Construct a Binary tree from Preorder and Inorder.

$$\begin{tikzcd}
\text{pre order} \arrow[r, bend left] & \text{root lst rst}
\end{tikzcd}$$

pre order : {10, 20, 40, 50, 80, 30, 60, 90, 70}

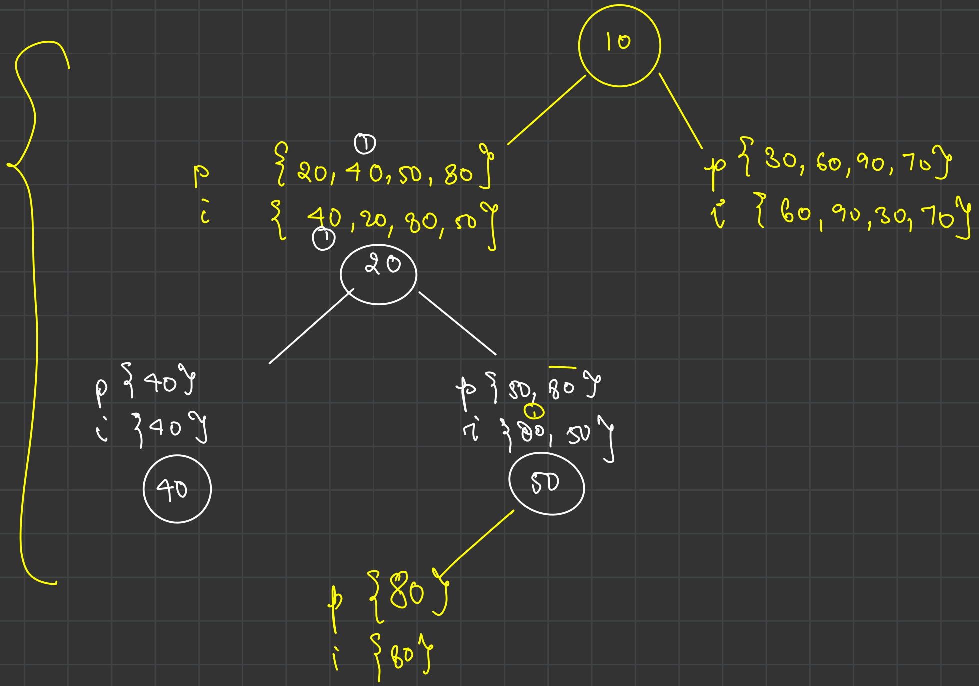
in order : {40, 20, 80, 50, 10, 60, 90, 30, 70}

