

Encryption by Reversal

Time Limit: 2.0s **Memory Limit:** 64M

John von Neumann was a Hungarian-American scholar who worked on developing the world's first nuclear weapons during World War II. He is highly regarded as one of the greatest mathematicians of his time. One day, John von Neumann is testing a new algorithm he created called "Mergesort" using a set of numbers he generated. Because he wants to keep his work on Mergesort secret, John von Neumann devises a way to encrypt his numbers. Starting from the original sorted set of numbers, he selects some of them and reverses the order of the digits. Given the encrypted set, determine the minimum amount of numbers that must be reversed to obtain a correctly sorted set of numbers.

Input Specification

The first line of input contains n ($1 \leq n \leq 10^5$) representing the quantity of unique numbers in the set. The next line contains n space separated integers n_i ($1 \leq n_i \leq 10^9$) representing the encrypted set of numbers, ordered from least to greatest once decrypted.

Output Specification

Print the minimum number of reversals needed such that the number list will be in ascending order.

Sample Input

```
4
43 39 471 250
```

Sample Output

```
2
```

Explanation for Sample

The first number and the third number can be reversed resulting in the list `[34, 39, 174, 250]`.