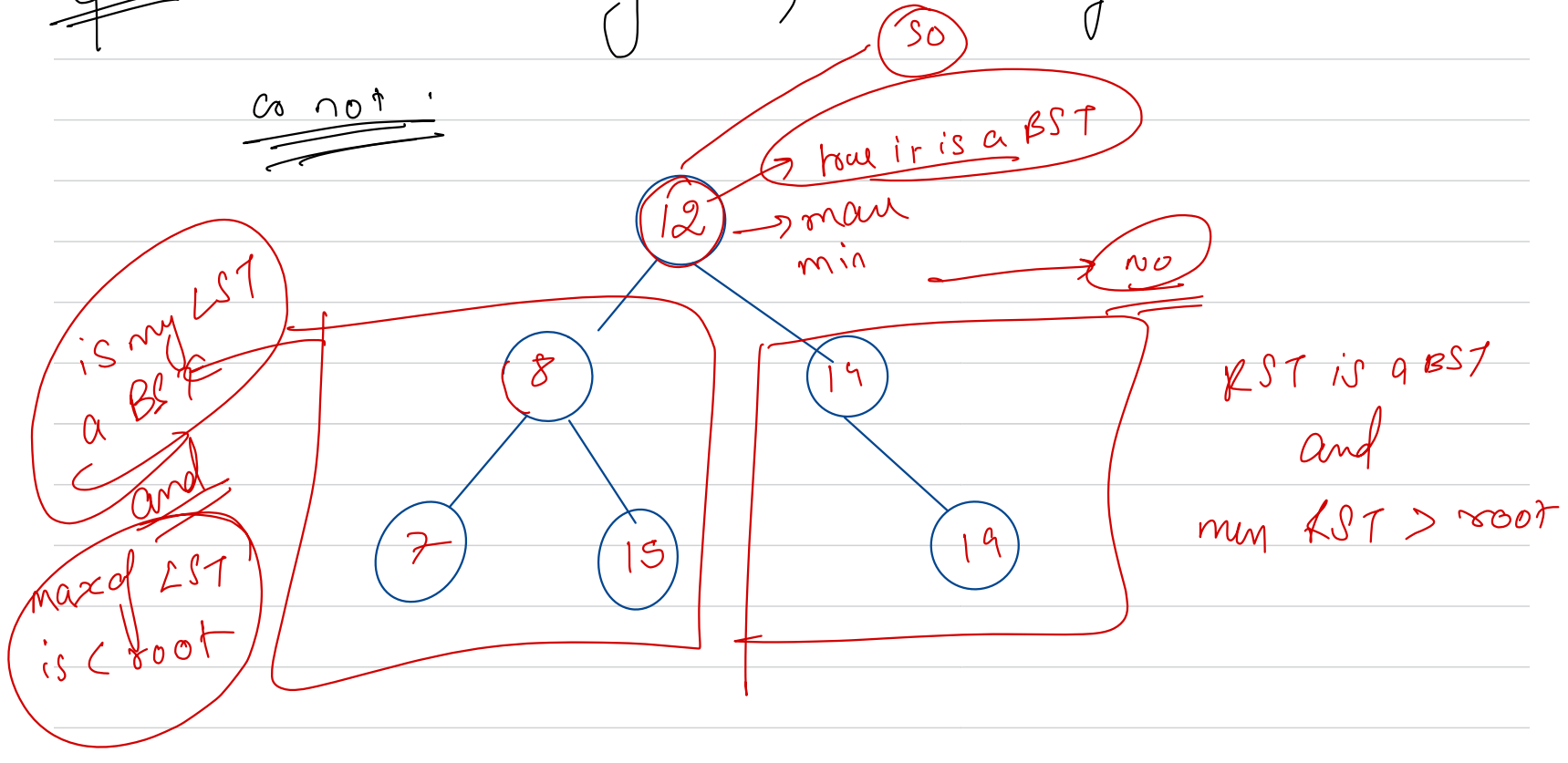
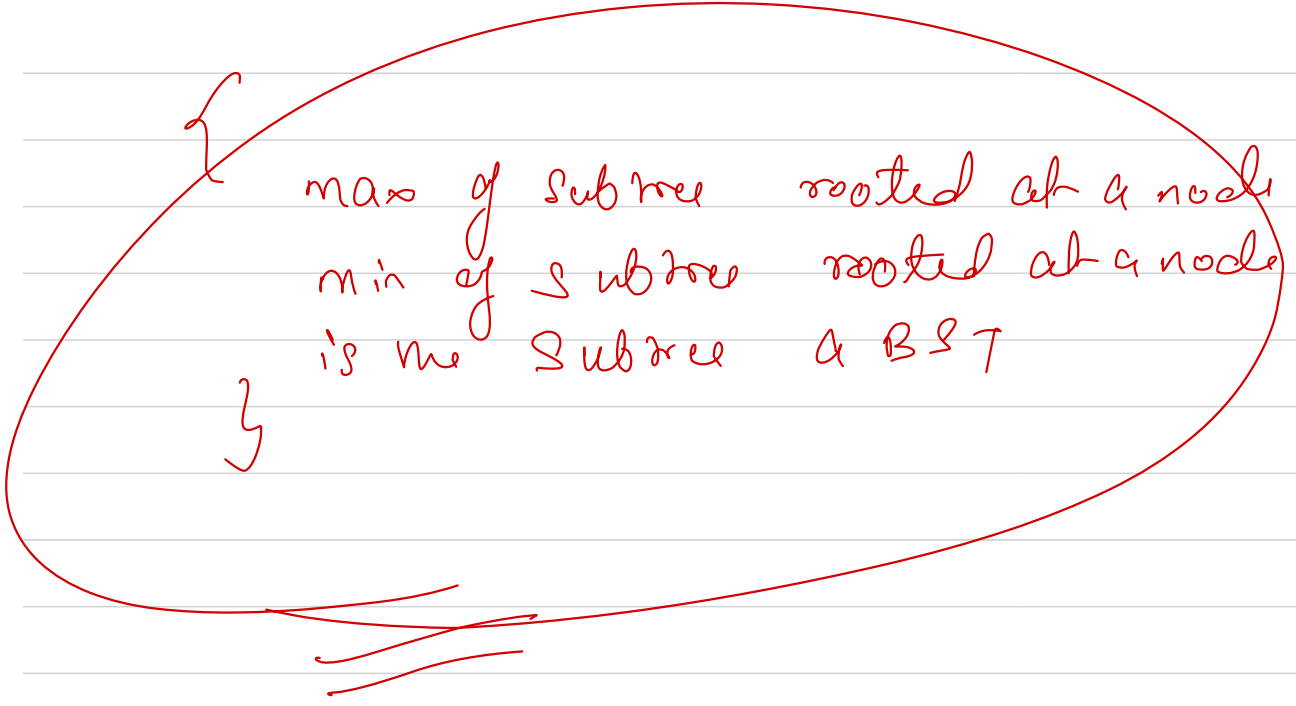
