A. Nicholas and Permutation

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Nicholas has an array a that contains n **distinct** integers from 1 to n. In other words, Nicholas has a permutation of size n.

Nicholas want the minimum element (integer 1) and the maximum element (integer n) to be as far as possible from each other. He wants to perform exactly one swap in order to maximize the distance between the minimum and the maximum elements. The distance between two elements is considered to be equal to the absolute difference between their positions.

Input

The first line of the input contains a single integer n ($2 \le n \le 100$) — the size of the permutation.

The second line of the input contains n distinct integers $a_1, a_2, ..., a_n$ ($1 \le a_i \le n$), where a_i is equal to the element at the i-th position.

Output

Print a single integer — the maximum possible distance between the minimum and the maximum elements Nicholas can achieve by performing exactly one swap.

Examples

input Copy

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

output Copy

```
3
```

input Copy

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

output Copy

```
6
```

input Copy

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

output Copy

```
5
```

Note

In the first sample, one may obtain the optimal answer by swapping elements 1 and 2.

In the second sample, the minimum and the maximum elements will be located in the opposite ends of the array if we swap 7 and 2.

In the third sample, the distance between the minimum and the maximum elements is already maximum possible, so we just perform some unnecessary swap, for example, one can swap 5 and 2.