105. Construct Binary Tree from Preorder and Inorder Traversal

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

```
class Solution {
    public TreeNode buildTree(int[] preorder, int[] inorder) {
        Map<Integer, Integer> map = new HashMap<Integer, Integer>();

        for (int i = 0; i < inorder.length; i++) {
            map.put(inorder[i], i);
        }

        TreeNode root = buildTreePreIn(preorder, 0, preorder.length-1,
inorder, 0, inorder.length-1, map);
        return root;
    }

    private TreeNode buildTreePreIn(int[] preorder, int preStart,
            int preEnd, int[] inorder, int inStart, int inEnd, Map<Integer,
Integer> map) {

        if (preStart > preEnd || inStart > inEnd)
            return null;

        TreeNode root = new TreeNode(preorder[preStart]);
        int inRoot = map.get(root.val);
        int numsLeft = inRoot - inStart;

        root.left = buildTreePreIn(preorder, preStart+1, preStart + numsLeft,
                    inorder, inStart, inRoot-1, map);
        root.right = buildTreePreIn(preorder, preStart + numsLeft + 1, preEnd,
                    inorder, inRoot+1, inEnd, map);

        return root;
    }
}
```