Plan for Week

Monday, 16 August 2021 10:32 PM

17th August 2021

1. Recover BST
2. Construct BST from level order
3. Serialise and Deserialise N-ary Tree
4. Serialise and Deserialise Binary Tree
5. left View of Binary Tree
6. Right View of Binary Tree

19th August 2021

1. width of Shadow of Binary Tree
2. Vertical order traversal of B-Tree
3. Vertical order traversal of B-Tree - II
4. Bottom View of a Binary Tree
5. Top View of a Binary Tree

21st August 2021 (Morning)

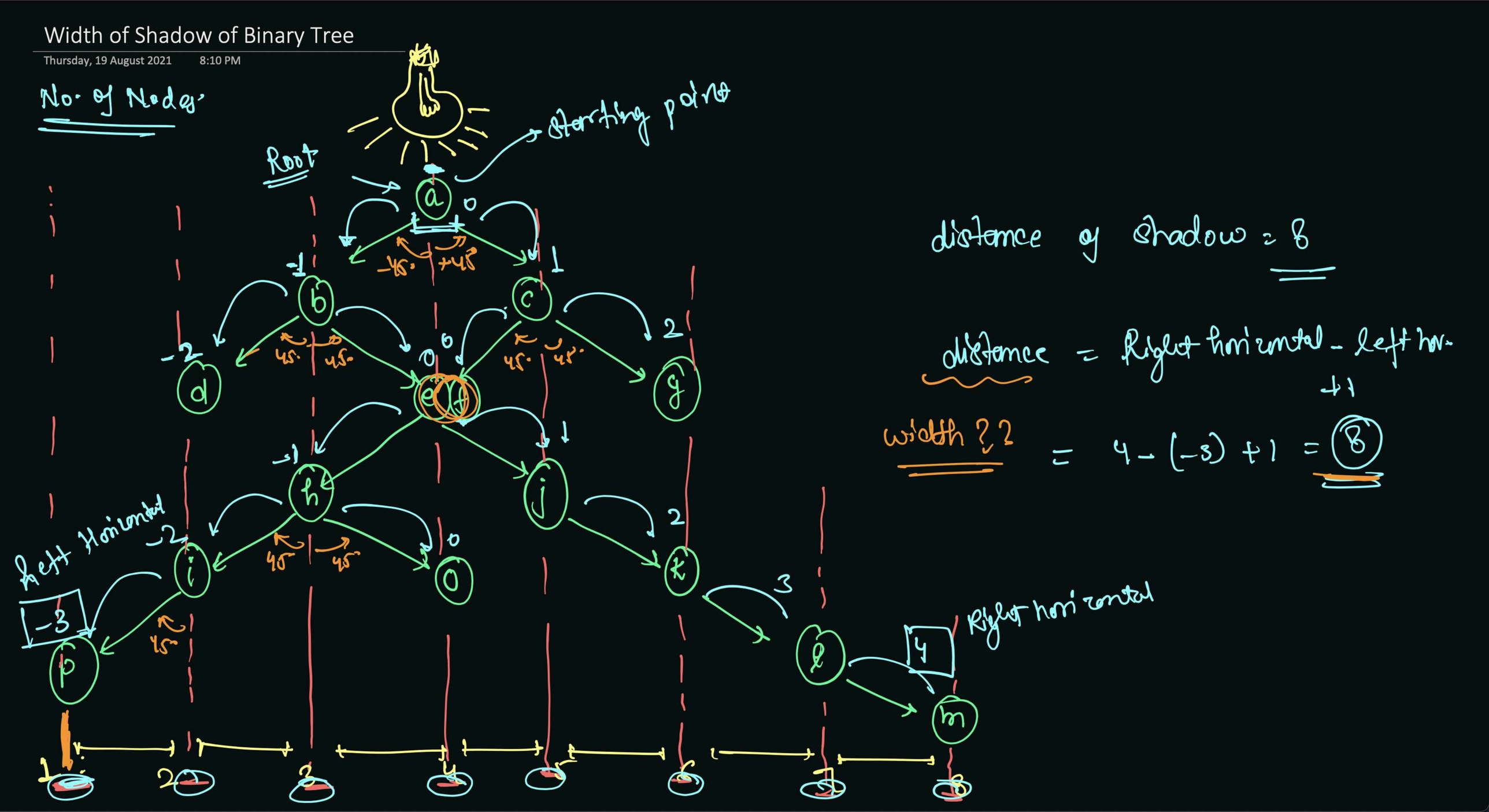
1. Diagonal order of a Binary Tree
2. Diagonal order of a Binary Tree (Anti-clockwise)
3. Vertical order sum of B Tree
4. Diagonal order sum of B-Tree
5. Node to root path in B-Tree.

