BST intro

in this tree all elements in the left side are smaller then root and on right side greater (similar to binary search) applied to all subtrees

① No Duplicates

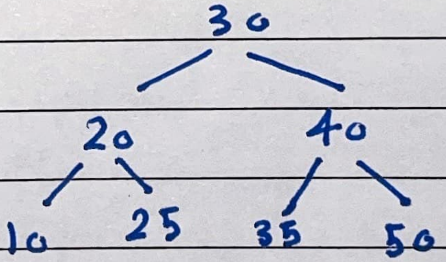
② Inorder is in Sorted order

③ Rember catlon number (for n nodes X BST can be created) $X = \frac{2nC_n}{n+1}$

representation can be Array or Linked-List most used

All elements are in left are smaller then root aswell right bigger

Searching Procedure



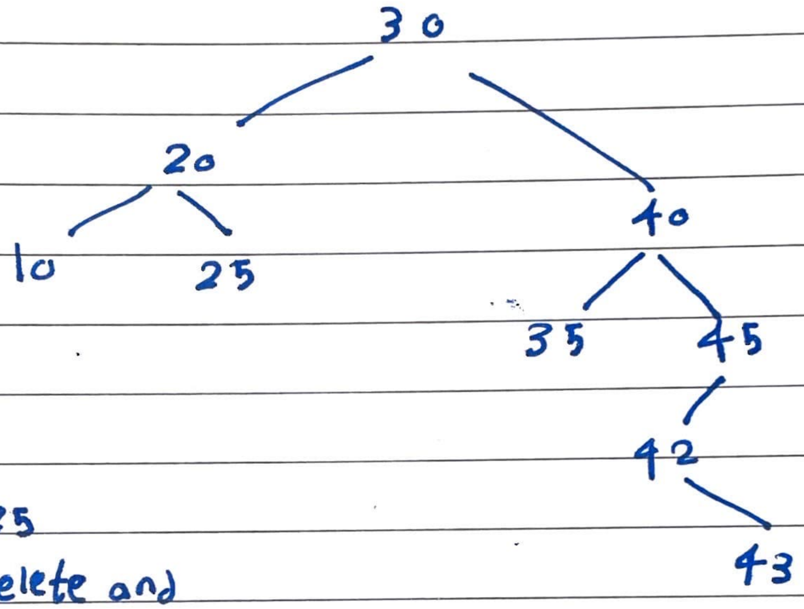
Key=25

1. compare with 30, 25 < 30 go left side
2. compare with 20, 25 > 20 go right side
3. Found since 25 ≮ 25, 25 ≯ 25

Time complexity based on tree height

/ /

How To delete From BST



1. Key=25
directly delete and
make Parent right = Null
2. Key=42 (1 child) Switich with child then delete
3. key=30 (2child) Find in order Preccessor (25) OR
inorder successor 35 then Switich, delete
4. key=40   35 OR 42 but in case choice was 42

Again you need to modify its old Place (Predessor OR Successor)

so change 43 again because you choose 42 and it has childs not as 30 with 25 OR 35 they don't have children

Generating Binary tree Search Preorder

Here we need only Preorder or Post order
video 308 For more understanding

we need Stack