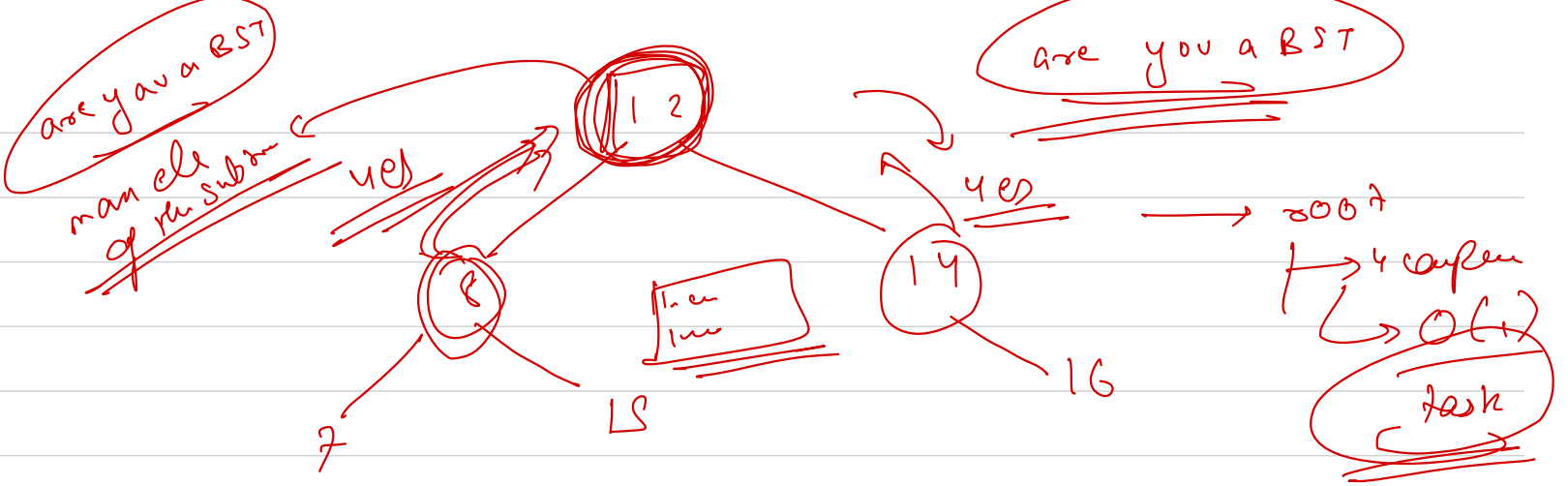
Q: Given a binary tree, check if it is a BST



