

Negsort

СЕЗОН 7 – ШЕСТИ РУНД



Наскоро Лора научи няколко алгоритъма за сортиране, но никой от тях не ѝ допадна. Сега тя иска да измисли свой собствен алгоритъм за сортиране, но понеже всички алгоритми базирани на разместване на елементи са вече измислени, тя иска да направи алгоритъм базиран на промяна на елементи. Тъй като произволна промяна на елементи очевидно не би била практична, тя си позволява само смяна на знака им.

Формално, Лора има редица **A** от **N** числа и на всяка стъпка тя избира число и сменя знака му (т.е. го умножава по -1). Сега момичето се чуди какъв е минималният брой стъпки, които са нужни, за да се получи редица от ненамаляващи числа, и дали това изобщо е възможно. Както обикновено, ваша е задачата да задоволите любопитството ѝ.

Вход

От първия ред на файла `negsort.in` се въвежда числото **N** – броят числа в редицата.

От втория ред се въвеждат **N** числа разделени с интервал – редицата **A**.

Изход

На единствен ред в изходния файл `negsort.out` отпечатайте минималния брой стъпки от описания вид, които са нужни за да получим редица от ненамаляващи числа. Ако това не е възможно, изведете -1.

Ограничения

$$1 \leq N \leq 100\,000$$

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

Ограничение за време: 0.5 сек

Ограничение за памет: 256 MB

Примерен тест

Вход (<code>negsort.in</code>)	Изход (<code>negsort.out</code>)
7 -2 -1 1 1 -1 -1 0	2
3 1 5 7	0
4 1 -2 1 2	-1

Negsort

СЕЗОН 7 – ШЕСТИ РУНД



Пояснения

В първият пример е оптимално да променим знака на третото и четвъртото число, получавайки сортираната редица “-2 -1 -1 -1 -1 -1 0”

Във втория пример дадената редица е вече сортирана.

В третия пример редицата не може да бъде сортирана използвайки само дадената операция.