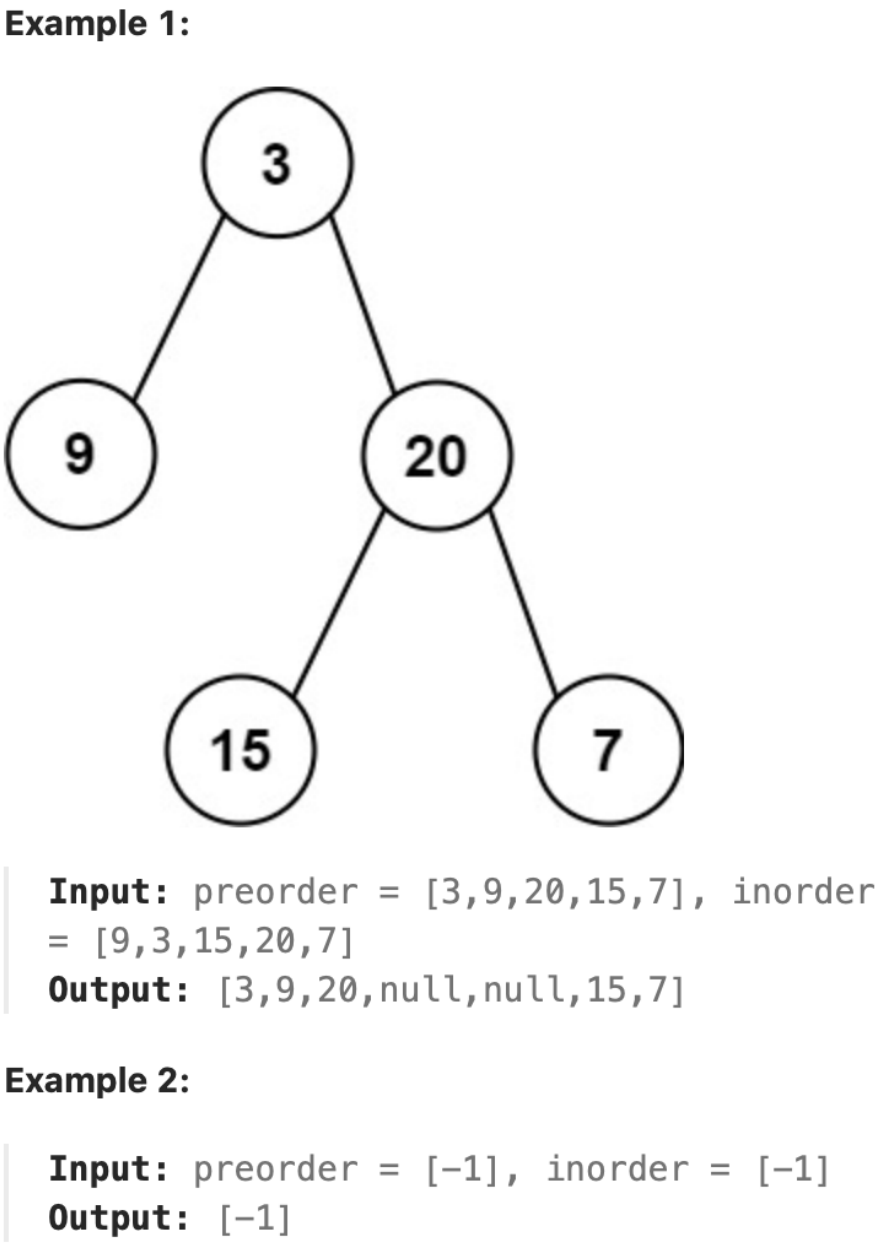
Construct Binary Tree from Preorder and Inorder Traversal

Difficulty	Medium
Category	Tree
Question	https://leetcode.com/problems/construct-binary-tree-from-preorder-and-inorder-traversal/
Solution	https://www.youtube.com/watch?v=ihj4IQGZ2zc
Status	Done

Question

> 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]

Solution

```
class Solution:
    def buildTree(self, preorder: List[int], inorder: List[int]) -> Optional[TreeNode]:
        # Base case: If either preorder or inorder list is empty, return None.
        if not preorder or not inorder:
            return None

        # Create a root node with the first value from the preorder list.
        root = TreeNode(preorder[0])

        # Find the index of the root's value in the inorder list, splitting it into left and right subtrees.
        mid = inorder.index(preorder[0])

        # Recursively build the left subtree using the portion of preorder and inorder lists for the left subtree.
        root.left = self.buildTree(preorder[1:mid + 1], inorder[:mid])

        # Recursively build the right subtree using the portion of preorder and inorder lists for the right subtree.
        root.right = self.buildTree(preorder[mid + 1:], inorder[mid + 1:])

        # Return the root of the constructed binary tree.
        return root
```

Explanation