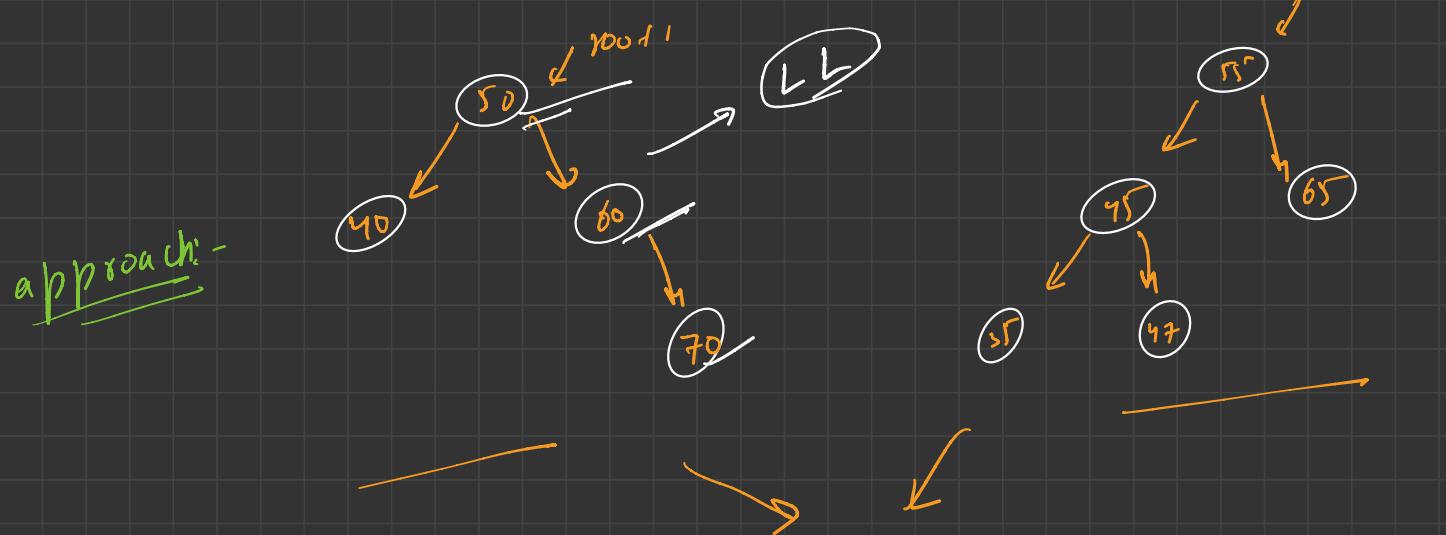
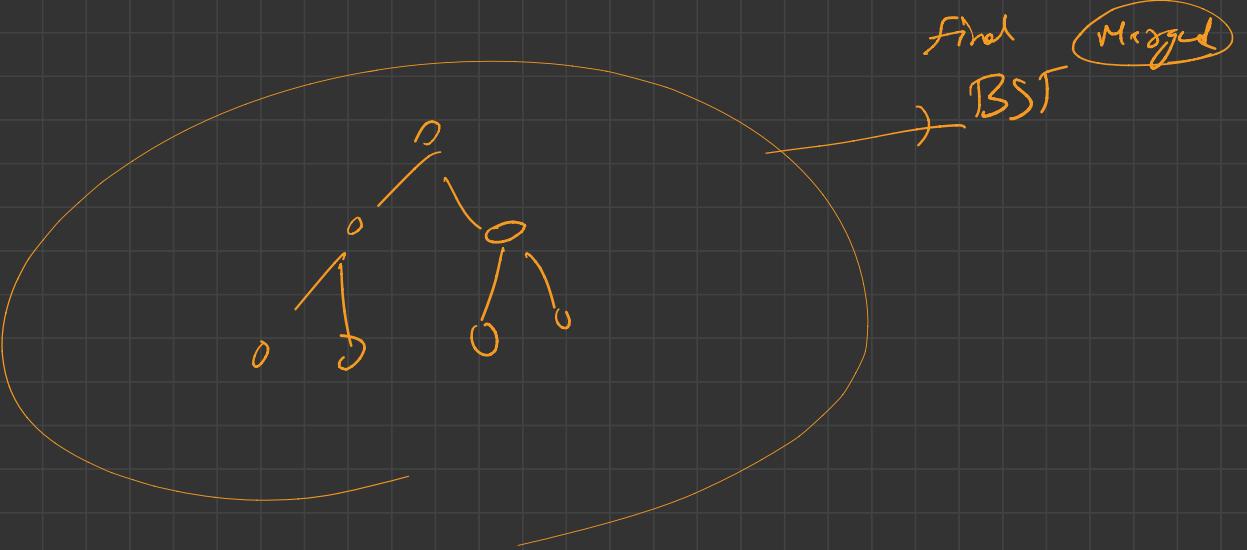



