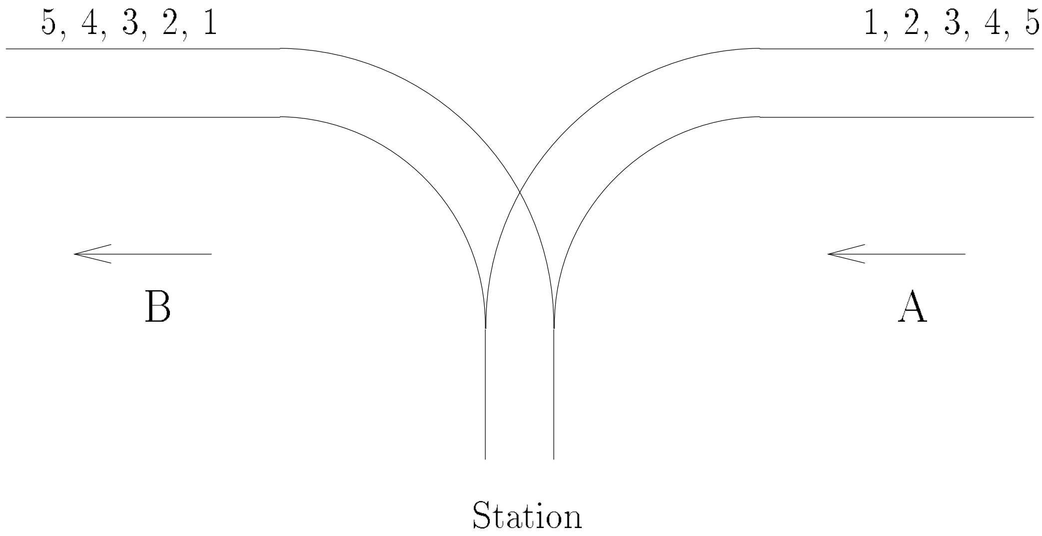
514 Rails

There is a famous railway station in PopPush City. Country there is incredibly hilly. The station was built in last century. Unfortunately, funds were extremely limited 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 picture) 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 $N \leq 1000$ coaches numbered in increasing order $1, 2, \ldots, N$. 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 $a_1.a_2, \ldots, a_N$. Help him and write a program that decides whether it is possible to get the required order of coaches. You can assume that 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. You can also suppose that at any time there can be located as many coaches as necessary in the station. But 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

The input file 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 $1, 2, \ldots, N$ The last line of the block contains just 0.

The last block consists of just one line containing 0.

Output

The output file contains the lines corresponding to the lines with permutations in the input file. A line of the output file 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.

Sample Input

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

Sample Output

```
Yes
No

Yes
```