21st August 2021 (Evening)

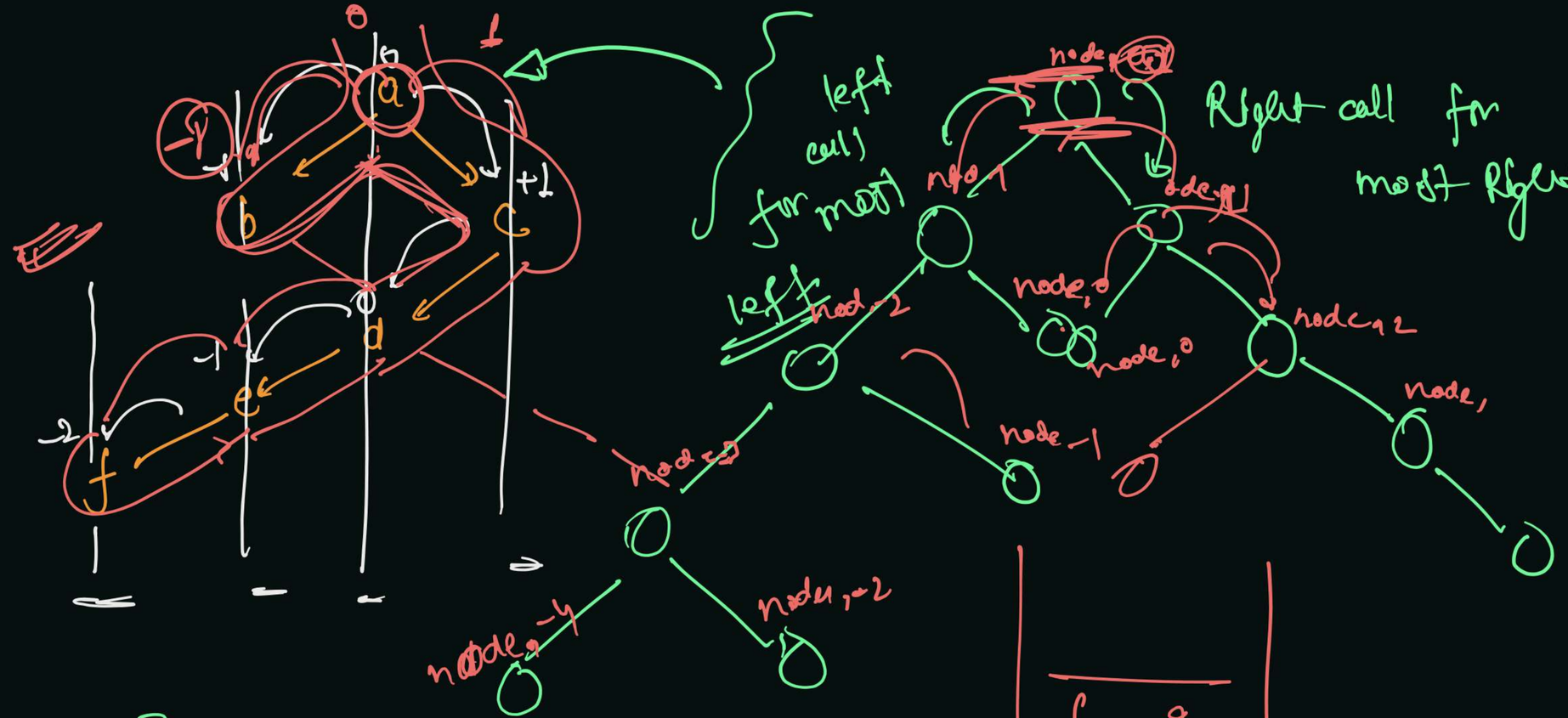
1. Iterator → Generic Tree from LL
   → BST Iterator - 2.
2. Inorder Morris Traversal
3. Pre Order Morris Traversal
4. Post order Morris Traversal