Merge 2 BST

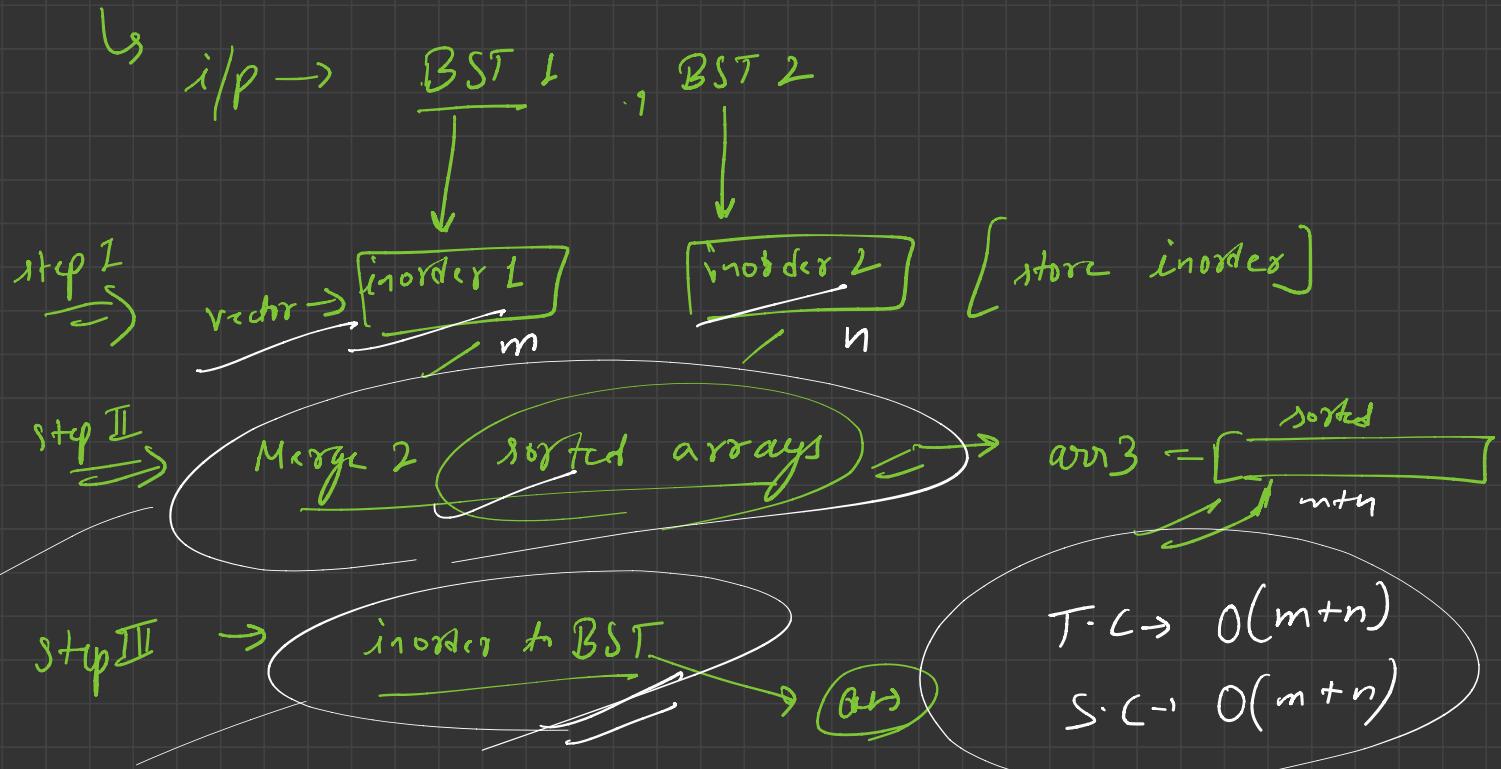
→ Merge 2 BST

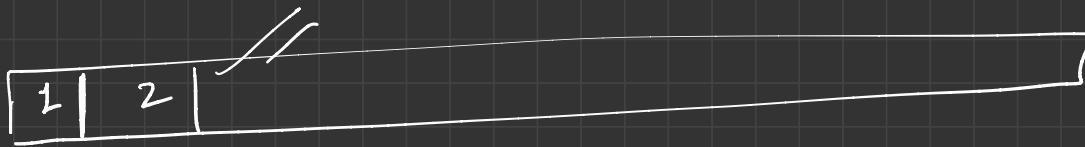
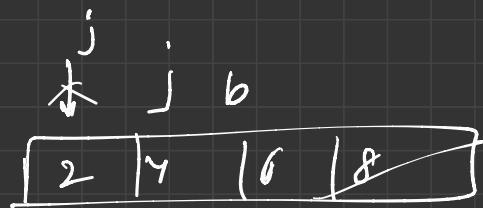
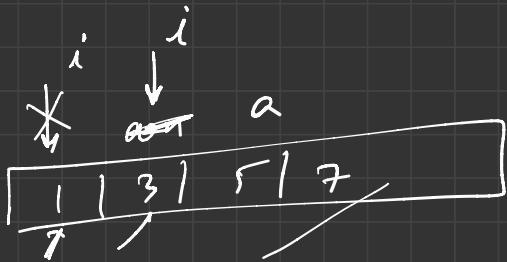




