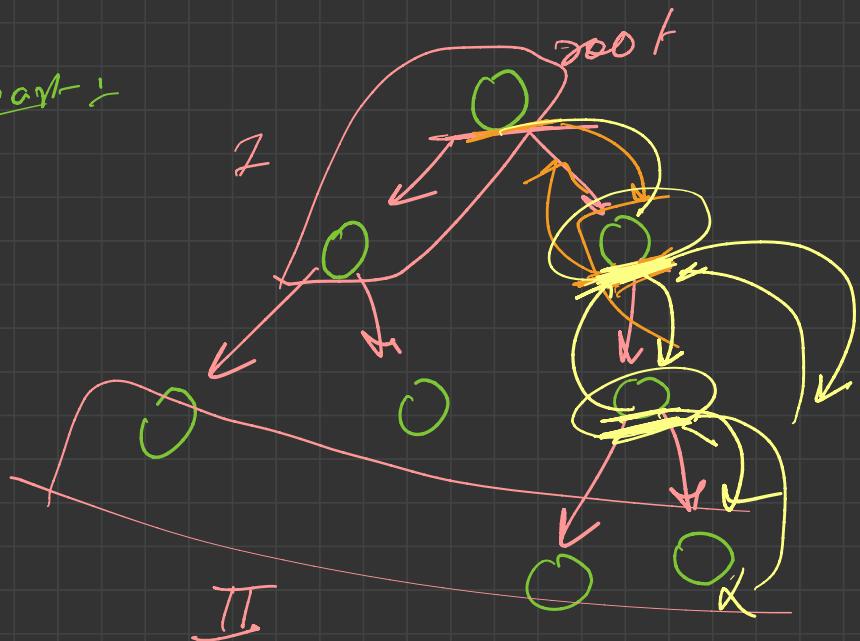
leaf node :-



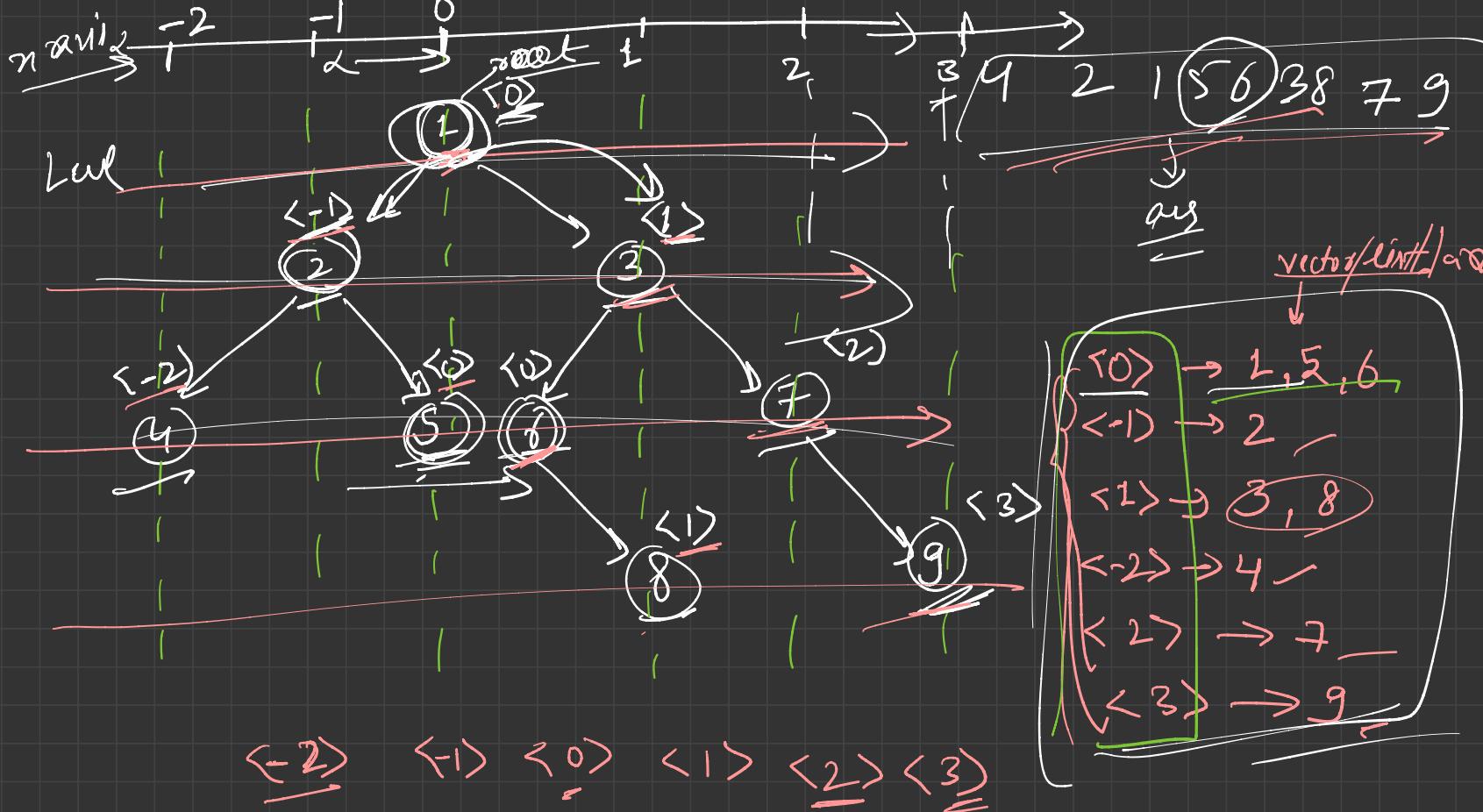
node

L N R
left

Right part :-

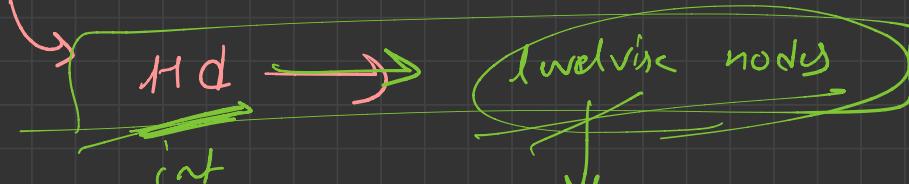


Vertical Order traversal



4, 2, 1, 5, 0, 3, 8, 7, 9

mapping



mapping → lvl → nodes

list < int >