Ca not :


{
max of subtree rooted at a node
min of subtree rooted at a node
is the subtree a BST
}

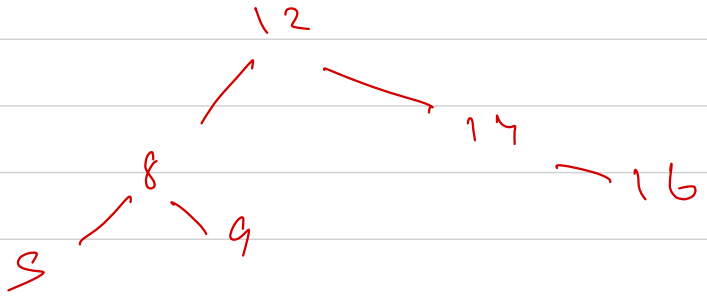


each node return 3 thing to it's root

- max of subtree
- min of subtree
- is my subtree BST

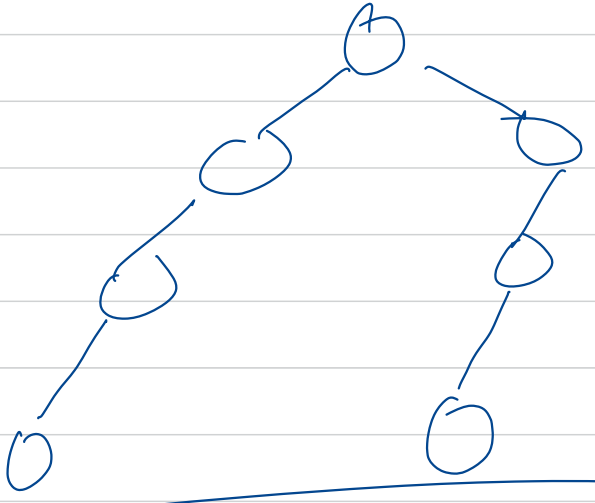
TC → O(n)

SC → O(h)



12 8 5 -1 -1 9 -1 -1 14 -1 16 -1 -1

Q: Given a BT, check if it is Balanced or not.