approach #1

inorder To BST =





ans $(a+b)$

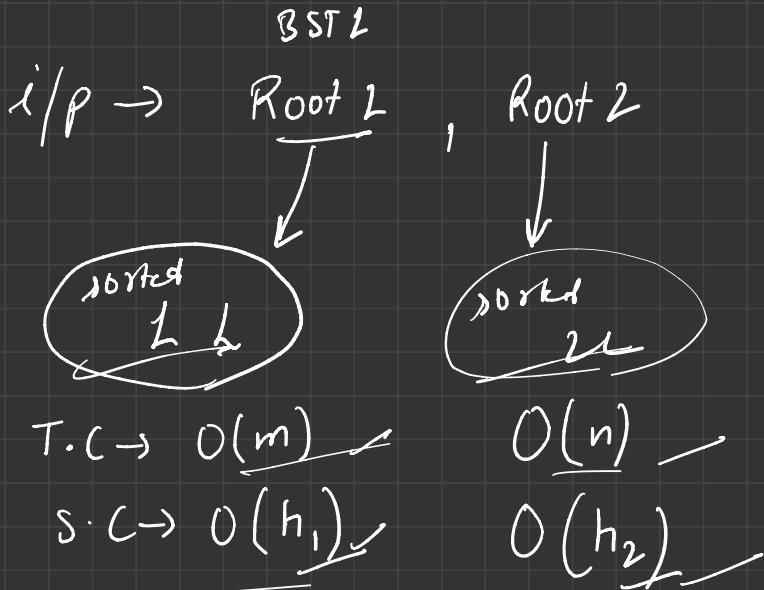
$\Rightarrow \mathcal{O}(m+n)$

S.C

$S.C \rightarrow \mathcal{O}(h_1 + h_2)$

Approach #2

convert BST
into sorted LL
step 1

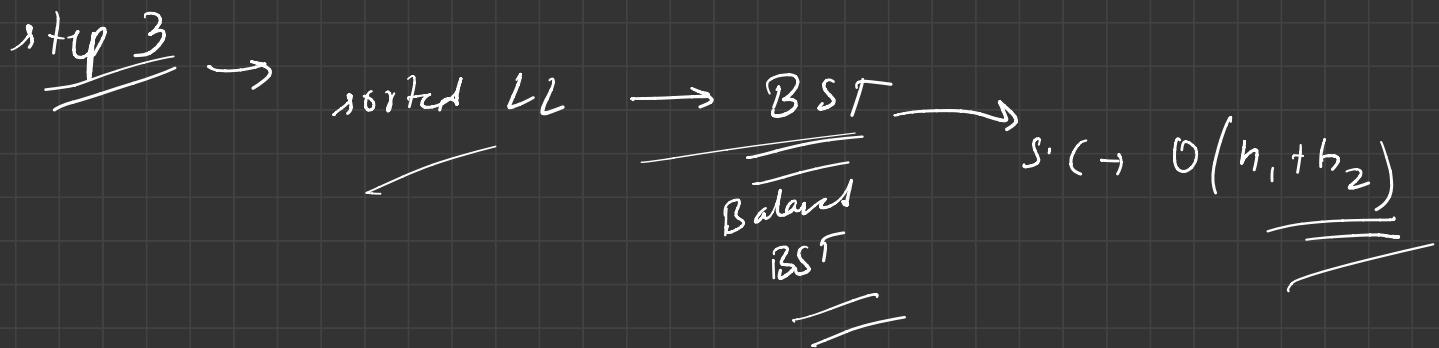


step 2 \rightarrow merge 2 sorted linked list

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

S.C $\rightarrow O(1)$

