Negsort

SEASON 7 – ROUND SIX

CODEIT.BG

Embrace The Challenge

Lora recently learned several sorting algorithms, but none of them appealed to her. Now she wants to come up with her own sorting algorithm, but since all the algorithms based on moving elements are already invented, she wants to make a algorithm based on changing elements. Since arbitrarily changing elements would obviously be impractical, she allows only changing the sign of the elements.

Formally, Lora has a sequence **A** of **N** integers and at each step she selects an integer and changes its sign (i.e. multiplies it by -1). Now the girl wonders what is the minimum number of steps that are needed to get a non-decreasing sequence, and whether it is possible at all. As usual, your task is to satisfy her curiosity.

Input

The first line of the input file `negsort.in` contains a single integer **N** – the number of integers in the sequence.

The next line contains N space-separated integers - the sequence **A.**

Output

On a single line in the output file `negsort.out` print the minimum number of steps of the type described, which are needed to get a sequence of non-decreasing integers. If this is impossible, print -1.

Constraints

$1 \leq N \leq 100\ 000$

$-10^9 \leq A_i \leq 10^9$

Time limit: 0.5 sec

Memory limit: 256 MB

Sample test

Input (negsort.in)	Output (negsort.out)
7 -2 -1 1 1 -1 -1 0	2
3 1 5 7	0
4 1 -2 1 2	-1

Clarifications

In the first example, it is optimal to change the sign of the third and fourth integers, obtaining the sorted sequence "-2 -1 -1 -1 -1 -1 0".

In the second example, the given sequence is already sorted.

In the third example, the sequence can’t be sorted using only the given operation.