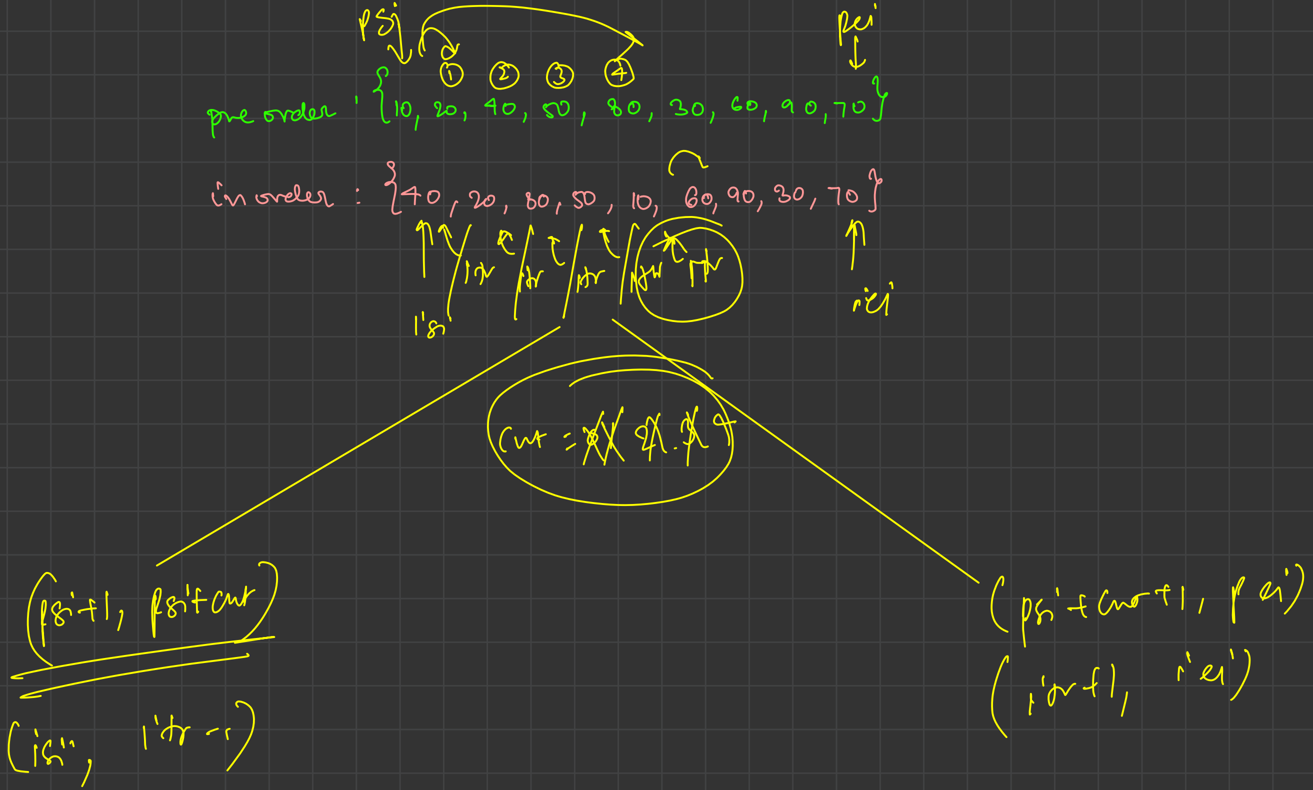
$$\begin{tikzcd}
\text{in order} \arrow[r, bend right] & \text{lst root rst}
\end{tikzcd}$$

pre order : $\{ \underline{\underline{10}}, \underbrace{\overset{①}{20}, \overset{②}{40}, \overset{③}{50}, \overset{④}{80}}_{lst}, \underbrace{30, 60, 90, 70}_{rst} \}$