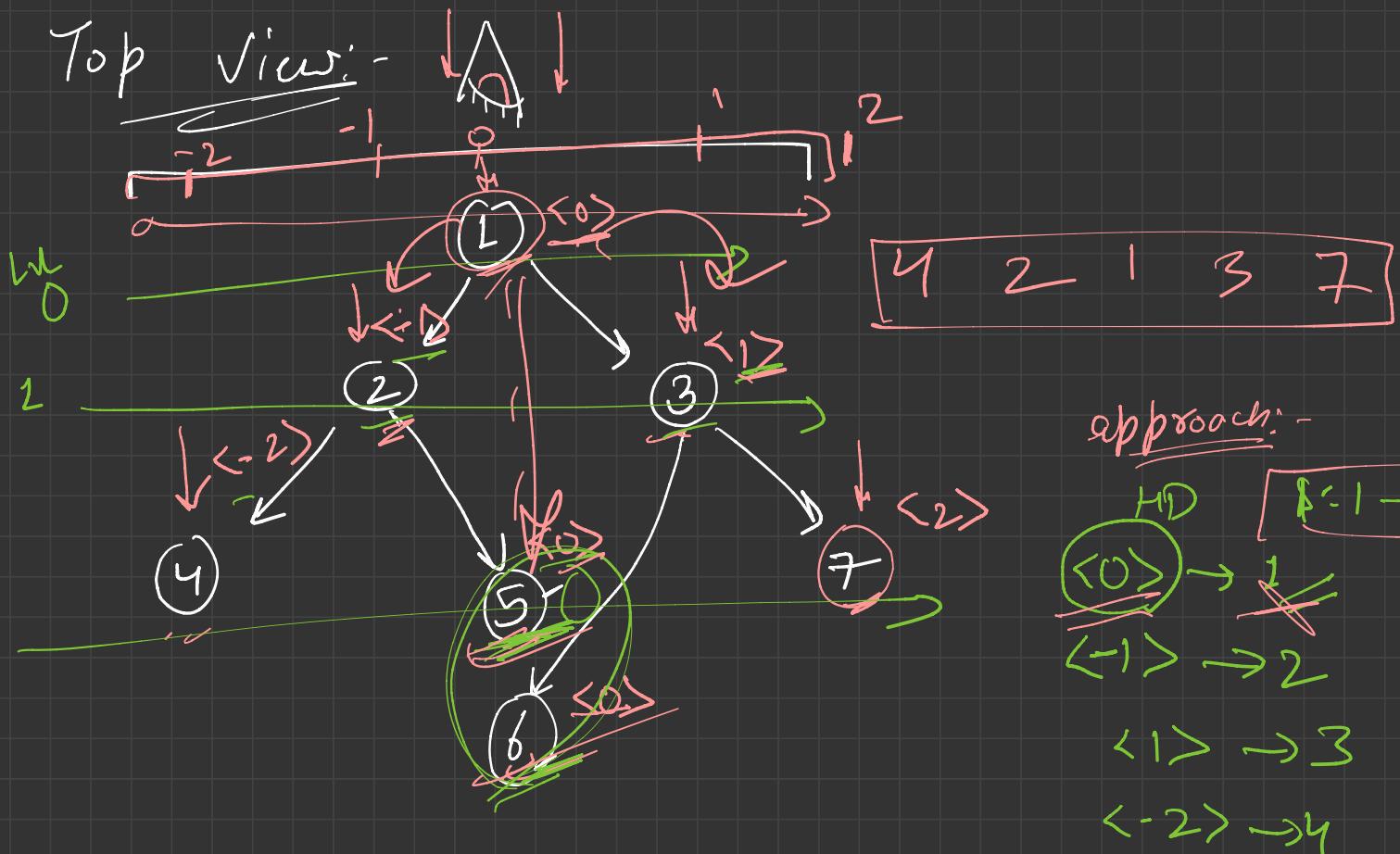
vector < int >

map < int, map < int, vector < int > > >

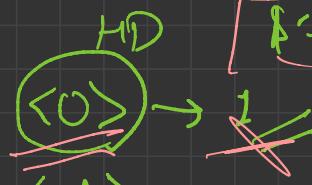
nodes

list of nodes

→ required → HD, Lvl
 → queue<pair<Node*, pair<int, int>>;
 ↓
 i → <int, mpl>
 i-first i-second
 j → <int, vector<int>>
