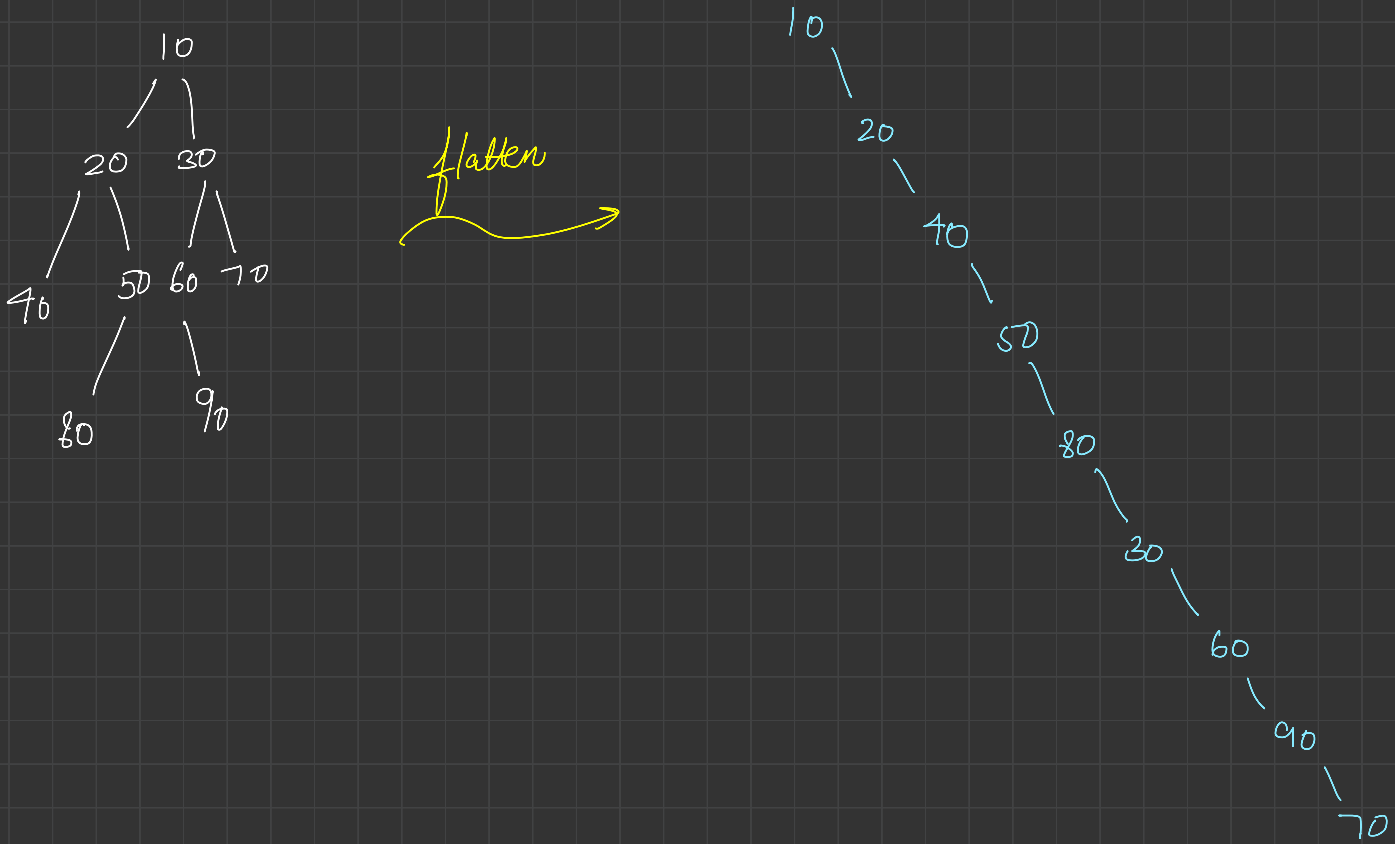
Flatten a Binary Tree



faith : flattens the given BT

```
void flatten( Node root)
{   if (root = = null) return

    flatten(root.left);

    flatten( root. right);

    Node temp = root.right;
    root.right = root.left;
    root.left = null;

    Node curr = root;
    while (curr.right != null)
        curr = curr.right;
    curr.right = temp;
}
```