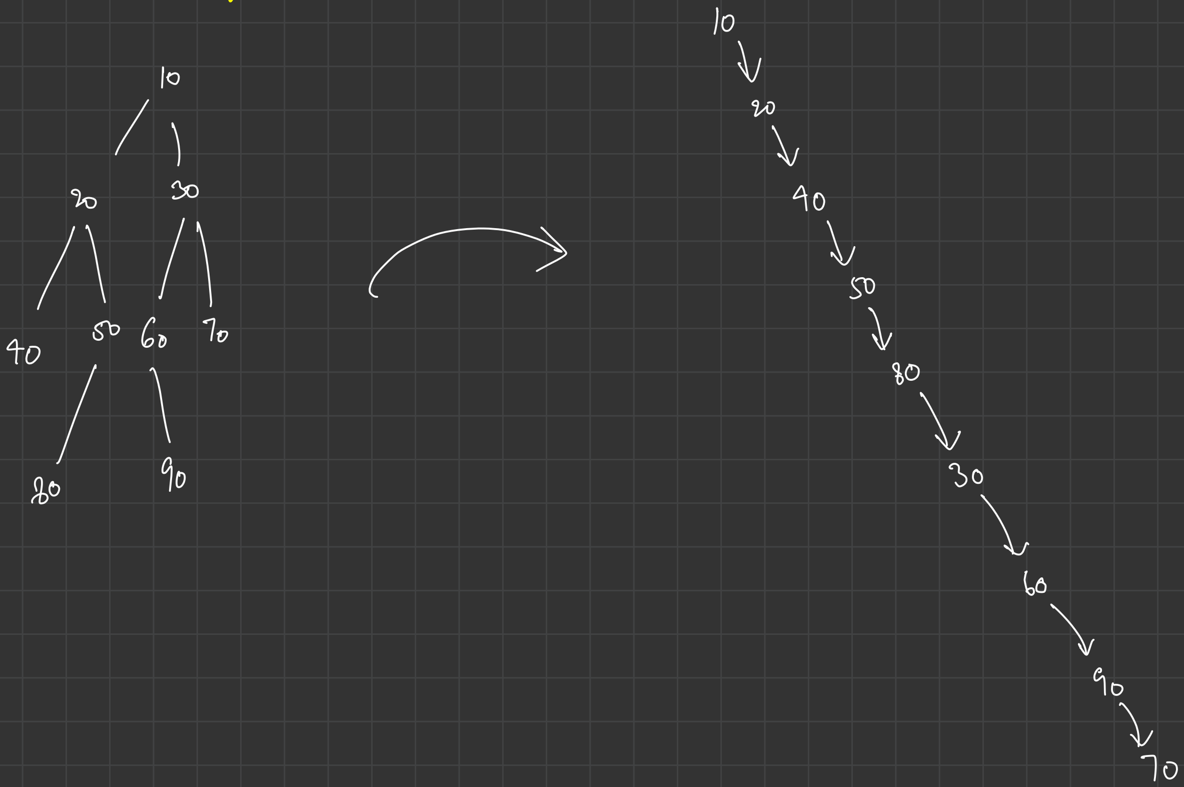
in order : $\{ \underbrace{\underset{①}{40}, \underset{②}{20}, \underset{③}{80}, \underset{④}{50}}_{lst}, \underline{\underline{10}}, \underbrace{60, 90, 30, 70}_{rst} \}$

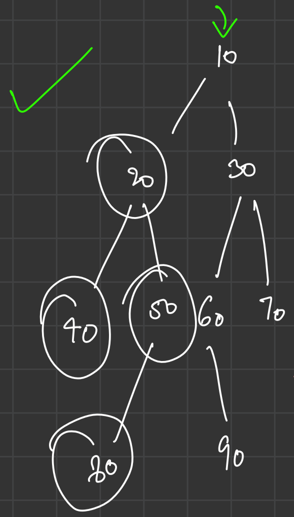
construct() ✓ → takes pre order & inorder of a tree and construct that tree

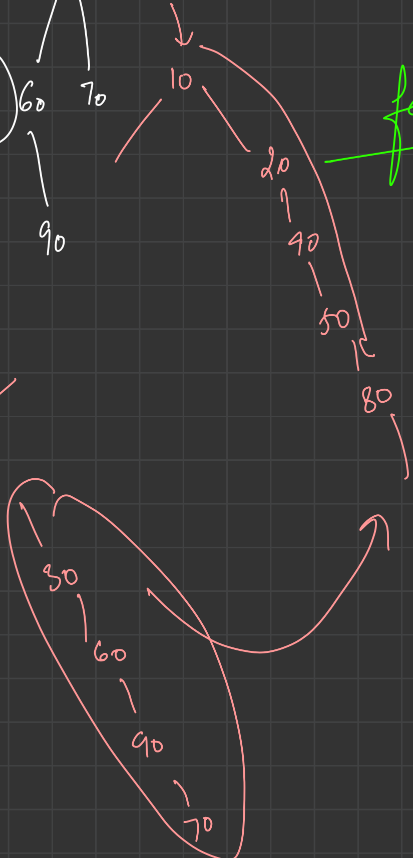



psi
pei
① ② ③ ④
pre order : {10, 20, 40, 50, 80, 30, 60, 90, 70}
in order : {40, 20, 80, 50, 10, 60, 90, 30, 70}
isi
idx idx idx idx idx
iei
cnt = ~~1~~ ~~2~~ ~~3~~ 4
(psi+1, psi+cnt)
(isi, idx-1)
(psi+cnt+1, pei)
(idx+1, iei)

flatten Binarytree { inplace }







flatten(Node root)

faith: it will flatten the tree

① Step 1 flatten(root.left)