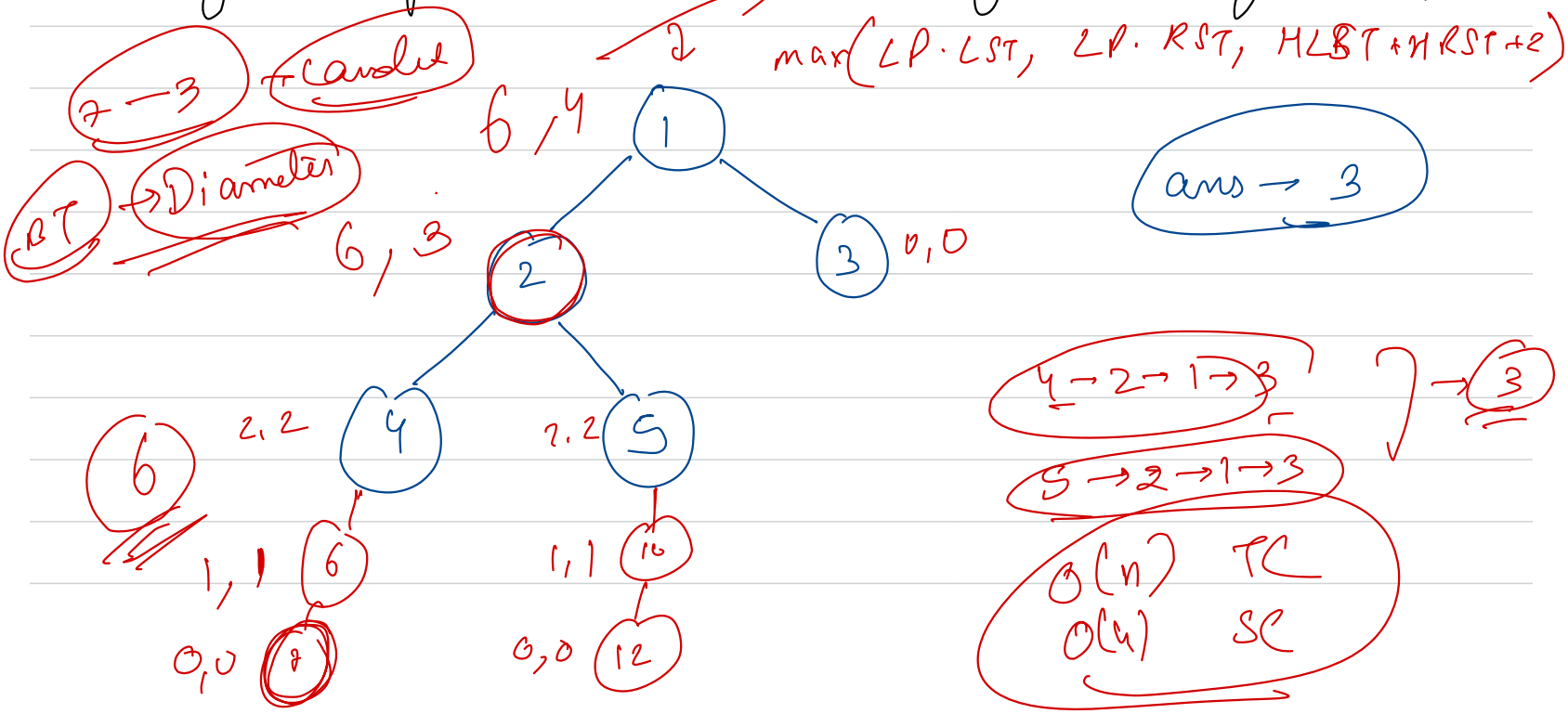


True

is balanced

height / class / struct (height , isBalanced)

Q. Given a binary tree, compute the length of the longest path between any 2 nodes of the BT.

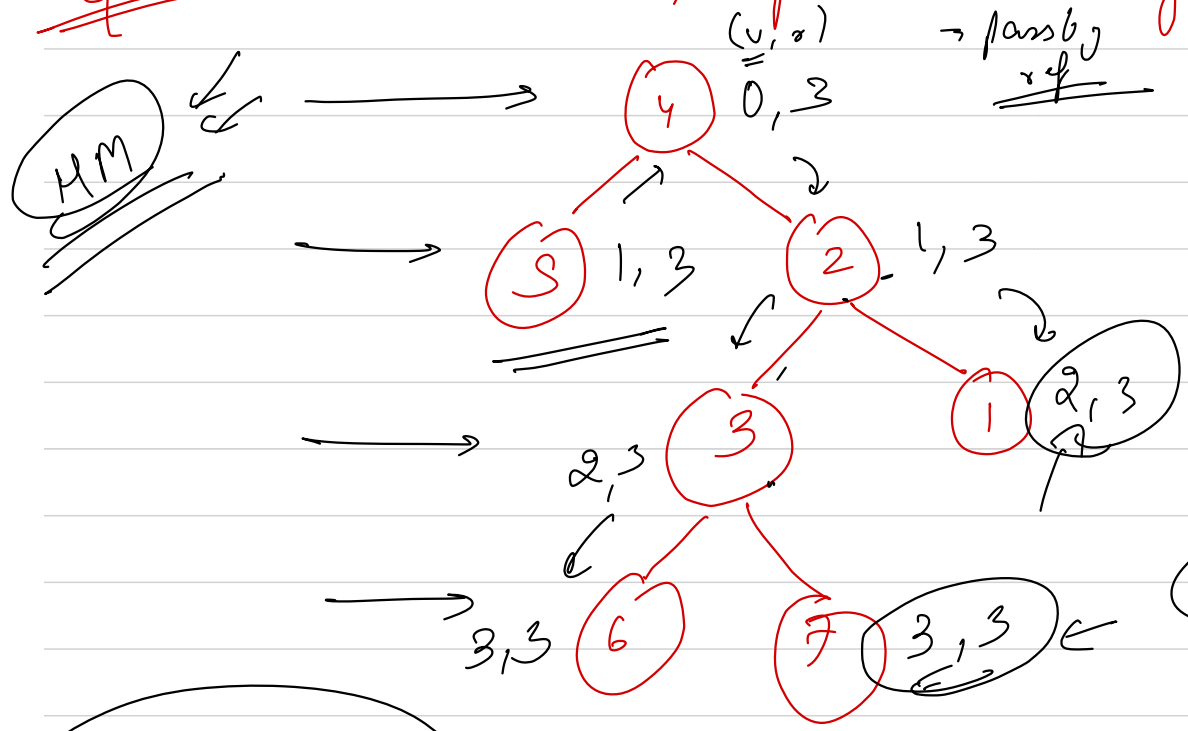


- There can be a case that some candidate of the longest-path comes root node.

