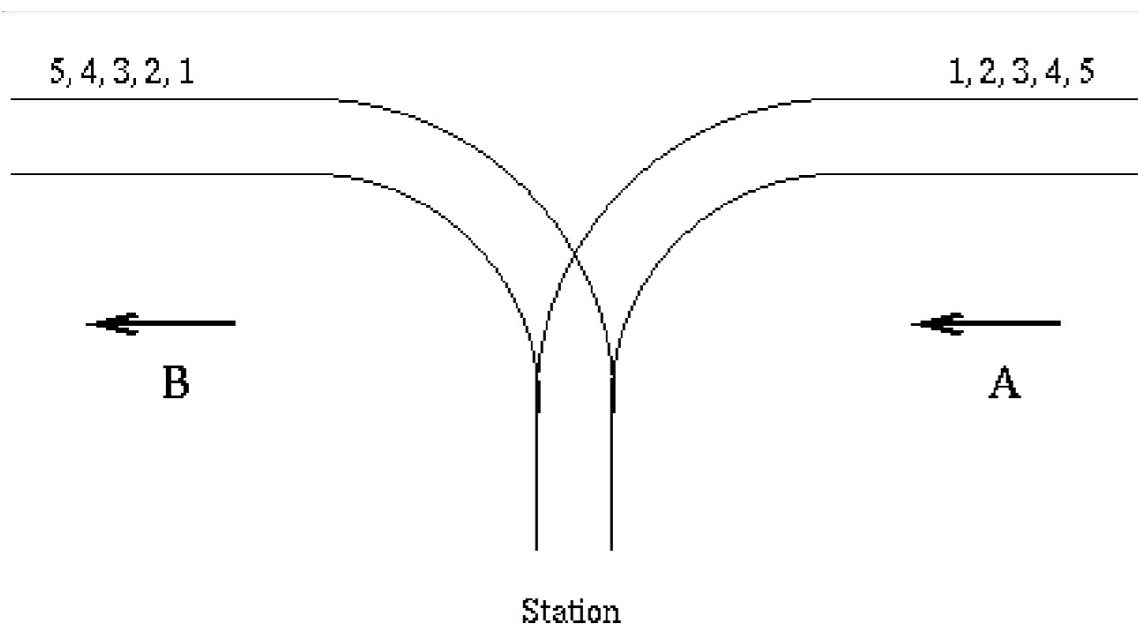


Problem 4: Rails

(Easy-Medium)

(Adapted from UVa 00514)

There is a famous railway station in PopPush City. The country there is incredibly hilly. The station was built in the last century. Unfortunately, funds were extremely limited at that time. It was possible to establish only a surface track. Moreover, it turned out that the station could be only a dead-end one (see figure) and due to lack of available space it could have only one track.



The local tradition is that every train arriving from the direction A continues in the direction B with coaches reorganized in some way. Assume that the train arriving from the direction A has N coaches numbered in increasing order $1, 2, \dots, N$. The chief for train reorganizations must know whether it is possible to marshal coaches continuing in the direction B so that their order will be a_1, a_2, \dots, a_N . Help him and write a program that decides whether it is possible to get the required order of coaches.

You may assume the following.

- Single coaches can be disconnected from the train before they enter the station and that they can move themselves until they are on the track in the direction B.
- There can be as many coaches as necessary in the station at any time. However, once a coach has entered the station it cannot return to the track in the direction A and also once it has left the station in the direction B it cannot return back to the station.

Input Format

The input consists of blocks of lines. Each block except the last describes one train and possibly more requirements for its reorganization. In the first line of the block there is the integer N described above. In each of the next lines of the block there is a permutation of $1, 2, \dots, N$. The last line of the block contains just `0`.

The last block consists of just one line containing `0`.

Constraints

- There are at most 500 blocks of lines
- $1 \leq N \leq 1000$

The time limit for this problem is 2 seconds.

Output Format

The output should contain the lines corresponding to the lines with permutations in the input. A line of the output contains `Yes` if it is possible to marshal the coaches in the order required on the corresponding line of the input file. Otherwise it contains `No`. In addition, there is one empty line after the lines corresponding to one block of the input file. There is no line in the output file corresponding to the last 'null' block of the input file (that is, the final block consisting only of `0`).

Sample Input 1

```
5
1 2 3 4 5
5 4 1 2 3
2 1 5 4 3
0
0
```

Sample Output 1

```
Yes
No
Yes
```

Explanation 1

This test case sees one train of length 5 entering the station.

- The first case asks whether the train can leave in the order `1 2 3 4 5`. If we ask every incoming coach to immediately leave the station we can achieve this order.
- The second case asks whether the train can leave in the order `5 4 1 2 3`. This is impossible.
- The final case asks whether the train can leave in the order `2 1 5 4 3`. If we allow the first two coaches to enter the station **before** letting them leave, the first two coaches to leave will be `2 1`. A similar logic applies for the arrangement `5 4 3`. Thus it is possible.

Sample Input 2

```
5
1 2 3 4 5
5 4 3 2 1
0
6
1 2 3 4 5 6
6 5 4 3 2 1
0
7
7 6 5 4 3 1 2
0
0
```

Sample Output 2

```
Yes
Yes
```

```
Yes
Yes
```

```
No
```