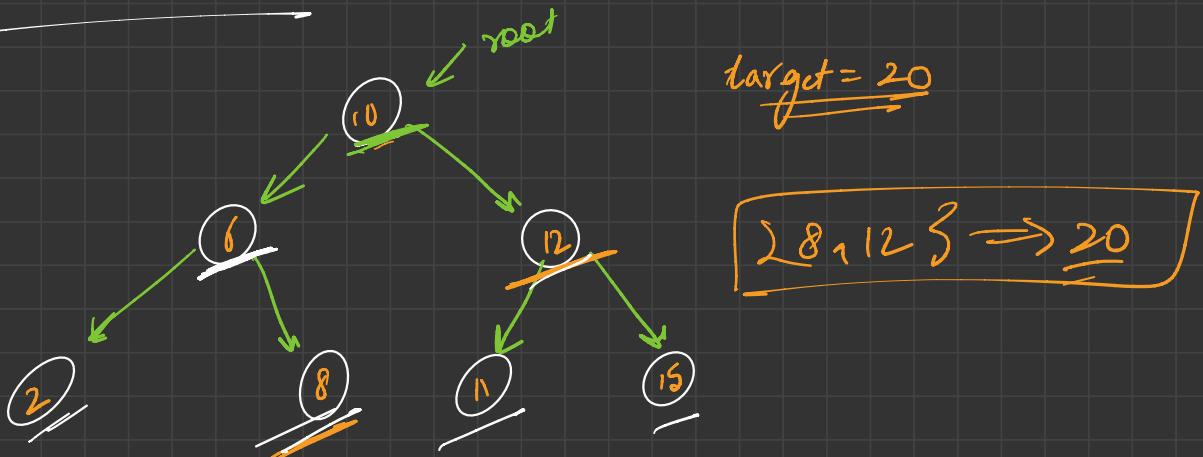



Binary Search Trees

→ 2 sum in BST

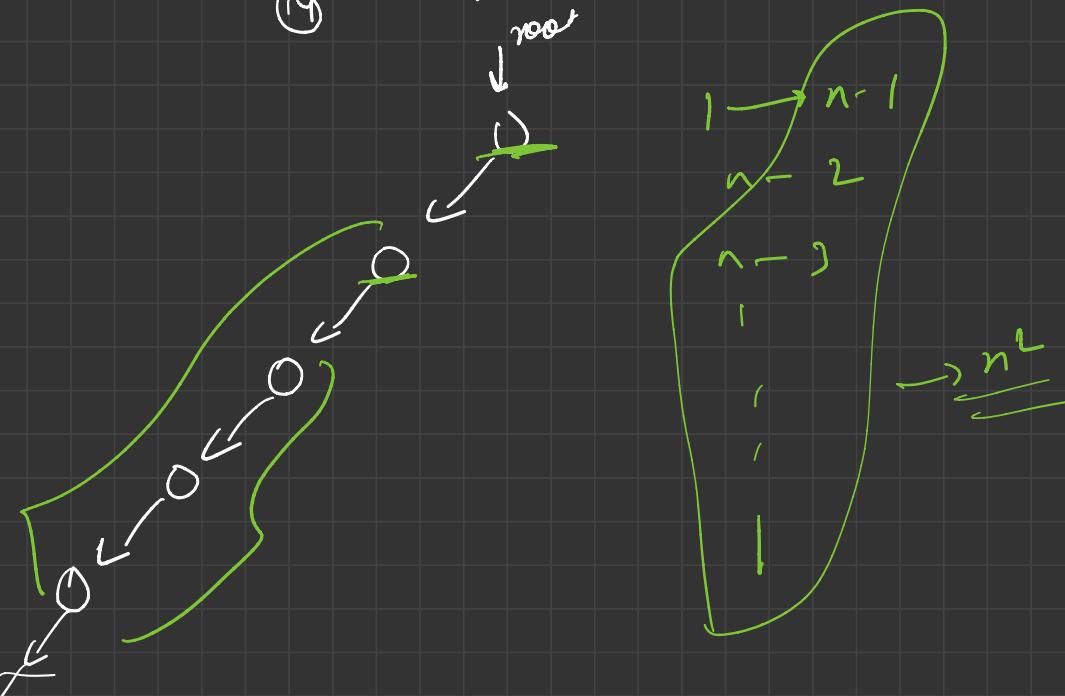


approach:-

Brute force

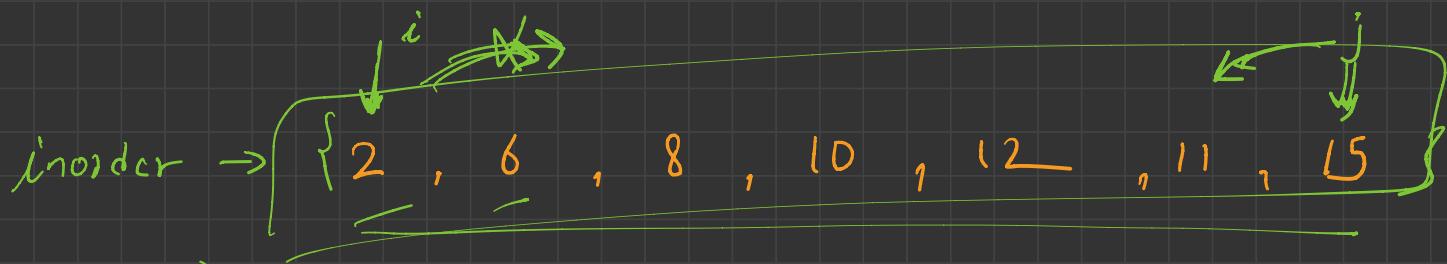
$$10, \quad \begin{array}{|c|} \hline \text{search in left/Right} \\ \hline \end{array} \quad (\text{target} - 10) = 10 \quad \rightarrow \quad \underline{\mathcal{O}(N^2)}$$

$$6 - \begin{array}{|c|} \hline \text{target} - 6 \\ \hline \end{array} \xrightarrow{(14)} \text{left/Right}$$



