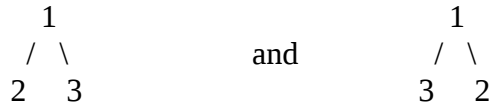
Special Binary Trees

A special binary tree is a binary tree such that the left and right child of every vertex is an ordered pair. The implication of such constraint is that



are different. Each vertex has a unique number.

Given N+1 special binary trees T1, T2, T3 … TN and Q.
The set of T1, T2, T3 … TN is such that **no tree in the set is a sub-tree of any other tree in the set. Also no 2 of them are identical**.
You need to tell if Q can be made constructed T1, T2, T3 …. TN, by doing join operations any number of times.

A join operation is defined as:

- Pick the root of any tree in the set and hook it to any vertex of another tree such that the resulting tree is also a binary tree. (i.e. the vertex you hook the root of the first tree to, must have less than 2 children)
- Now the two trees merge into a single tree and the size of the set containing the trees reduce by one (You can't perform a join operation if size of the set is 1).

Constraints:

1 <= Number of vertices in Ti <= 100 for all i in [1, N]
1 <= Number of vertices in Q <= 2000
Sum of vertices of all Ti <= 5000
Time = 2 sec

Input:

Next, definition of Q.
Now definition of N special trees follow, i'th definition is for Ti.

For definition a special tree:
First line contains V, the number of vertices in that special tree.
Next line contain pre-order traversal of the tree (space separated numbers).
Next line contain in-order traversal of the tree (space separated numbers).
The last line of input is denoted by a “-1”.

Output:

“Yes” if Q can be constructed else “No”.

Note: Output these without quotes.

Sample Input 1:

```
10
1 2 5 4 7 8 10 6 3 9
5 2 7 10 8 4 6 1 9 3
2
1 2
1 2
-1
```

Sample Output 1:

```
No
```

Sample Input 2:

```
10
1 5 6 10 7 2 3 4 9 8
10 6 7 5 1 9 4 3 2 8
2
20 59
59 20
1
10
10
2
6 7
6 7
1
5
5
2
4 9
9 4
1
3
3
2
2 8
2 8
1
1
1
2
13 14
14 13
-1
```

Sample Output 2:

```
Yes
```