Width of Shadow of Binary Tree

Thursday, 19 August 2021 8:10 PM



lh / rh



rh = 1
lh = -2

width = rh - lh + 1
= 1 - (-2) + 1
= 4

minlh = ~~0~~ ~~-1~~ -2
maxrh = ~~0~~ 1

1 - (-2) + 1
1 = 4 quickly

-2

f_9, -2
e_9, -1
d_9, 0
c_9, 1
a_9, 0

Root, lh, rh

Stacky

Vertical Order Traversal Of Binary Tree

Thursday, 19 August 2021 8:31 PM

level wise

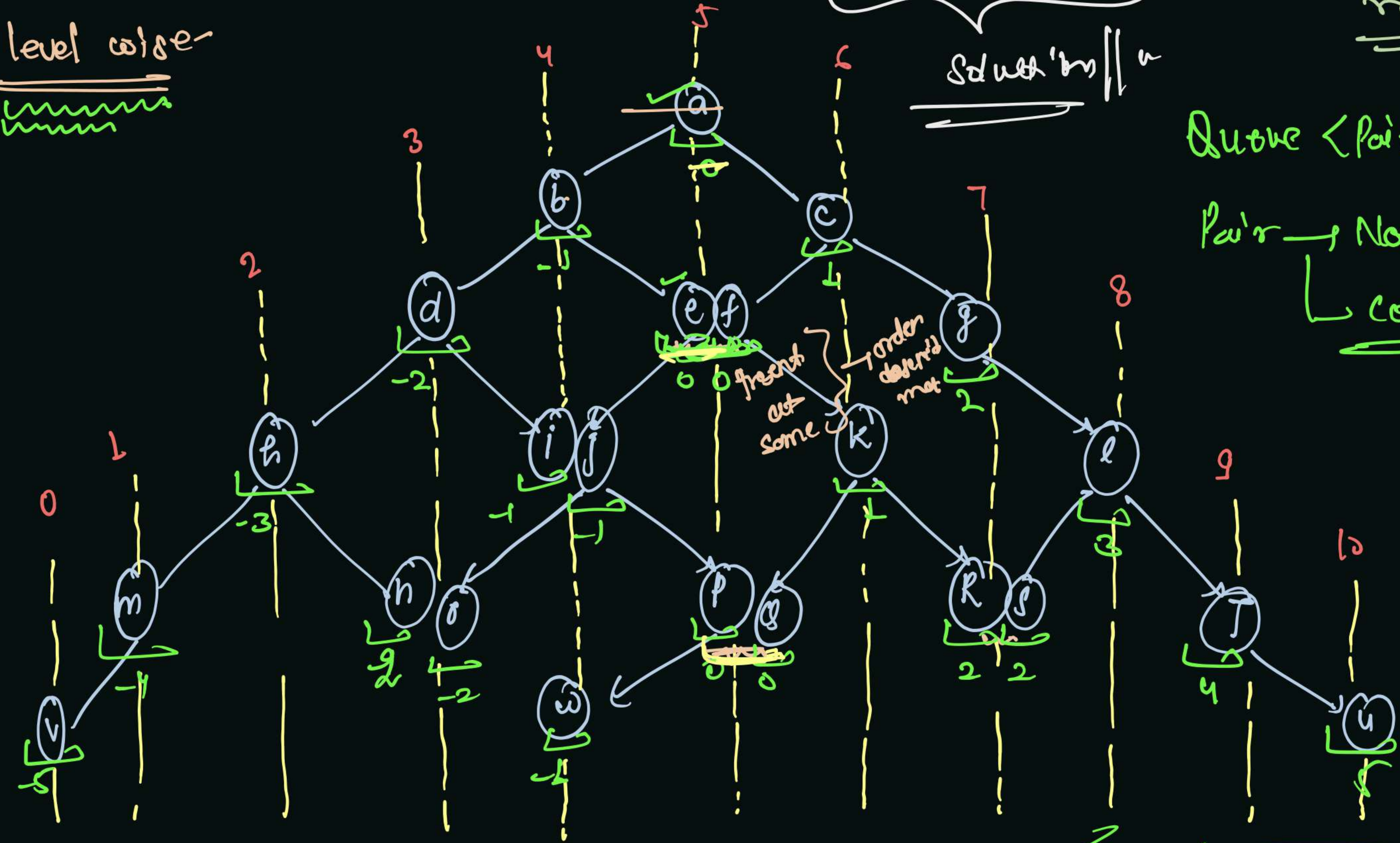
BFS + HashMap

List<List<Integer>>

why not DFS + HashMap