$\Rightarrow O(N) \rightarrow$ Single traversal

BST \rightarrow inorder sorted



T. $\hookrightarrow O(N)$

S. $\hookrightarrow O(N)$

$$\text{target} = 20$$

$$arr[i] + arr[j] \rightarrow \text{sum}$$

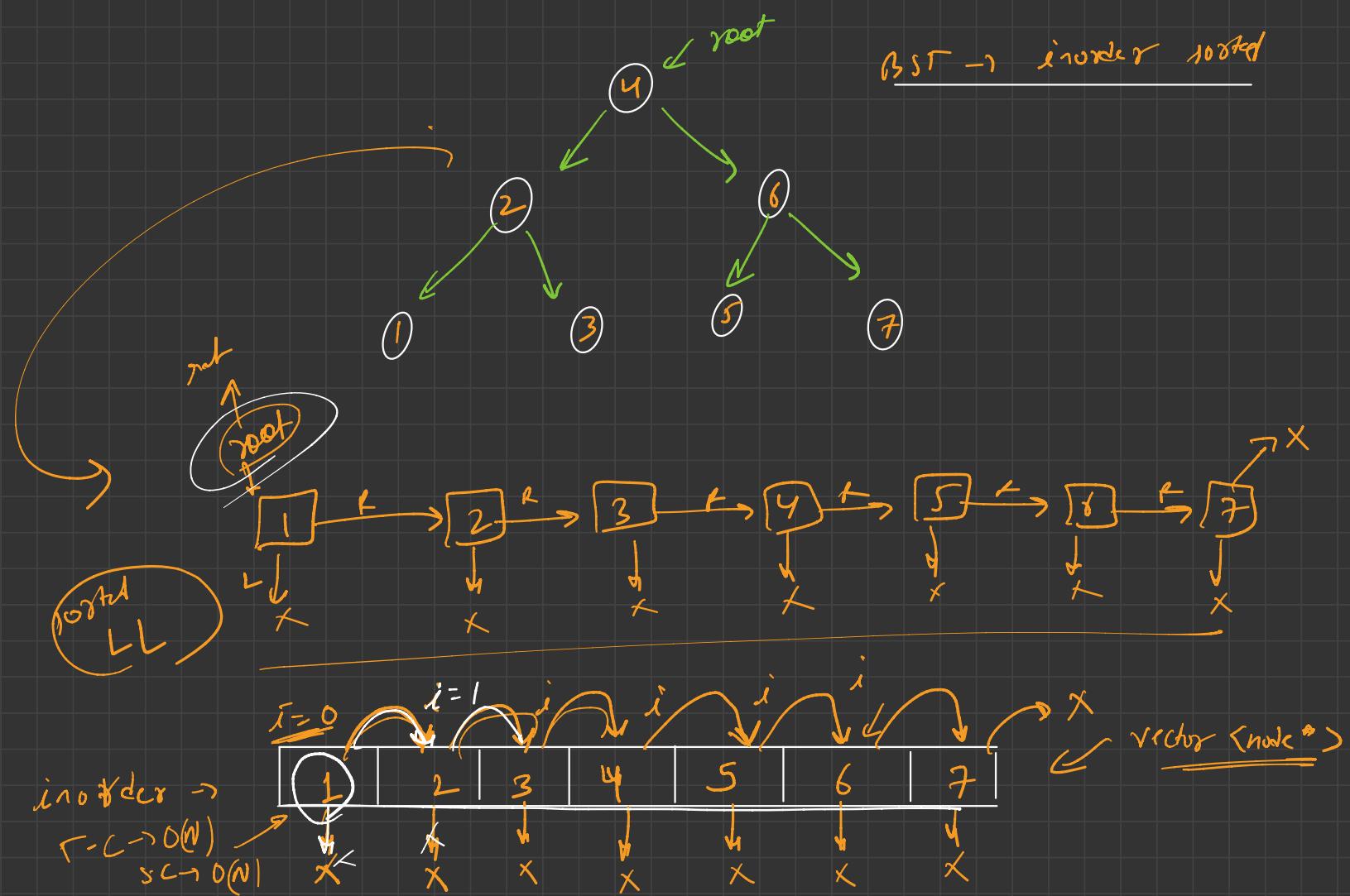
sum = target

\hookrightarrow return ans