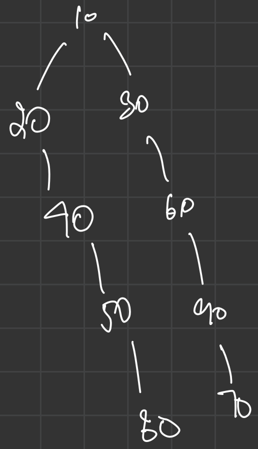
② Step flatten(root.right)

flatten(Node root)

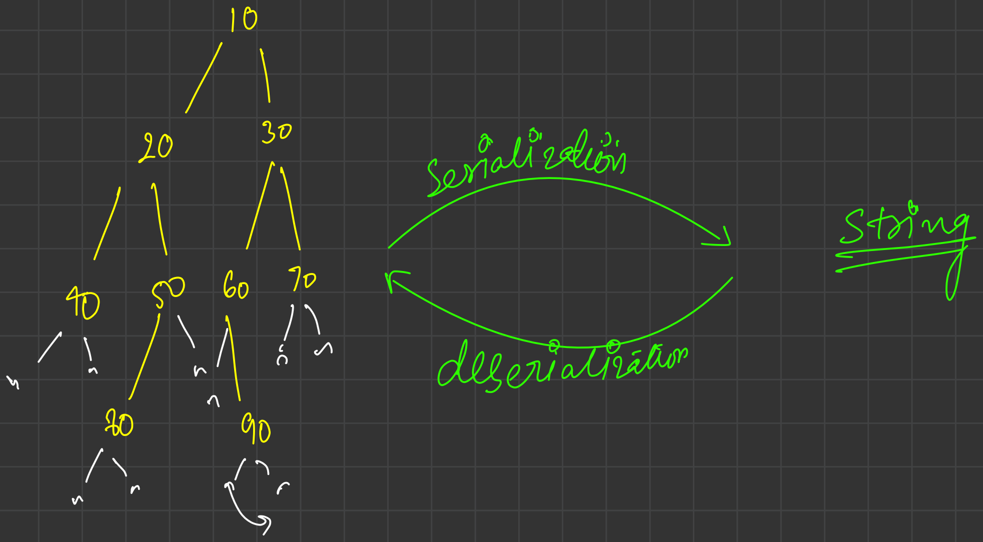
faith: It flattens a Binary Tree

① flatten(root.left)

② flatten(root.right)

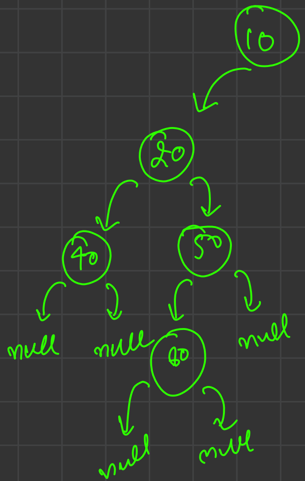


Serialize and deserialize a Binary tree.

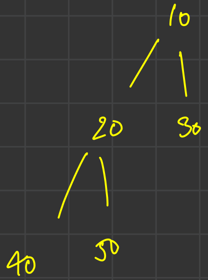


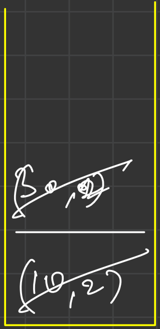
tree = "10, 20, 40, null, null, 50, 80, null, null, null, 30, 60, null, 90, null, null, 70, null, null"

Pre = ["~~10~~, 20, 40, null, null, 50, 80, null, null, null, 30, 60, null, 90, null, null, 70, null, null"]



pre order traversal of a tree (iteratively)





Stack

10, 20, 40, 50, 30

call 0 $\longrightarrow$ print + left call ✓

call 1 $\longrightarrow$ right call

call 2 $\longrightarrow$ remove