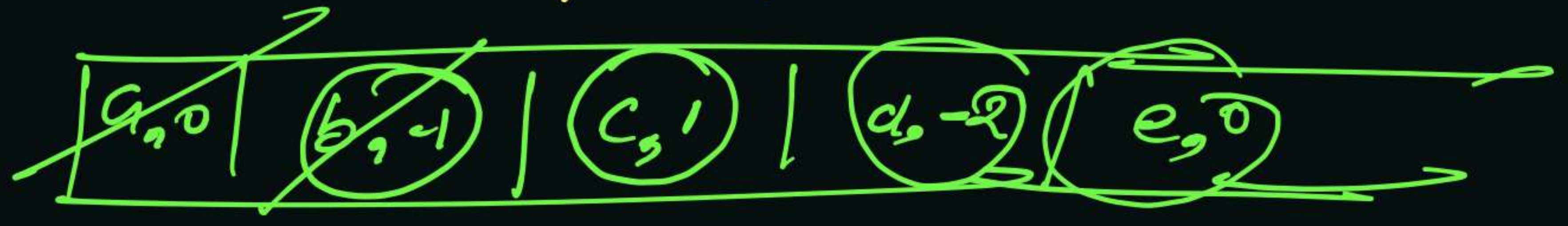
Queue <Pair>

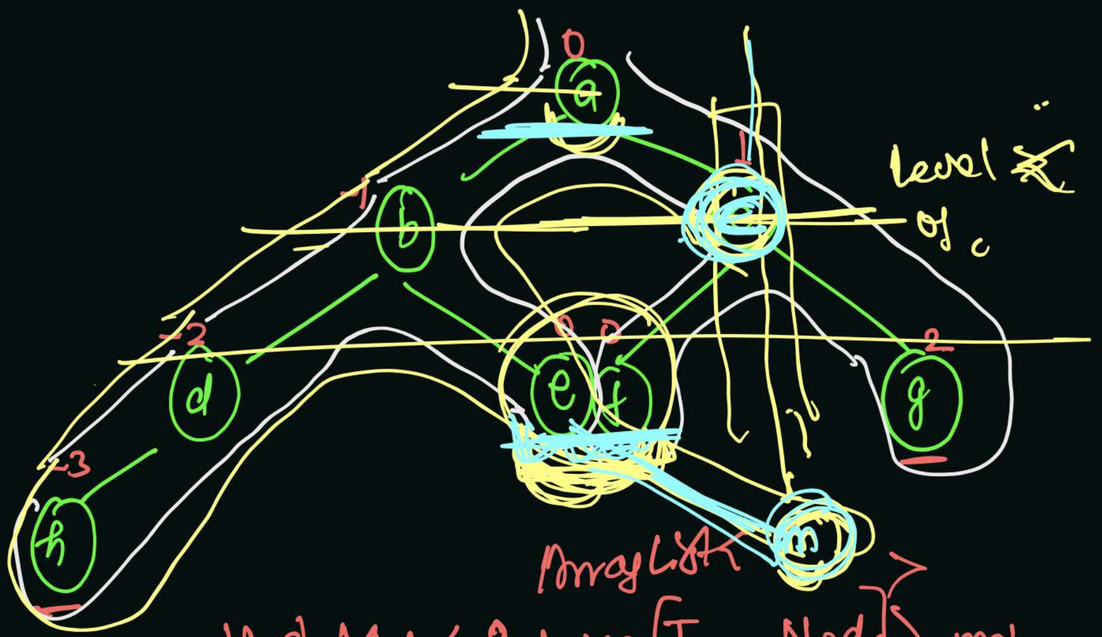
Pair → Node
└ count



0 → v
1 → m
2 → h
3 → d, n, o
4 → b, i, j, w
5 → a, e, f, p, q
6 → c, k
7 → g, R, S
8 → l
9 → t
10 → u

-1 → b
0 → a





level ≤ label of vertex of c

ArrayList

HashMap<Integer, [TreeNode]> map.

- 0 → a, e, f
- -1 → b
- -2 → d
- -3 → h
- 1 → c
- 2 → g.

-3 to 2

loop

List<List<Integer>>

Why Not DFS + Hash map -

e, f, a, g

0 → a, e, f

-1 → b.

