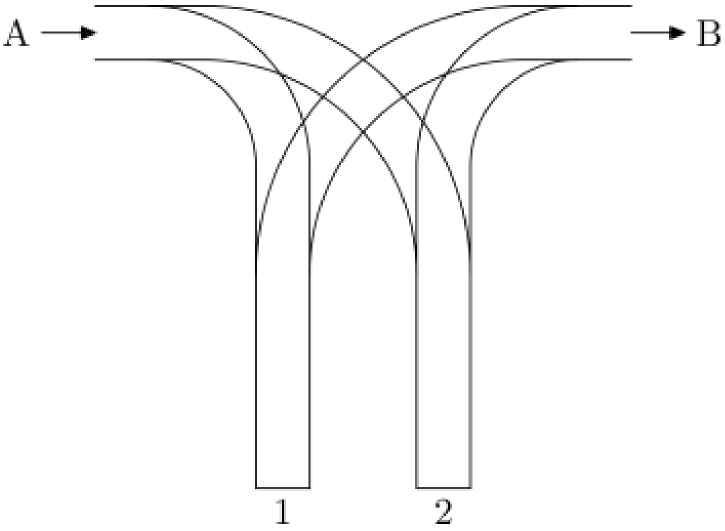
Railway

A railroad siding consists of two (dead-end) sidetracks 1 and 2. The siding is entered by track A, and left by track B.



There are N cars on track A, numbered from 1 to N. They are arranged in such a way that they enter the siding in the order $a_1, a_2, a_3, ..., a_N$. The cars are to be transferred to the siding, so that they leave it by track B in the order 1, 2, 3, ..., N. Each car is to be transferred once from track A to one of the sidetracks 1 or 2, and later (possibly after some transfers of the remaining cars) once from that sidetrack to the track B. The sidetracks are long enough to store even the longest trains, so there is no need to worry about their capacity.

Input

The first line of the standard input holds one integer N ($1 <= N <= 10^5$) that denotes the number of cars for transfer. The second line stores the numbers $a_1, a_2, a_3, ..., a_N$ that are a permutation of 1, 2, 3, ..., N (i.e., each a_i belongs to {1, 2, 3, ..., N}, and all these numbers are unique), separated by single spaces.

Output

The first line of the standard output should contain the word TAK (*yes* in Polish) if there is a way of transferring the cars so that they enter track B in the order 1, 2, 3, ..., N, or the word NIE (*no* in Polish) if it is impossible. If the answer is TAK, the second line should give, separated by single spaces, the numbers of sidetracks (1 or 2) to which successive cars $a_1, a_2, a_3, ..., a_N$ are moved in a correct transfer. If there are several ways of making the transfer, choose lexicographically smallest.

Example

For the input data:

```
4
1 3 4 2
```

the correct result is:

```
TAK
1 1 2 1
```

And for the input:

```
4
2 3 4 1
```

the correct result is:

```
NIE
```