$$\underline{\text{sum} > \text{target}} \rightarrow j - 1$$

$$\underline{\text{sum} < \text{target}} \rightarrow i + 1$$

BST \rightarrow inorder sorted



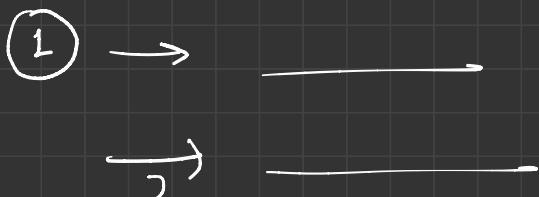
frds \hookrightarrow last element

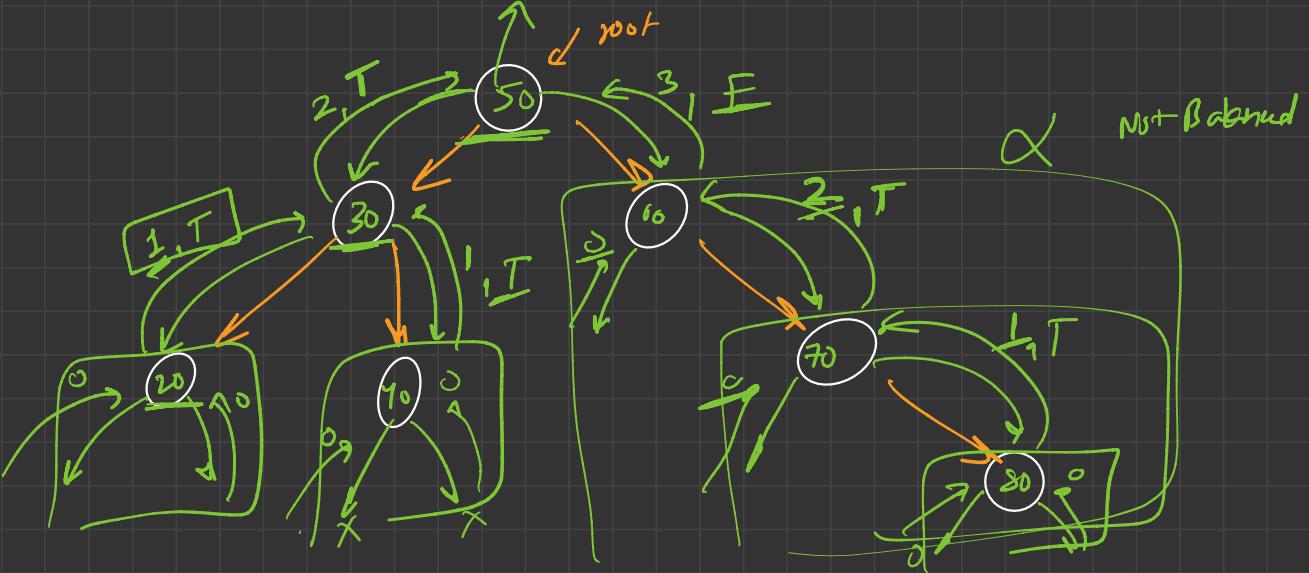
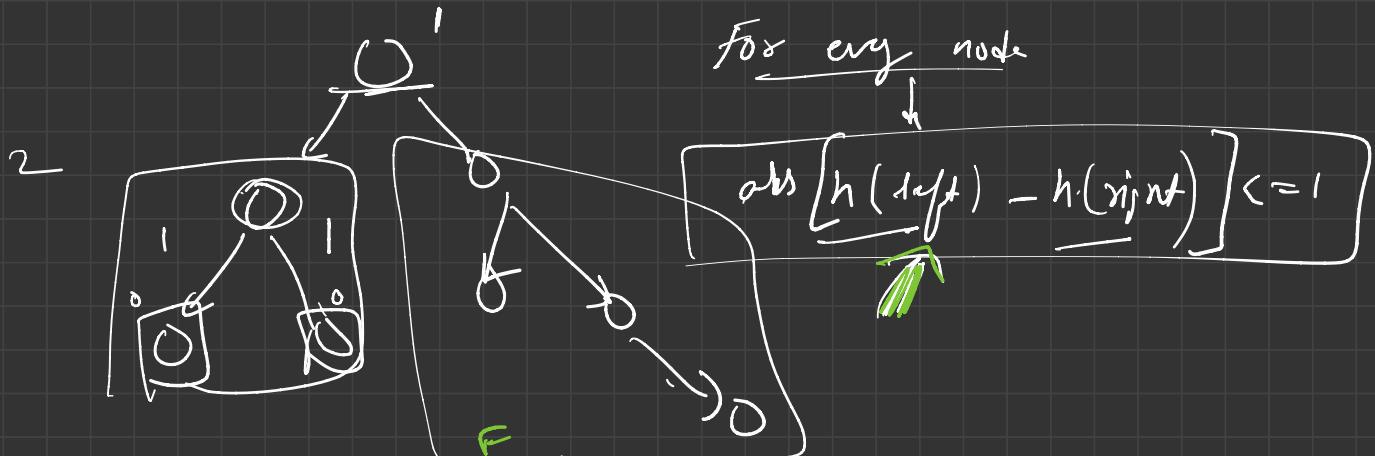