Sorted LL

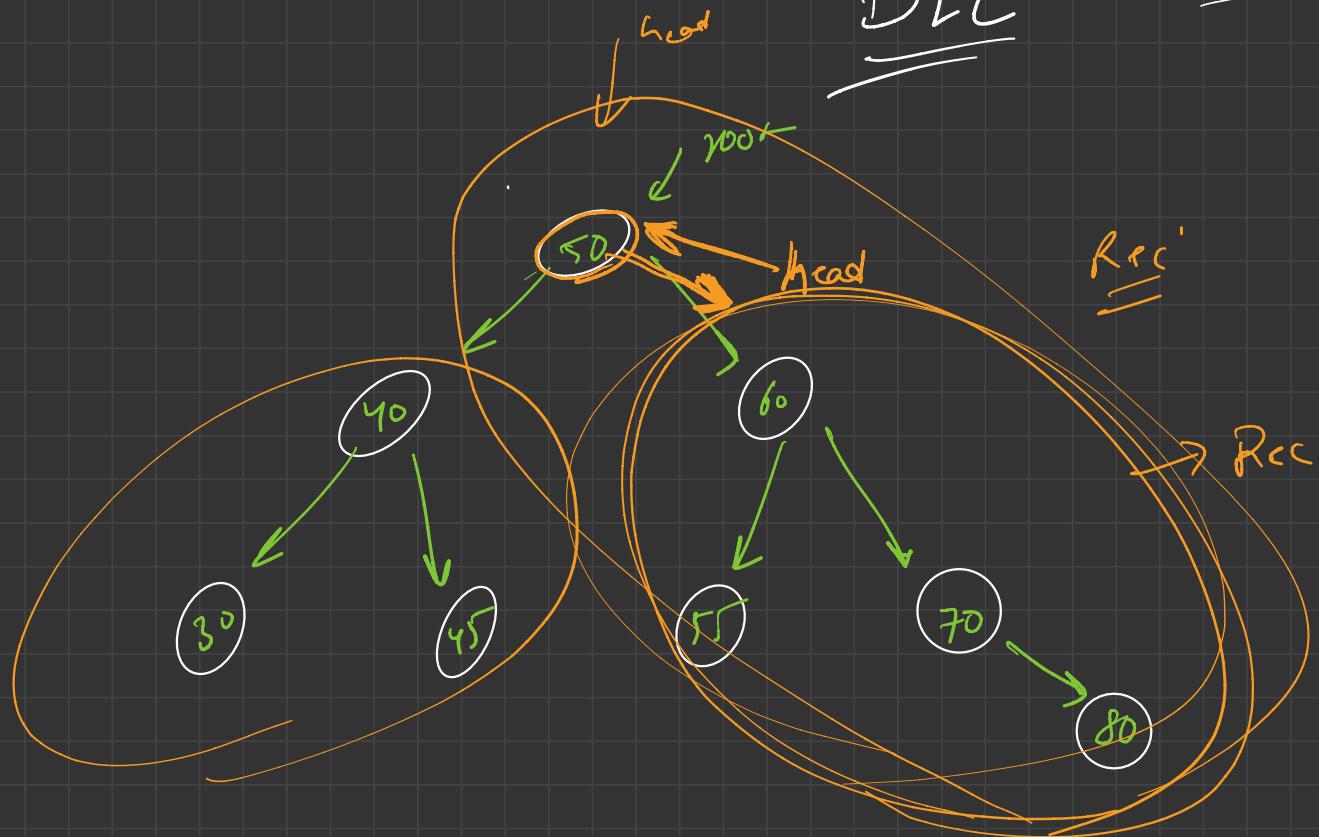


- ① (Convert a BST into sorted LL) → $O(n)$
- ② Merge 2 sorted linked list → $O(h)$
- ③ sorted LL → BST

\rightarrow convert a BST into SLL

$\underline{\underline{O(H)}}$

DLL

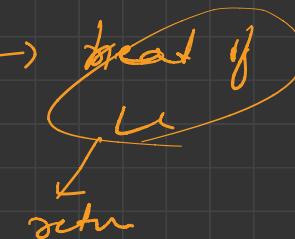


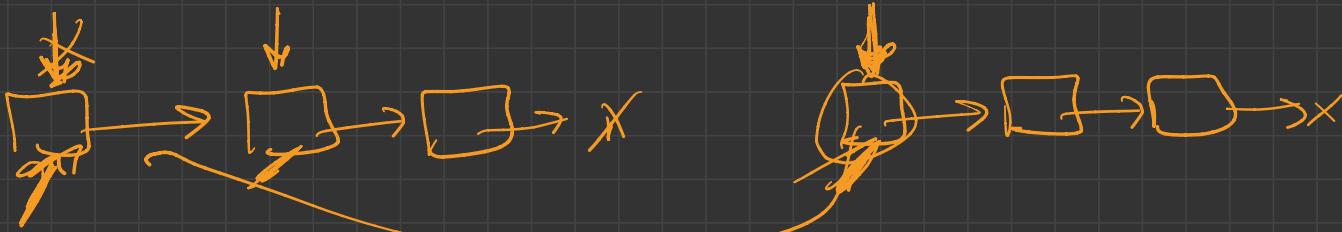
algo \rightarrow convert right subtract into LL \rightarrow head of LL