-2 → d

-3 → h

1 → c, m

2 → g

requirement

0 → a, e

-1 → b

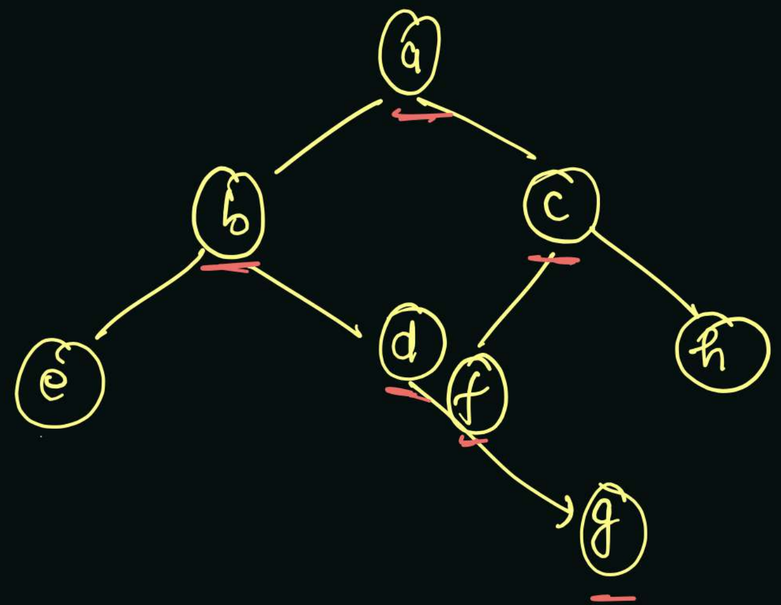
-2 → d

-3 → h

1 → m

wrong

BFS + HashMap



Get + Remove
works
add children

Pair → Node
comt

left horizontal = ~~0~~ ~~1~~ -2 +2 = 0
Right hori-zontal = ~~2~~ ~~4~~ 2 +2 4

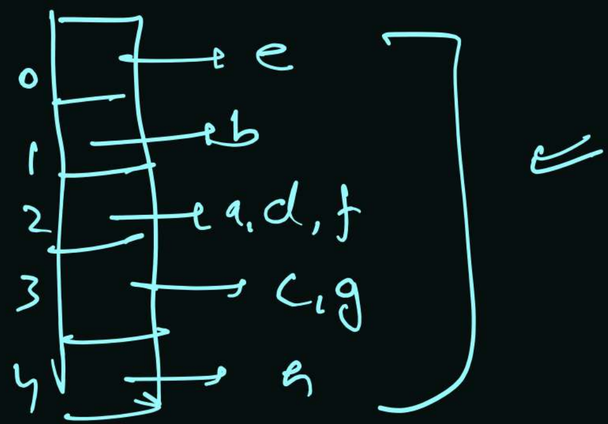
Queue of pair class

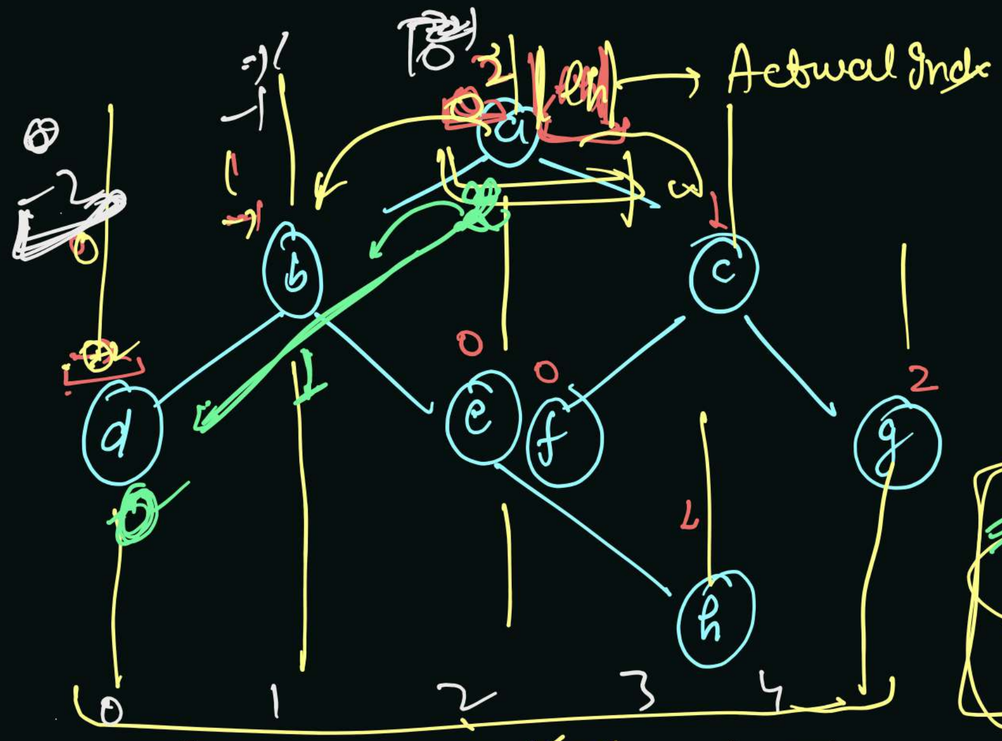
HashMap<Integer, AL<Integer>> map.