algo \rightarrow inorder \uparrow store
(vector<node>)

\rightarrow left \rightarrow x
right \rightarrow next Element

\rightarrow final step \rightarrow last char
left \rightarrow x
right \rightarrow x

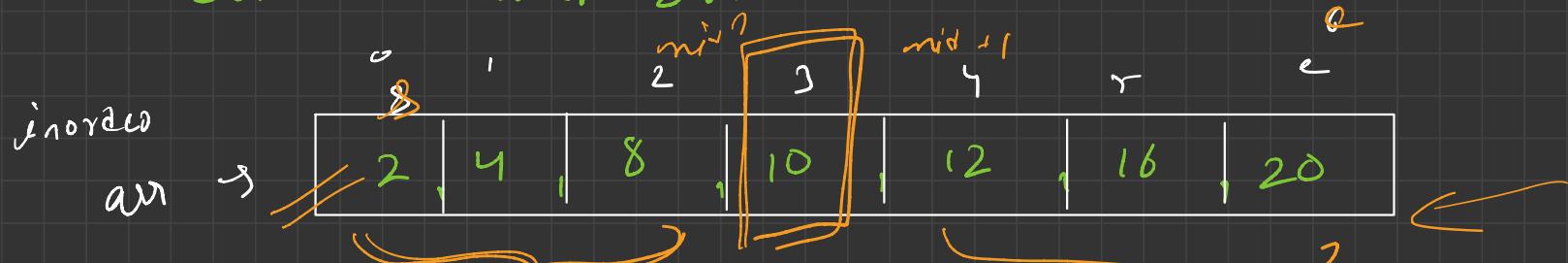
algo:-) \rightarrow store inorder \rightarrow (vector<int>)