 j-first j-second



approach:-



$\langle 1 \rangle \rightarrow 3$

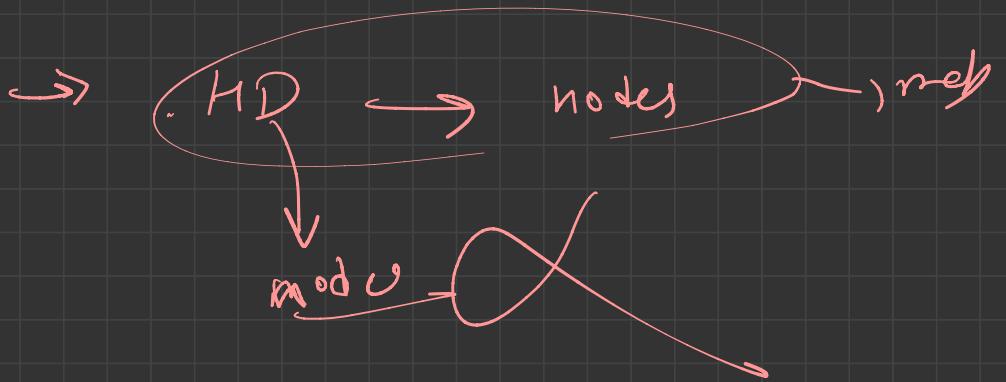
$\langle -2 \rangle \rightarrow 4$

map $\langle \frac{\text{int}}{4}, \text{int} \rangle m$

logic \rightarrow L.O.F map $\langle \text{int}, \text{int} \rangle \rightsquigarrow$

