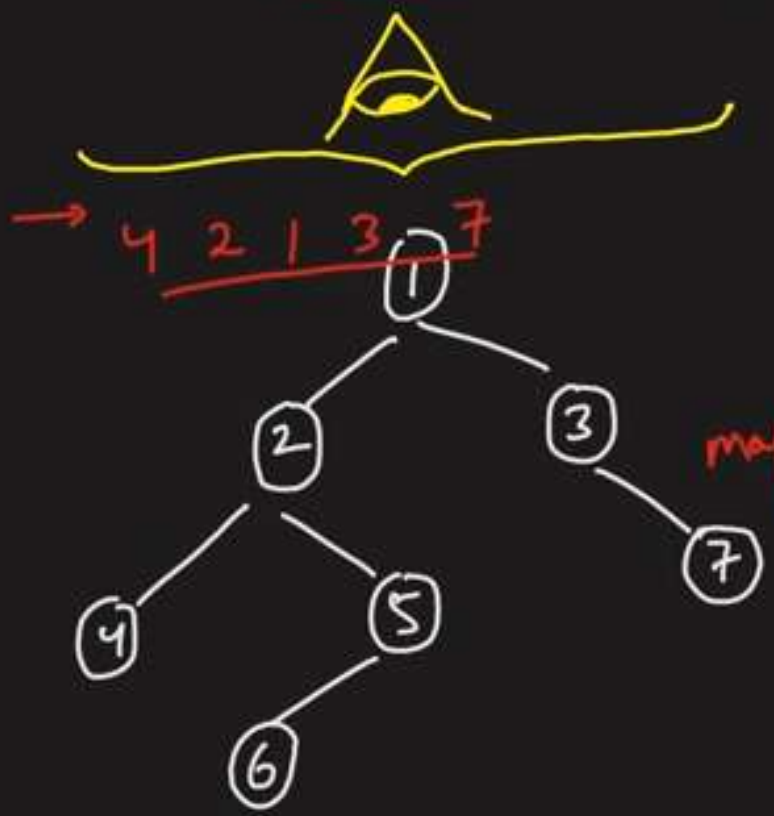
Top View of Binary Tree



map (line, node)