or

the L.P comes only from LST or RST

Q:) Given a BT, print the left view.



→ pass by ref

HM

4, 5, 3, 6

level order traversal

iterative

4, 5, 3, 6

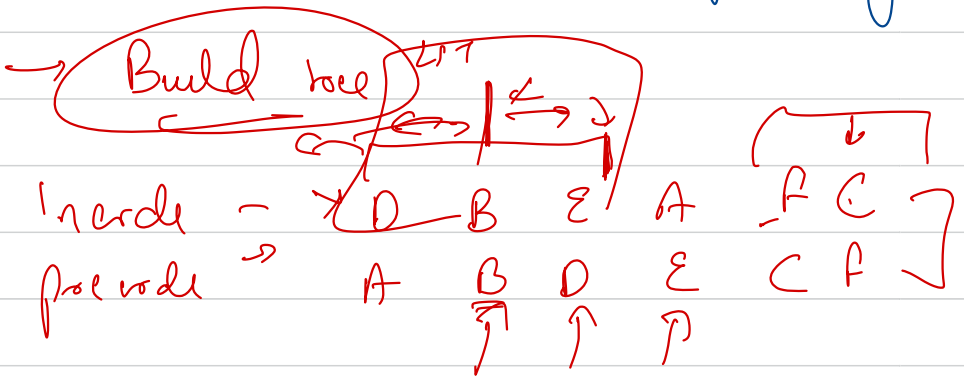
recursive preorder

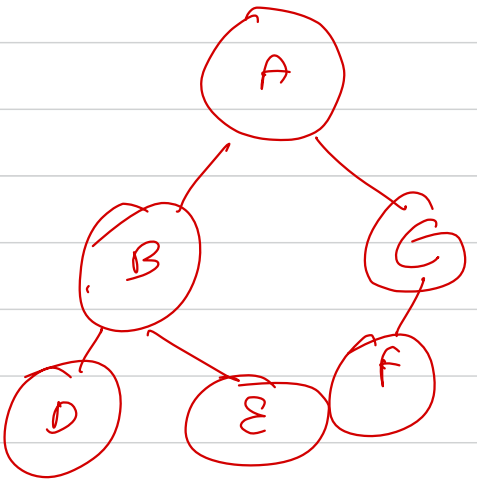
Right view

Right

Right

Q- Given a preorder, inorder & postorder traversal. Check if they are of same BT.





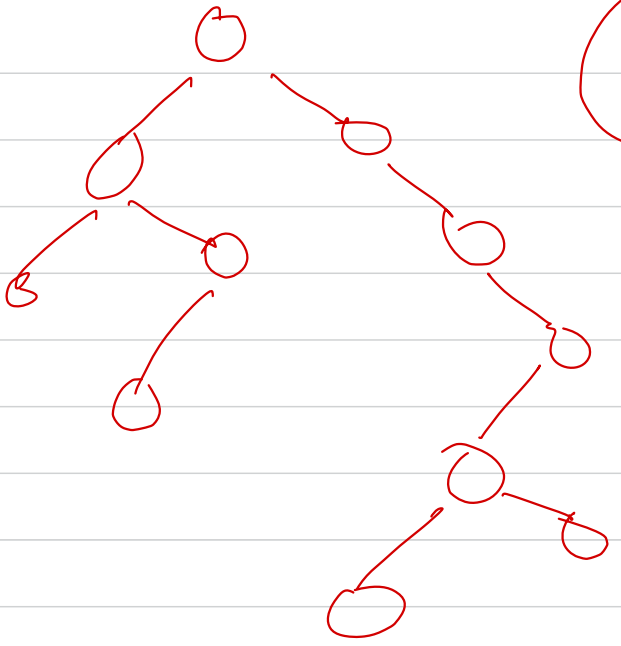
O(n x n)

if elements are unique

HashMap

O(n)

Q →



Boundary traversal