Construct a tree from PreOrder & InOrder

root, pre LST, pre RST

PreOrder: 10, 20, 40, 50, 60, 30, 70, 90, 80

{root, lst, rst}

In LST, root, In RST

InOrder: 40, 20, 60, 50, 10, 70, 90, 30, 80

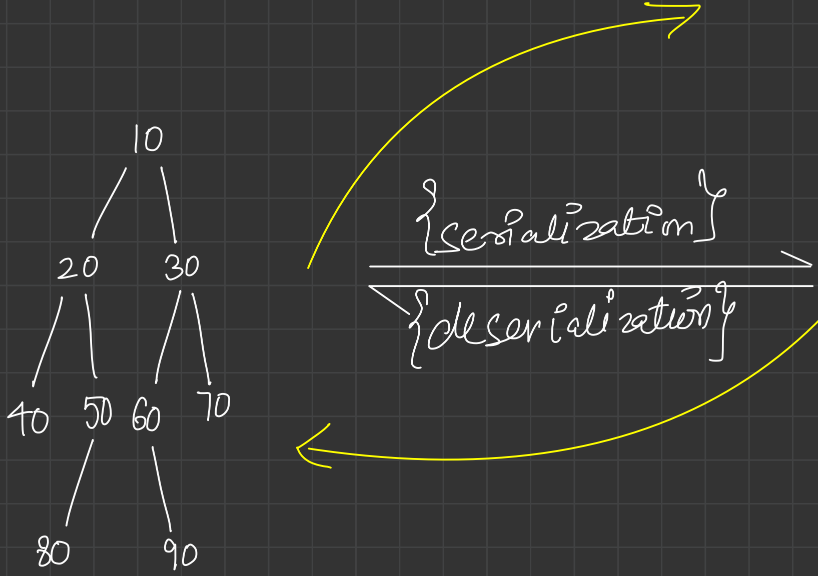
{lst, root, Rst}

faith: Constructs the tree from pre & In order.

construct(int pre[], int In[])

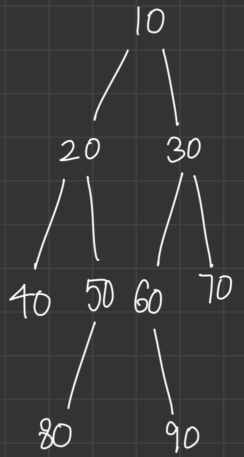
Serialize and deserialize

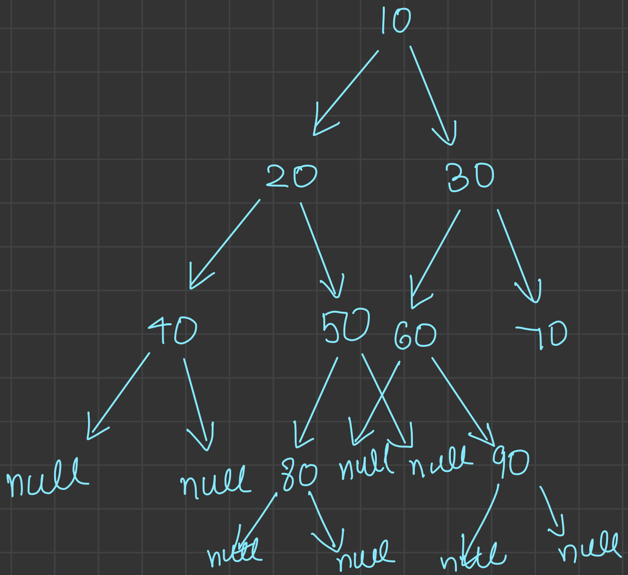
“ 10, 20, 40, 50, 80, 30, 60, 90, 70 $ 40, 30 . . . - ”



BT

String

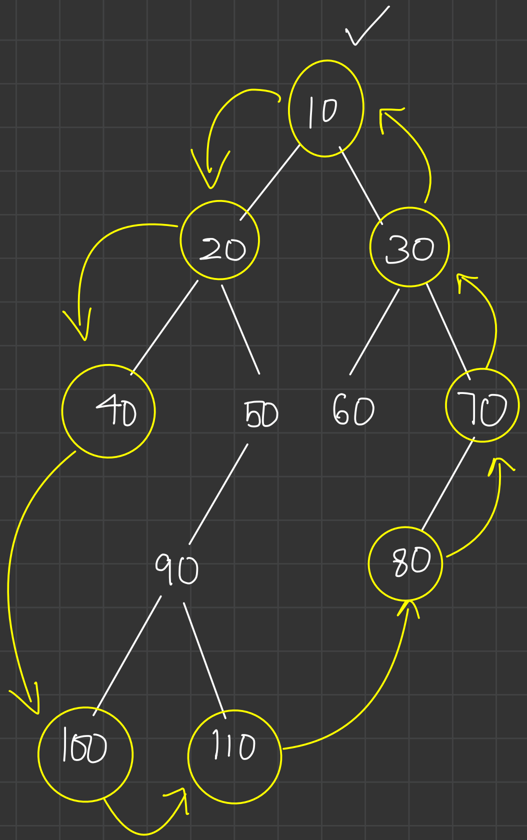




preOrder o

“10, 20, 40, null, null, 50, 80, null, null, null, 30, 60, null, 90, null, null, 70, null, null”

Boundry Tree.



- root Node
- left wall
- leaf Nodes
- right wall