

[Description](#)
[Solution](#)
[Discuss \(999+\)](#)
[Submissions](#)
i C++

105. Construct Binary Tree from Preorder and Inorder Traversal

Medium

9656

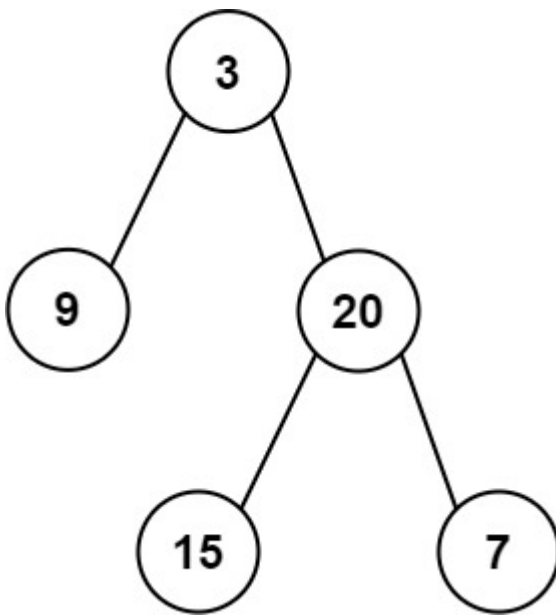
261

Add to List

Share

Given two integer arrays `preorder` and `inorder` where `preorder` is the preorder traversal of a binary tree and `inorder` is the inorder traversal of the same tree, construct and return *the binary tree*.

Example 1:



Input: `preorder = [3,9,20,15,7]`, `inorder = [9,3,15,20,7]`

Output: `[3,9,20,null,null,15,7]`

Example 2:

Input: `preorder = [-1]`, `inorder = [-1]`

Output: `[-1]`

Constraints:

- `1 <= preorder.length <= 3000`
- `inorder.length == preorder.length`
- `-3000 <= preorder[i], inorder[i] <= 3000`
- `preorder` and `inorder` consist of **unique** values.
- Each value of `inorder` also appears in `preorder`.

```

1  TreeNode* left(nullptr), right(nullptr), int val):
2      :val(val), left(left), right(right) {}
3
4  // Definition for a binary tree node.
5  struct TreeNode {
6      int val;
7      TreeNode *left;
8      TreeNode *right;
9      TreeNode() : val(0), left(nullptr), right(nullptr) {}
10     TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}
11     TreeNode(int x, TreeNode *left, TreeNode *right) : val(x), left(left), right(right) {}
12 };
13
14 class Solution {
15 public:
16     TreeNode* buildTree(vector<int> &preorder, vector<int> &inorder) {
17         if (preorder.empty()) return nullptr;
18         int root = preorder[0];
19         int pivot = -1;
20         for (int i = 0; i < inorder.size(); i++) {
21             if (inorder[i] == root) {
22                 pivot = i;
23             }
24         }
25         TreeNode* node = new TreeNode(root);
26         node->left = buildTree(vector<int> &preorder, vector<int> &inorder, 0, pivot);
27         node->right = buildTree(vector<int> &preorder, vector<int> &inorder, pivot+1, inorder.size());
28         return node;
29     }
30 };
    
```

Testcase Run Code Re

Accepted Runtime

Your input [3, 9, 20, 15, 7]

Output [3, 9, 20, 15, 7]

Expected [3, 9, 20, 15, 7]

[Problems](#)
[Pick One](#)
[Prev](#)

105/2340

[Next](#)
[Example cases](#)

?

[Run C](#)