03-树1 树的同构（25 分）

给定两棵树T1和T2。如果T1可以通过若干次左右孩子互换就变成T2，则我们称两棵树是“同构”的。例如图1给出的两棵树就是同构的，因为我们把其中一棵树的结点A、B、G的左右孩子互换后，就得到另外一棵树。而图2就不是同构的。

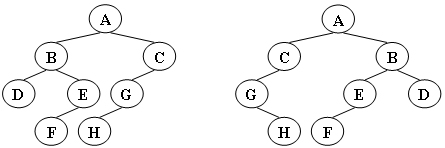


图1

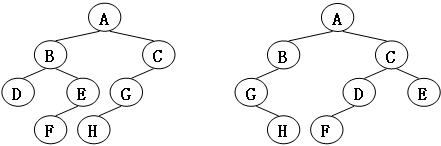


图2

现给定两棵树，请你判断它们是否是同构的。

输入格式:

输入给出2棵二叉树树的信息。对于每棵树，首先在一行中给出一个非负整数*N* (≤10)，即该树的结点数（此时假设结点从0到*N*−1编号）；随后*N*行，第*i*行对应编号第*i*个结点，给出该结点中存储的1个英文大写字母、其左孩子结点的编号、右孩子结点的编号。如果孩子结点为空，则在相应位置上给出“-”。给出的数据间用一个空格分隔。注意：题目保证每个结点中存储的字母是不同的。

输出格式:

如果两棵树是同构的，输出“Yes”，否则输出“No”。

输入样例1（对应图1）：

8

A 1 2

B 3 4

C 5 -

D - -

E 6 -

G 7 -

F - -

H - -

8

G - 4

B 7 6

F - -

A 5 1

H - -

C 0 -

D - -

E 2 -

输出样例1:

Yes

输入样例2（对应图2）：

8

B 5 7

F - -

A 0 3

C 6 -

H - -

D - -

G 4 -

E 1 -

8

D 6 -

B 5 -

E - -

H - -

C 0 2

G - 3

F - -

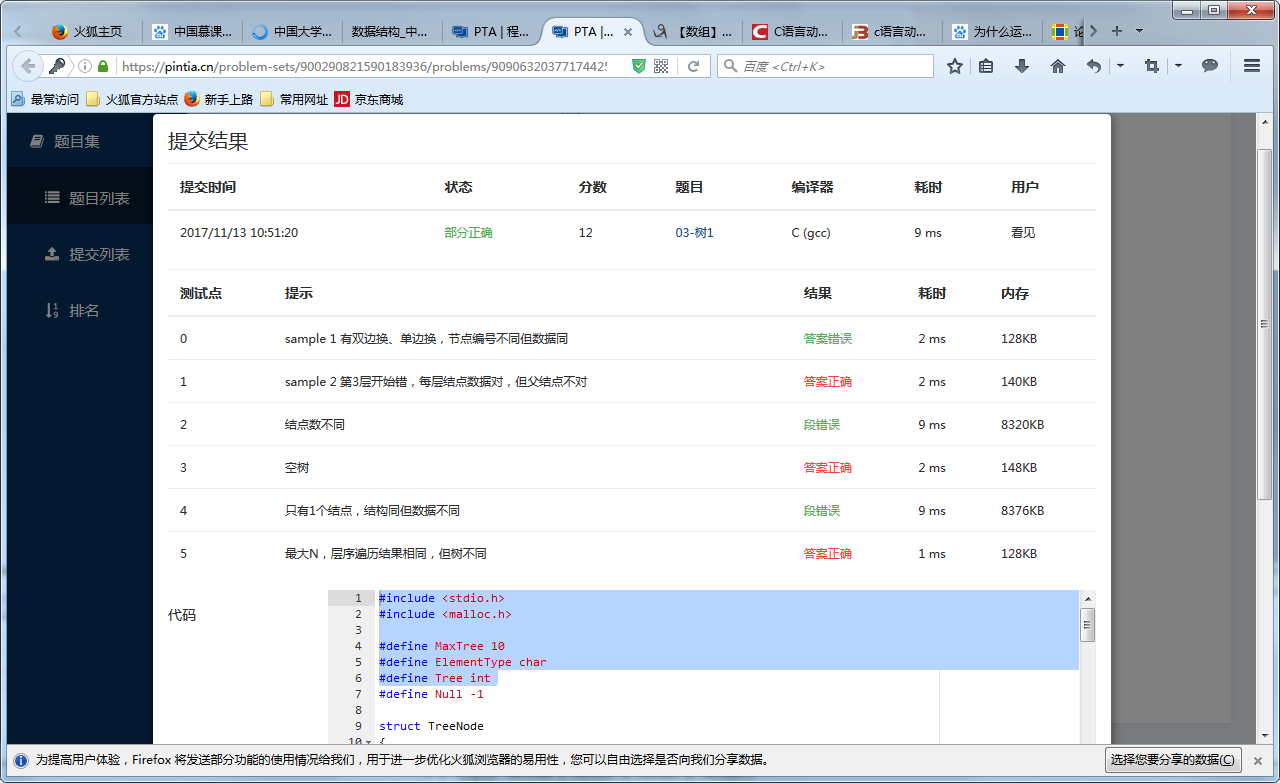
A 1 4

输出样例2:

No

# 评分标准





#### 03-树2 List Leaves（25 分）

Given a tree, you are supposed to list all the leaves in the order of top down, and left to right.

### Input Specification:

Each input file contains one test case. For each case, the first line gives a positive integer N (≤10) which is the total number of nodes in the tree -- and hence the nodes are numbered from 0 to N−1. Then N lines follow, each corresponds to a node, and gives the indices of the left and right children of the node. If the child does not exist, a "-" will be put at the position. Any pair of children are separated by a space.

### Output Specification:

For each test case, print in one line all the leaves' indices in the order of top down, and left to right. There must be exactly one space between any adjacent numbers, and no extra space at the end of the line.

### Sample Input:

8

1 -

- -

0 -

2 7

- -

- -

5 -

4 6

### Sample Output:

4 1 5