0 → a, d, f
-1 → b
1 → c, g
-2 → e
2 → h

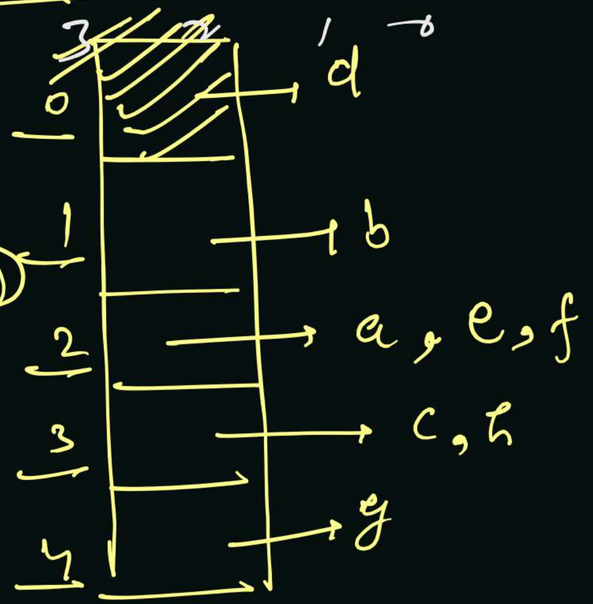
Make a loop from -2 to 2

2 - (-2) + 1
= 5





width = 5



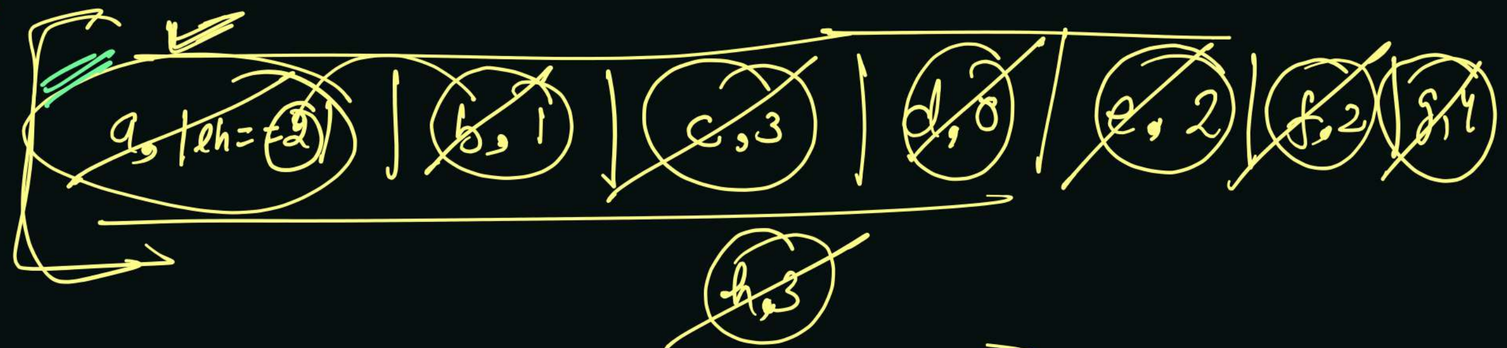
ArrayList<Array<Int>>

Step 1. Find width, width = 5

lh = -2

rh = +2

Step 2 Make array list of size 5



No need of HashMap

Vertical Order Traversal Of Binary Tree -II

Thursday, 19 August 2021 8:31 PM

2 - priority Queue.

Seond vertical order.

