给定一个有 N 个节点的二叉树，每个节点都有一个不同于其他节点且处于 {1, ..., N} 中的值。

通过交换节点的左子节点和右子节点，可以翻转该二叉树中的节点。

考虑从根节点开始的先序遍历报告的 N 值序列。将这一 N 值序列称为树的行程。

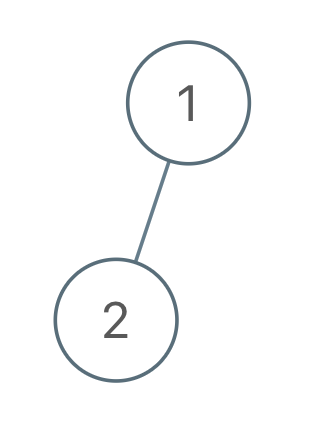
（回想一下，节点的先序遍历意味着我们报告当前节点的值，然后先序遍历左子节点，再先序遍历右子节点。）

我们的目标是翻转**最少的**树中节点，以便树的行程与给定的行程 voyage 相匹配。

如果可以，则返回翻转的所有节点的值的列表。你可以按任何顺序返回答案。

如果不能，则返回列表 [-1]。

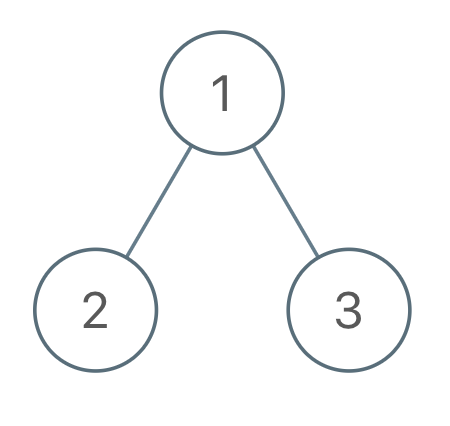
**示例 1：**

****

**输入：**root = [1,2], voyage = [2,1]

**输出：**[-1]

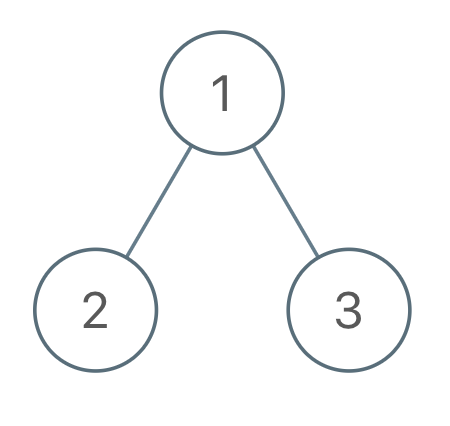
**示例 2：**

****

**输入：**root = [1,2,3], voyage = [1,3,2]

**输出：**[1]

**示例 3：**

****

**输入：**root = [1,2,3], voyage = [1,2,3]

**输出：**[]

**提示：**

1. 1 <= N <= 100