# LeetCode 145

## Description

Given a binary tree, return the postorder traversal of its nodes' values.

For example:
Given binary tree [1,null,2,3],

1
\
2
/
3

return [3,2,1].

Note: Recursive solution is trivial, could you do it iteratively?

## Thought

Use stack(last in first out)

## Solution

```
public List<Integer> postorderTraversal(TreeNode root) {
    List<Integer> list = new ArrayList<>();
    if(root == null) return list;
    Stack<TreeNode> stack = new Stack<>();
    stack.push(root);
    while(!stack.empty()){
        root = stack.pop();
        list.add(0, root.val);
        if(root.left != null) stack.push(root.left);
        if(root.right != null) stack.push(root.right);
    }
    return list;
}
```

## Takeaways

- generalize the model for inorder, preorder, postorder traversal of binary tree.