B. Arpa's obvious problem and Mehrdad's terrible solution

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

There are some beautiful girls in Arpa's land as mentioned before.

Once Arpa came up with an obvious problem:

Given an array and a number x, count the number of pairs of indices i, j $(1 \le i < j \le n)$ such that , where is bitwise `xor` operation (see notes for explanation).

Immediately, Mehrdad discovered a terrible solution that nobody trusted. Now Arpa needs your help to implement the solution to that problem.

Input

First line contains two integers n and x $(1 \le n \le 10^5, 0 \le x \le 10^5)$ — the number of elements in the array and the integer x.

Second line contains n integers $a_1, a_2, ..., a_n$ $(1 \le a_i \le 10^5)$ — the elements of the array.

Output

Print a single integer: the answer to the problem.

Examples

input
2 3 1 2

output
1

input
6 1 5 1 2 3 4 1

output
2

Note

In the first sample there is only one pair of $i = 1$ and $j = 2$. so the answer is 1.

In the second sample the only two pairs are $i = 3, j = 4$ (since) and $i = 1, j = 5$ (since).

A bitwise `xor` takes two bit integers of equal length and performs the logical `xor` operation on each pair of corresponding bits. The result in each position is 1 if only the first bit is 1 or only the second bit is 1, but will be 0 if both are 0 or both are 1. You can read more about bitwise `xor` operation here: https://en.wikipedia.org/wiki/Bitwise_operation#XOR.