$\xrightarrow{\text{HD}}$ $\xrightarrow{\text{you} \rightarrow \text{num}}$

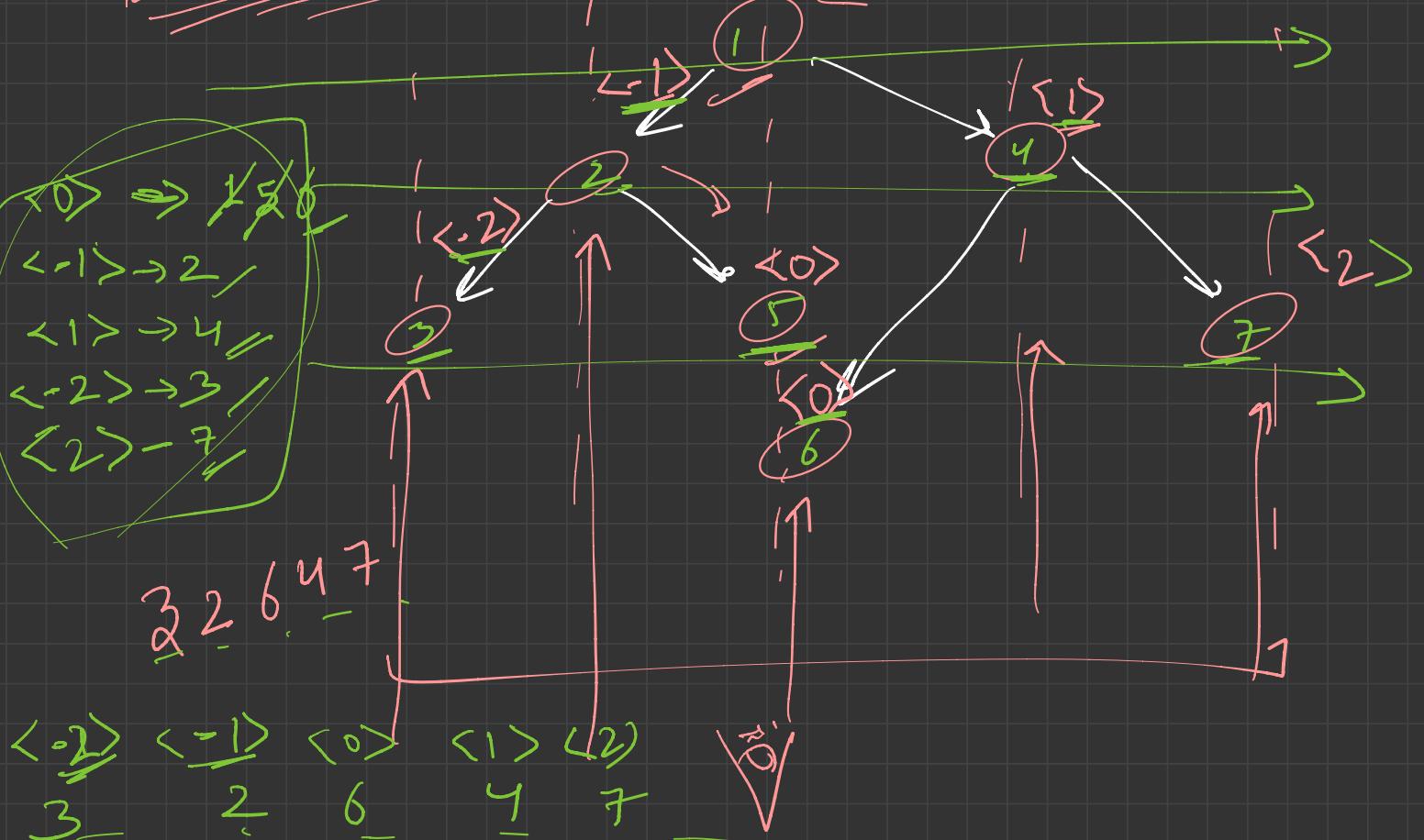
HD $\xrightarrow{\text{root} \rightarrow \text{dict}}$

