LC 105. Construct Binary Tree from Preorder and Inorder Traversal

Question

Given preorder and inorder traversal of a tree, construct the binary tree.

Note:
You may assume that duplicates do not exist in the tree.

For example, given

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

Return the following binary tree:

```
    3
   / \
  9  20
    /  \
   15   7
```

Solution

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

class Solution:
    def buildTree(self, preorder: List[int], inorder: List[int]) -> TreeNode:
        #Solution
        if not preorder or not inorder:
            return None
        rootVal = preorder.pop(0)
        rootIndex = inorder.index(rootVal)
        root = TreeNode(rootVal)
        root.left = self.buildTree(preorder, inorder[:rootIndex])
        root.right = self.buildTree(preorder, inorder[rootIndex + 1:])
        return root
```