\rightarrow root \rightarrow right = head

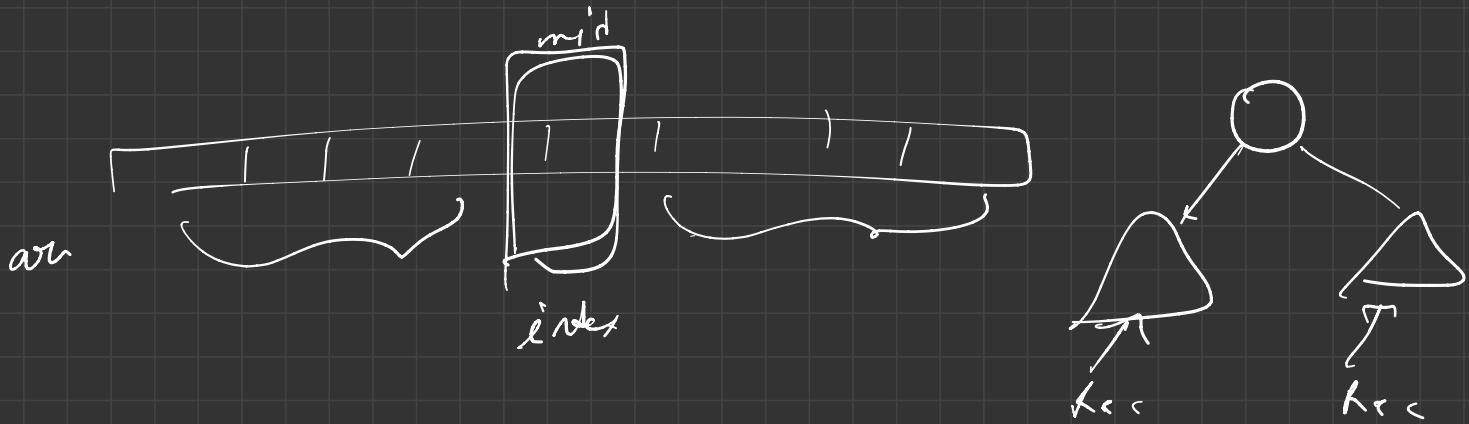
head \rightarrow left = root⁺

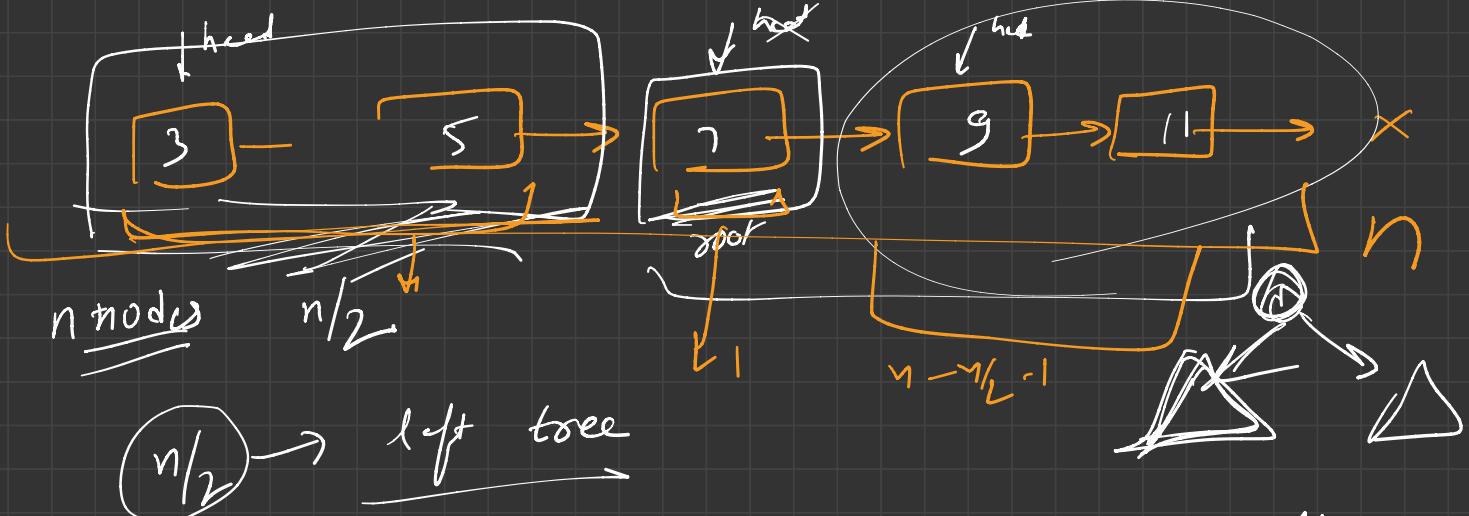
\rightarrow head = root

\rightarrow convert left part subtract into LL \rightarrow head of
LL




$\text{head} = \text{NULL}$
 $\text{tail} = \text{NULL}$





$n/2 \rightarrow$ left tree

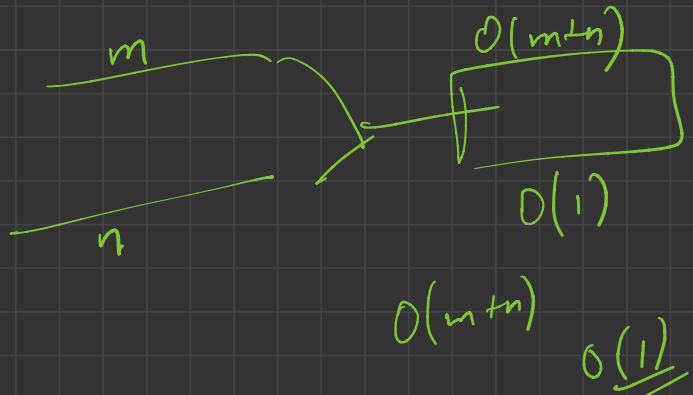
