

CS203B DSAA B Assignment3

Q1 二叉搜索树的判断和删除

问题描述

给定一个二叉树的根节点 root，判断该树是否是一个二叉搜索树，若不是则直接输出no，若是的话进行下一步操作：1个key值，删除二叉搜索树中的 key 对应的节点，并保证二叉搜索树的性质不变。返回二叉搜索树（有可能被更新）的根节点root。

Input

第一行：root数组的长度

$1 \leq n \leq 10000$

第二行，root数组的数据

$0 \leq a[i] \leq 10^9$, $a[i]=-1$ 代表该结点为空

数据都为整数，且保证没有重复元素

第三行：k

$1 \leq k \leq 100$

后k行：对应的key值

$1 \leq key \leq 10^9$

注意：如果该树中不包含对应的key值就证明不用删除

Output

第一行：该树是否为二叉搜索树，若是输出Yes,若不是输出No

第二行：若是，输出删除结点后的root数组

Sample

Input

```
7
5 3 6 2 4 -1 7
2
3
8
```

output

```
Yes
5 4 6 2 -1 -1 7
```

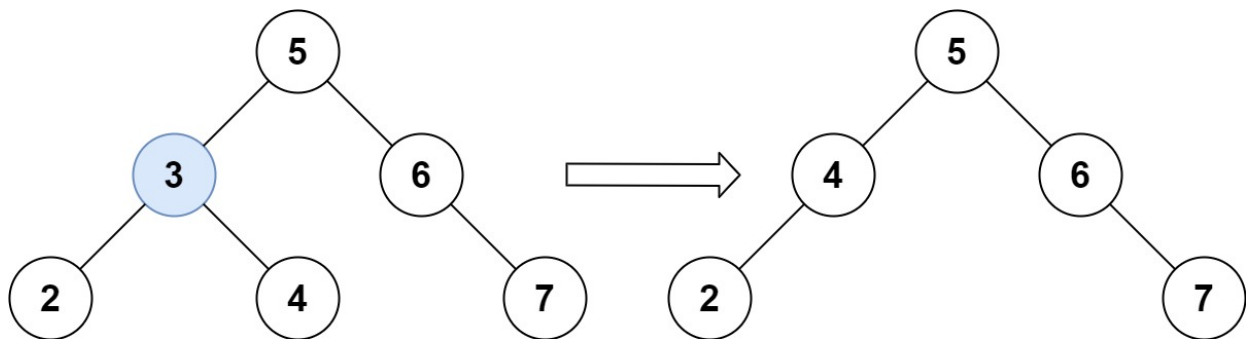
Input

```
7
5 3 6 2 4 -1 1
2
3
8
```

output

```
No
```

Explanation



结点编号规则：根结点被存放在`root[1]`中，`root[k]`的左儿子被存放在`root[k+k]`，`root[k]`的右儿子被存放在`root[k+k+1]`，在本题中5是`root[1]`，他的左右儿子是`root[2]`和`root[3]`，以此类推

删除二叉搜索树中的点：**必须保证删除完后，这颗树仍然是二叉搜索树，而不是简单的把他设置为空**！如这个样例，在删除3后，你应该将3的**后继结点**作为5的新儿子！（注意，只有在被删节点同时有左右儿子的时候才需要找后继节点，思考一下）

后继节点：简单来讲就是在中序遍历中某节点的下一个节点。如果一个节点有右孩子，那么后继节点就是该节点的右子树的最左叶节点

对于样例1，删除后的树型如右图，这时4的右儿子为空，6的左儿子为空，所以按照编号规则，答案是：5 4 6 2 -1 -1 7

对于样例2，若7的值被修改成了1,那么根的右子树出现了比他自身更小的值，就不是二叉搜索树了，直接输出No

Requirement

1. 你需要在 `tree.java` 中实现解决问题的算法。

2. 请在文件中**解释你的算法并正确分析它的时间复杂度**，这个文件可以是 `.txt`、`.doc`、`.md` 或任何你喜欢的文件。(可以用中文写) 如果复杂度分析与你写的算法不相符，你会损失一些分数。

3. 请生成随机的新测试数据来测试你的程序并证明它的正确性。

测试代码需要使用另一个 `.java` 文件编写，而不能在 `tree.java` 中实现，该代码将用来生成测试数据。

你应该**能够正确地进行二叉搜索树以及错误的二叉搜索树两种不同的数据的测试**，无论是直接生成两种数据还是生成正确的后手动将其改错

你可以使用读取文件的方式进行测试，**但在提交时请将代码内的输入方式改为标准输入，以便于批改时对代码正确性进行检验。**

4. 请提交压缩文件，文件内包含代码文件 `tree.java`，生成测试数据的代码文件和解释文档，压缩包命名格式为 `学号.zip`。压缩包命名错误将会分数*0.8。