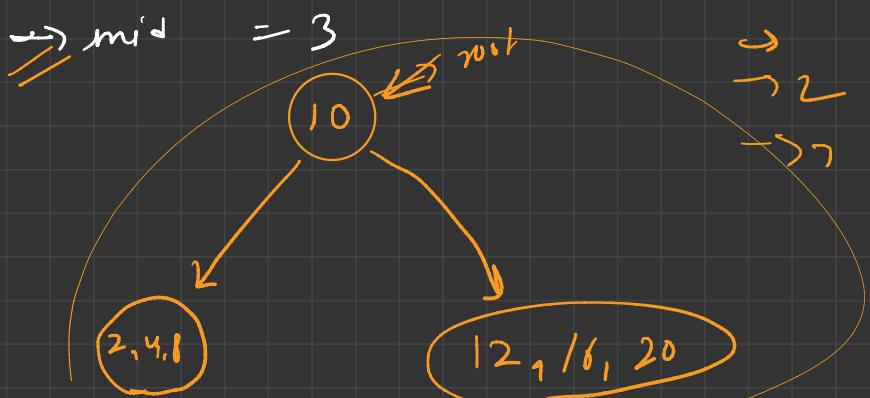


$\rightarrow i/p \rightarrow$ Normal BST

BST \rightarrow inorder sorted



inorder
arr
BST

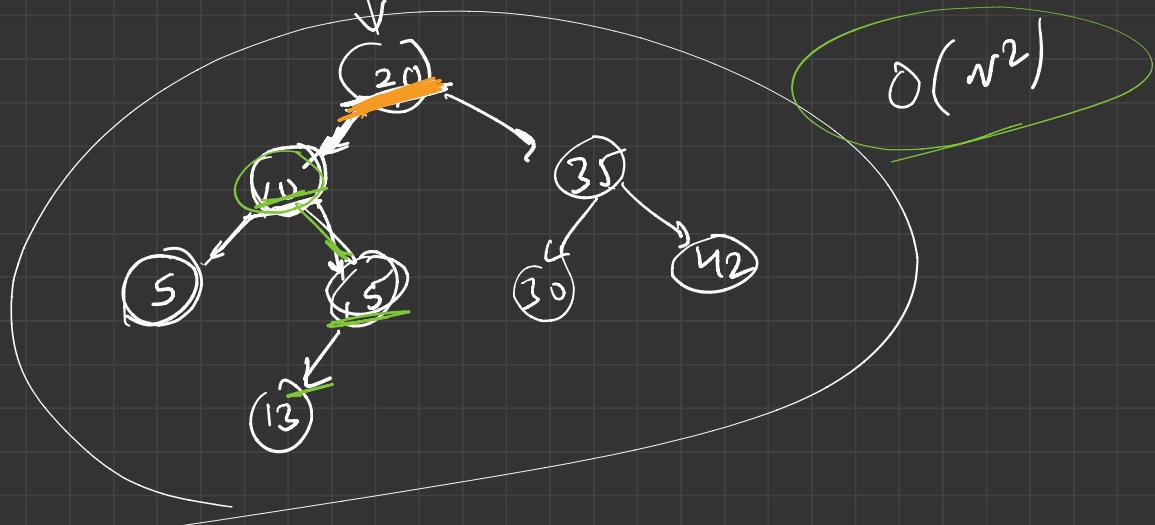


→

BST
from PreOrder

Approach #1

[20, 10, 5, 15, 13, 35, 30, 42]



App ②

PreOrder → []

Inorder → []

