145. Binary Tree Postorder Traversal

Description (?tab=Description) Submission (?tab=Submission) Solutions (?tab=Solutions) Add to List (?tab=List)

Total Accepted: **128411** Total Submissions: **331959** Difficulty: **Hard** Contributors: **Admin**

Notes

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

For example:

Given binary tree `{1,#,2,3}`,

```
   1
    \
     2
    /
   3
```

return `[3,2,1]`.

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

Subscribe (/subscribe/) to see which companies asked this question.

Show Tags

Show Similar Problems

Have you met this question in a real interview? Yes No

Discuss (https://discuss.leetcode.com/category/153) Pick One (/problems/random-one-question/) Editorial Solution

Python

```
# Definition for a binary tree node.
# class TreeNode(object):
#     def __init__(self, x):
#         self.val = x
#         self.left = None
#         self.right = None

class Solution(object):
    def postorderTraversal(self, root):
        """
        :type root: TreeNode
        :rtype: List[int]
        """
        if root is None:
            return []

        s1=[]
        s2=[]
        s1.append(root)

        while s1:
            x=s1.pop()
            s2.append(x)
            if x.left:
                s1.append(x.left)
            if x.right:
                s1.append(x.right)

        traversal=[]
        while s2:
            node=s2.pop()
            traversal.append(node.val)
        return traversal

```

Custom Testcase □

Contribute Testcase

Shortcut: Ctrl + enter

Run Code Submit Solution

Send Feedback (mailto:admin@leetcode.com?subject=Feedback)