2 → 7
-2 → 4
1 → 3
-1 → 2
0 → 1

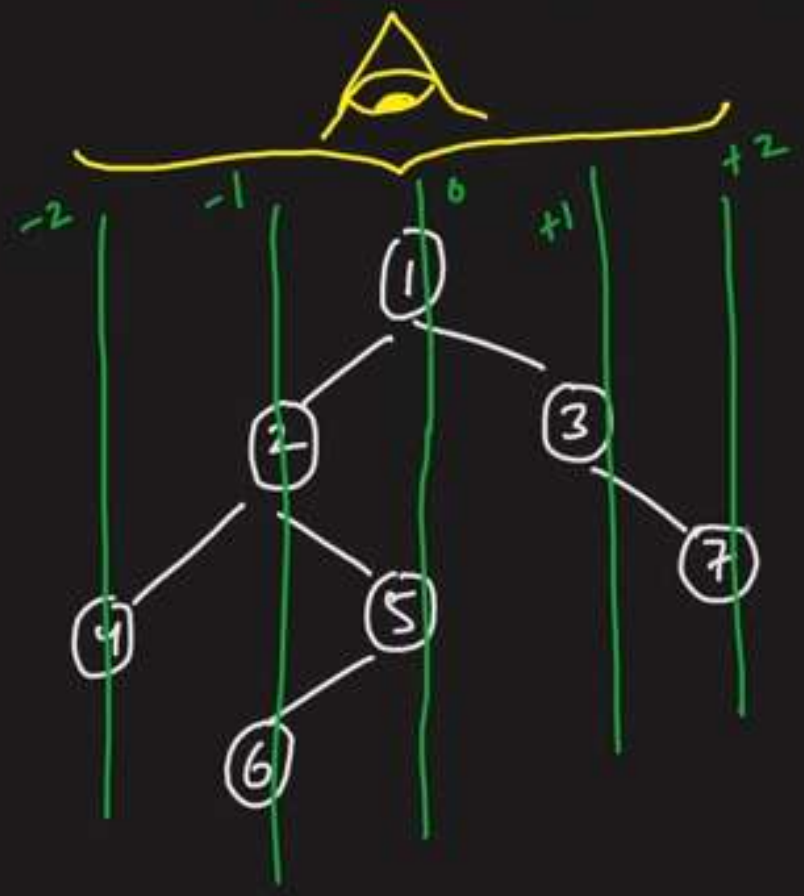
Q

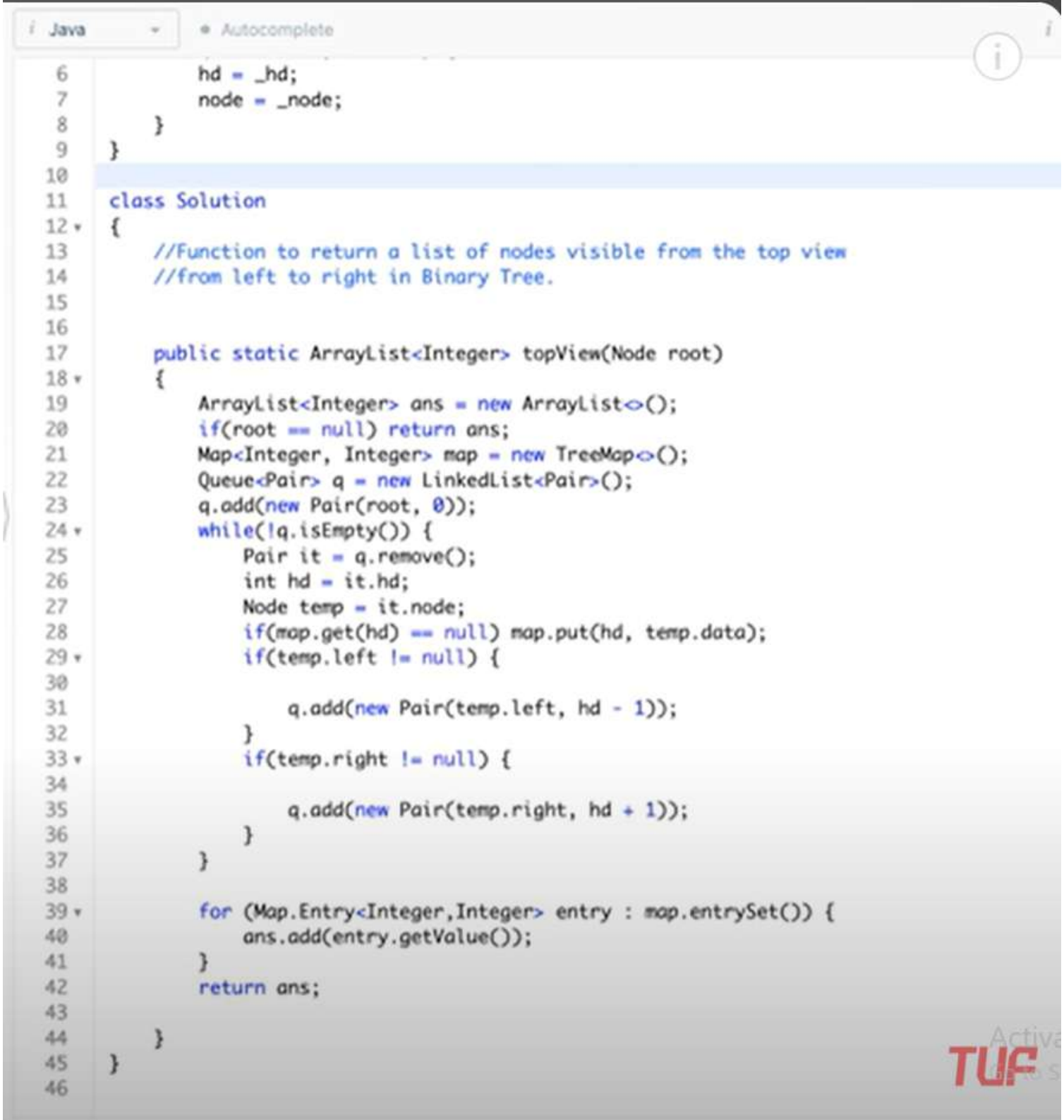
node = 1 2 3 4 5 7 6

0 -1 1 -2 0 2 -1

Top View of Binary Tree

LO




Java
Autocomplete
hd = _hd;
node = _node;
}
}
class Solution
{
//Function to return a list of nodes visible from the top view
//from left to right in Binary Tree.
public static ArrayList<Integer> topView(Node root)
{
ArrayList<Integer> ans = new ArrayList<>();
if(root == null) return ans;
Map<Integer, Integer> map = new TreeMap<>();
Queue<Pair> q = new LinkedList<Pair>();
q.add(new Pair(root, 0));
while(!q.isEmpty()) {
Pair it = q.remove();
int hd = it.hd;
Node temp = it.node;
if(map.get(hd) == null) map.put(hd, temp.data);
if(temp.left != null) {
q.add(new Pair(temp.left, hd - 1));
}
if(temp.right != null) {
q.add(new Pair(temp.right, hd + 1));
}
}
for (Map.Entry<Integer,Integer> entry : map.entrySet()) {
ans.add(entry.getValue());
}
return ans;
}
}