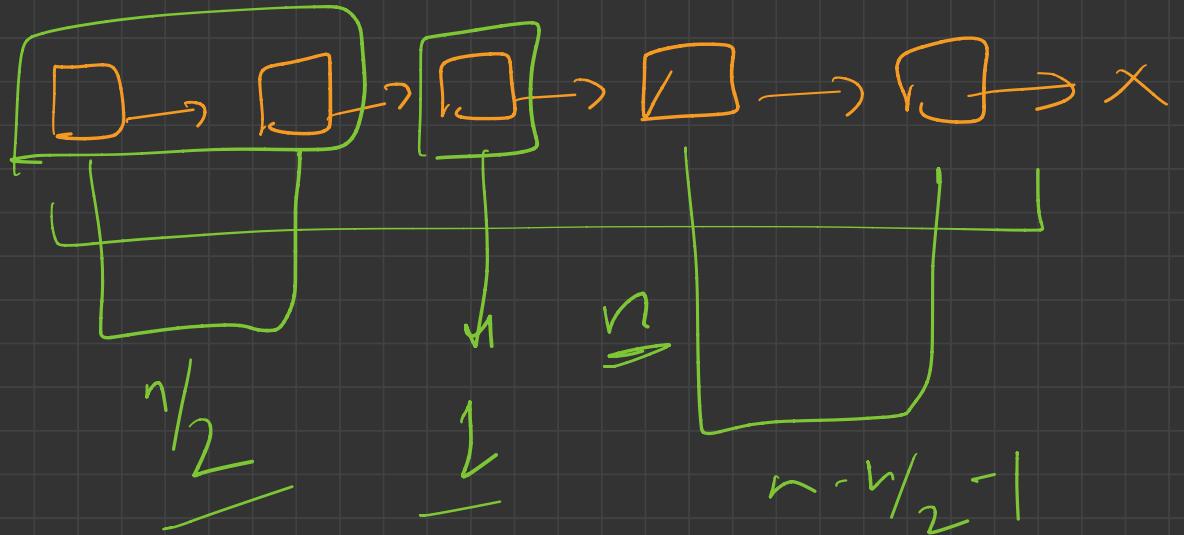
① $n/2 \rightarrow$ left tree \Rightarrow left

② create root

root \rightarrow left = left

③ head \rightarrow next

④ right subtree handle
 $root \rightarrow right = Rec(right, n - n/2 - 1);$



$S.C \rightarrow O(4) \cong$

$H(v \rightarrow Dry Run \rightarrow 2-3)$

$W \rightarrow$ Dy Run

where y times \rightarrow T-C

S-C \rightarrow 