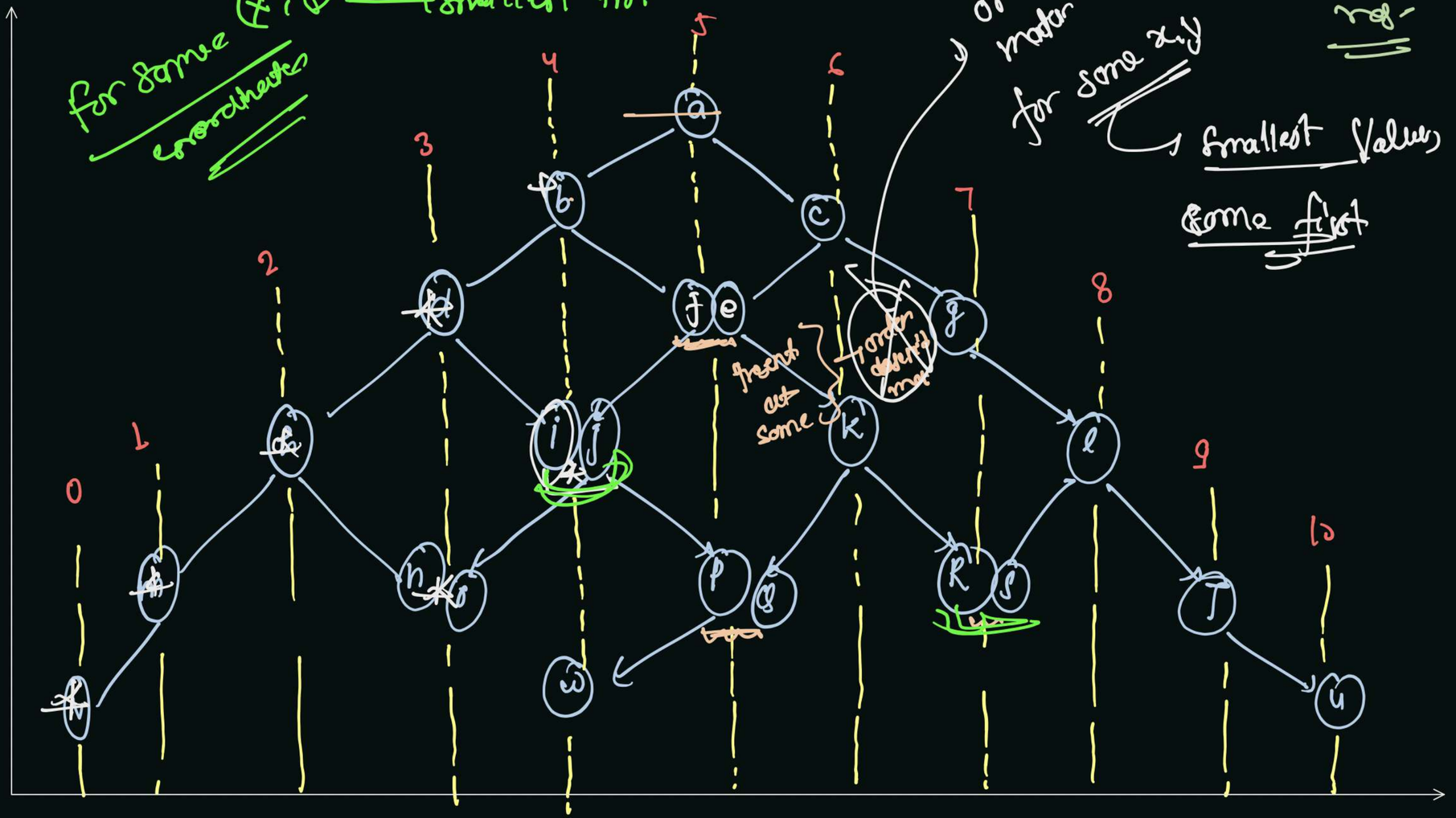
List<List<Integer>>

for same (x, y) → smallest first

for same coordinates

order matter for some x, y

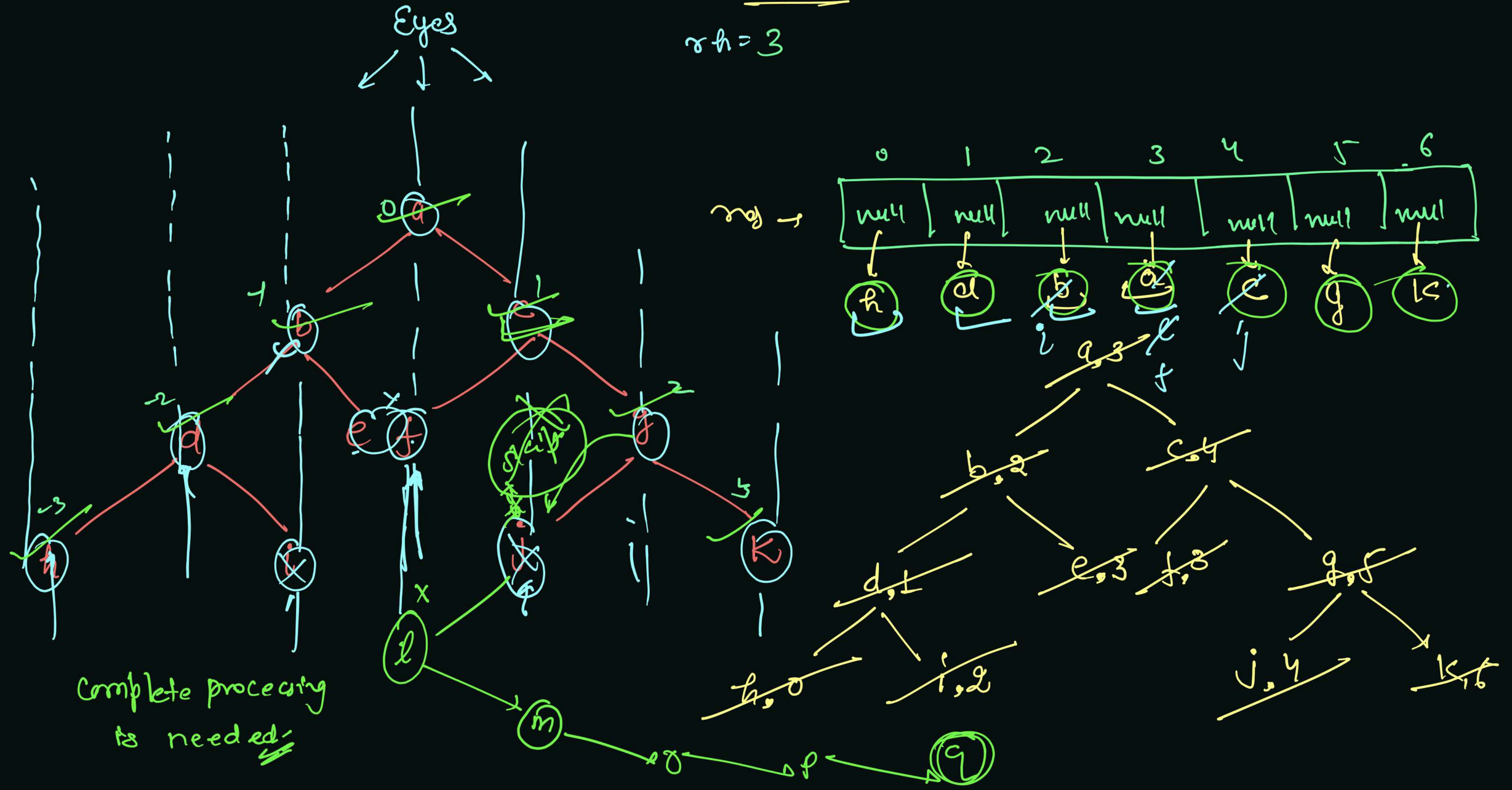
smallest Value come first



0 → v
1 → m
2 → h
3 → d, n, o
4 → b, i, j, w
5 → a, e, f, p, q
