preorder/postorder/inorder without recursion

```
class Pair:
    def __init__(self,node,state):
        self.node = node
        self.state = state
class Solution:
    # @param A : root node of tree
    # @return a list of integers
    def preorderTraversal(self, root):
        ans = []
        stack = [Pair(root,1)]
        while len(stack)>0:
            top = stack[-1]
            if top.state==1:
            #Preorder
                ans.append(top.node.val)
                top.state+=1
                if top.node.left:
                    stack.append(Pair(top.node.left,1))
            elif top.state==2:
            #Inorder
                top.state+=1
                if top.node.right:
                    stack.append(Pair(top.node.right,1))
            else:
            #Postorder
                stack.pop()
        return ans
```