→ sort Preorder → Inorder

→ TREE → $O(N \log N)$

Approach ↗

Desired → T-C → $O(N)$

Preorder \rightarrow

N
↓
root

L R
↓
left ↓
right

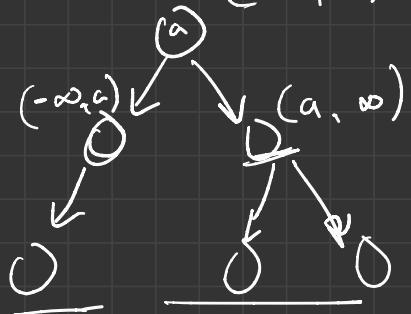
Postorder

0	1	2	3	4	5	6	7
20	10	5	15	13	35	30	42

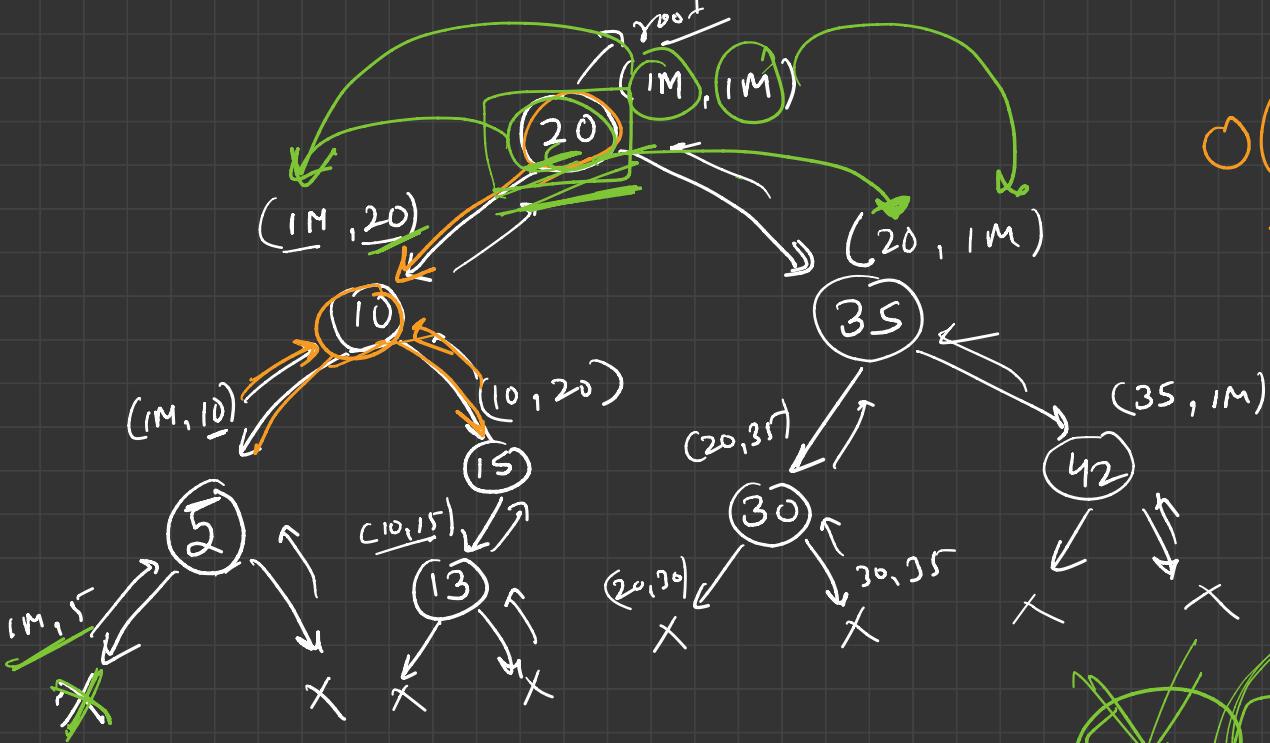
↑
1

1) BST

$\boxed{\text{INT_MIN}, \text{INT_MAX}}$
 $(-\infty, \infty)$



\rightarrow $[\underline{\text{INT_MIN}} \quad \underline{\text{INT_MAX}}]$



$O(3N)$

$\cancel{O(N)}$

$\sim O(N)$

T.C