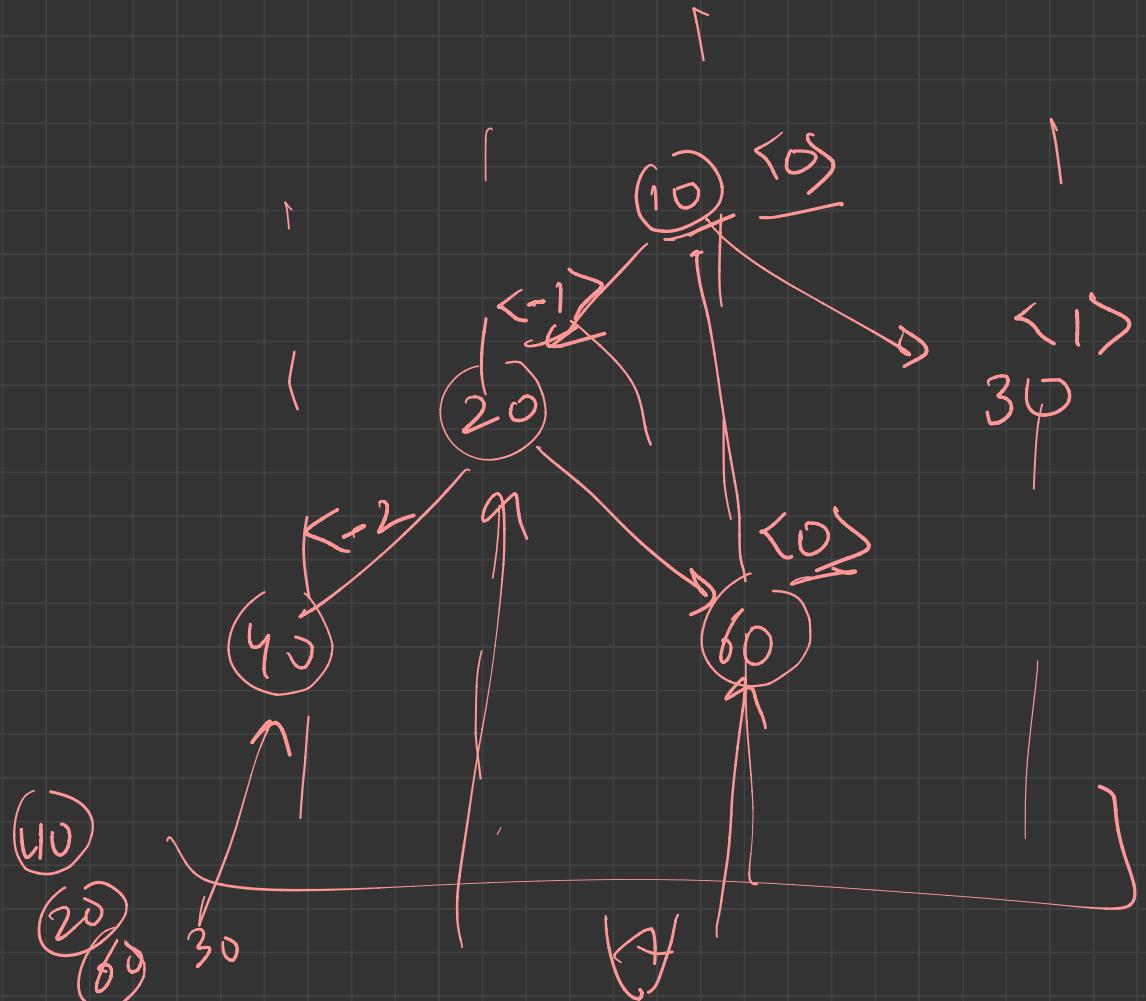


$i \rightarrow \langle \text{int}, \text{int} \rangle$

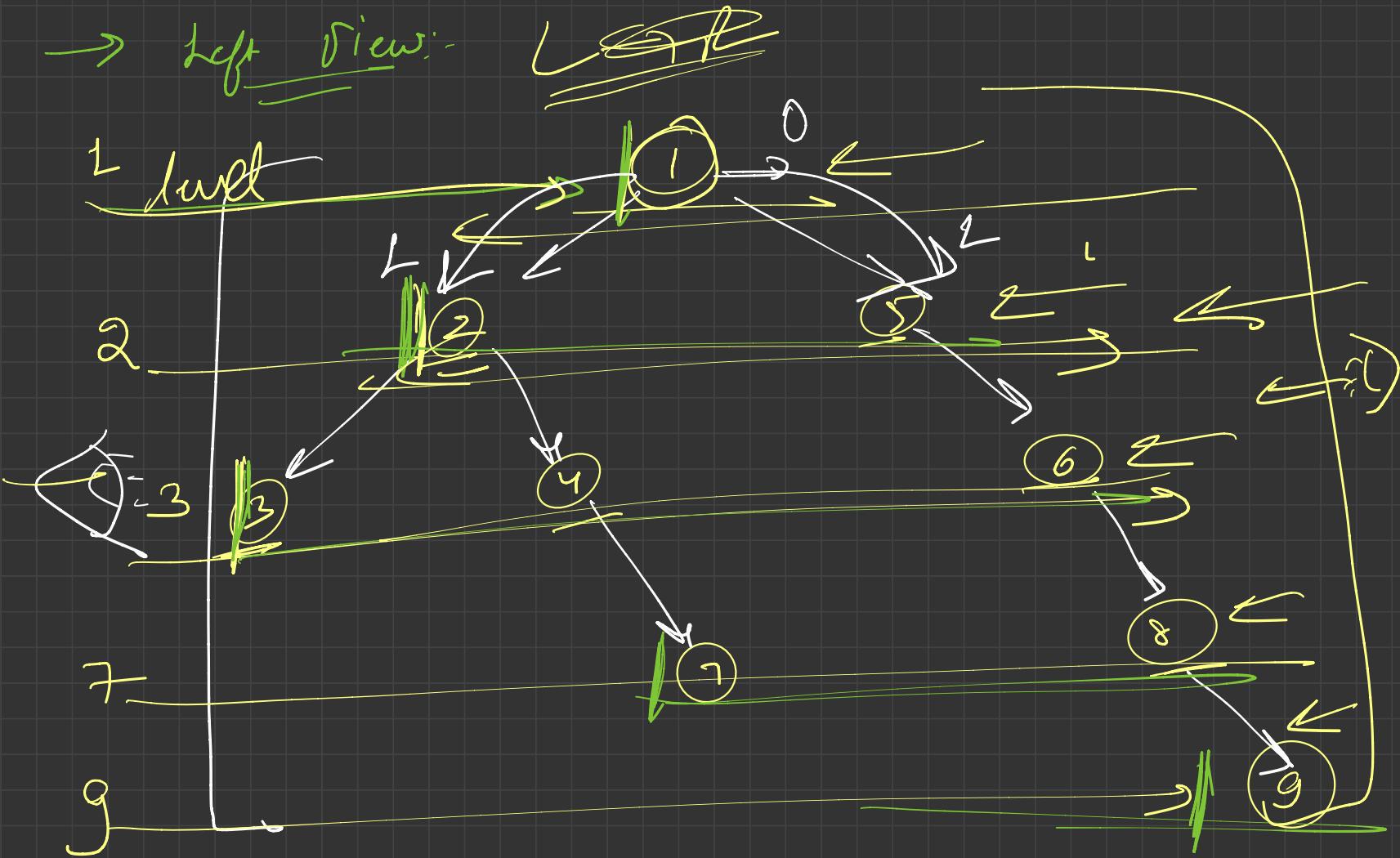
$\xrightarrow{\text{NI}}$ $\xrightarrow{\text{top} \underline{\text{nodes}}}$

Bottom View:

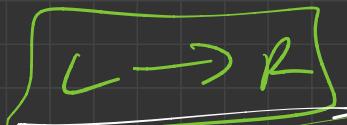




→ Left view:-



Approach :-



level → first node pair

L.O.T

Recursive

1st node → ?

level → stack
int level
next level

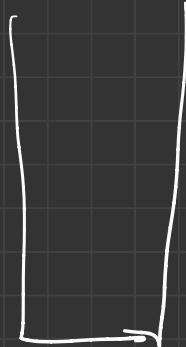
func (root, lvl)

{

// base case

if (root == NULL)

return;



*you entered
into a new
level*

if

(lvl == vector.size())

vector-store (root → level)

vector

fr(left, lvl + 1)

fr(right, lvl + 1)

{

Recursive
Tree

function-
call
stack

$H/\omega \rightarrow$ Diagonal - Transversal