6 → c, k
7 → g, R, S
8 → l
9 → t
10 → u

Top View

Thursday, 19 August 2021 8:31 PM



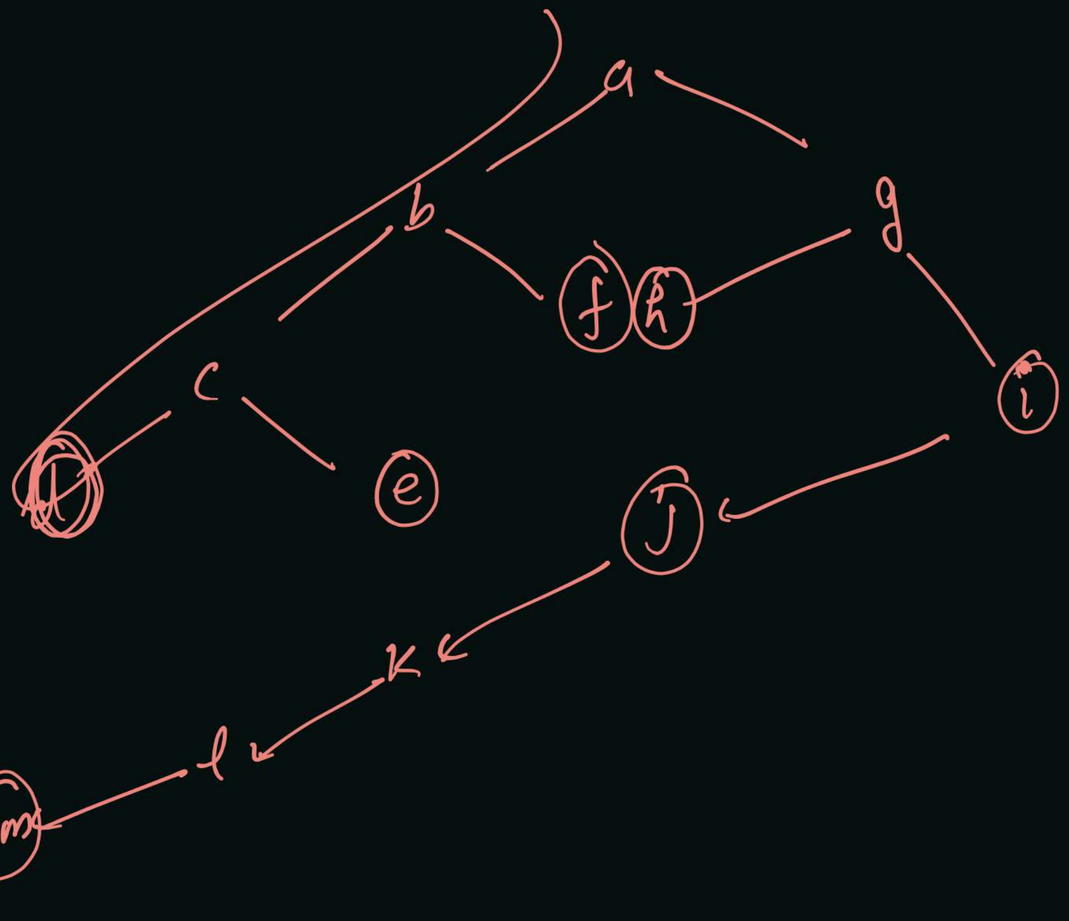
Bottom View

Thursday, 19 August 2021 8:31 PM

Same as top View

just comment if and

continue add Elements in arraylist



Not first in DFS

Example