

Mega Inversion

The n^2 upper bound for any sorting algorithm is easy to obtain: just take two elements that are misplaced with respect to each other and swap them. Conrad conceived an algorithm that proceeds by taking not two, but three misplaced elements. That is, take three elements $a_i > a_j > a_k$ with $i < j < k$ and place them in order a_k, a_j, a_i . Now if for the original algorithm the steps are bounded by the maximum number of inversions

$\frac{n(n-1)}{2}$, Conrad is at his wits' end as to the upper bound for such triples in a given sequence. He asks you to write a program that counts the number of such triples.



Input

The first line of the input is the length of the sequence, $1 \leq n \leq 10^5$. The next line contains the integer sequence a_1, a_2, \dots, a_n . You can assume that all $a_i \in [1, n]$.

Output

Output the number of inverted triples.

Sample Input 1

```
3
1 2 3
```

Sample Output 1

```
0
```

Sample Input 2

```
4
3 3 2 1
```

Sample Output 2

```
2
```