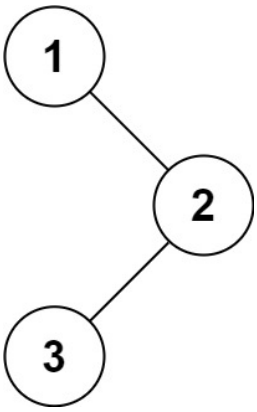
94. Binary Tree Inorder Traversal

Easy 8346 383 Add to List Share

Given the `root` of a binary tree, return *the inorder traversal of its nodes' values*.

Example 1:




```
Input: root = [1,null,2,3]
Output: [1,3,2]
```

Example 2:

```
Input: root = []
Output: []
```

Example 3:

```
Input: root = [1]
Output: [1]
```

Constraints:

- The number of nodes in the tree is in the range `[0, 100]`.
- `-100 <= Node.val <= 100`

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

```
 8    *     Tree
      left(nullpt
 9    *     Tree
      *left, Tree
      val(x), lef
      {}
10    * };
11    */
12 ▾ class Solut
13    public:
14 ▾     vector<
      inorderTrav
      {
15            vec
16            sta
17            Tre
      *node;
18 ▾          whi
      !st.empty()
19 ▾
20
21
      >left;
22
23 ▾
24
25
26
       res.push_b
27
      >right;
28
29            }
30            ret
31        }
32
33    };
```

Testcase Run Code Re

Accepted Runti

Your input [1,

Output [1,

Expected [1,

Console Use