Xor

SEASON 7 – ROUND SIX

Like all programmers, Ivancho is very intrigued by the xor bitwise operation. In his attempts to learn more about it, he came across the following problem:

Given an array with N elements and a number K, count the pair of indices i, j ($1 \leq i < j \leq N$), such that $A_i \oplus A_j = K$, where $\oplus$ is bitwise xor operation.

As usual, you are asked to help Ivancho by solving the task.

Input

The first line of the input file `xor.in` contains two integers **N** and **K** – the number of elements in the array and the integer K.

The second line contains N integers separated by spaces – the elements of the array.

Output

On the first line of the output file `xor.out` print a single integer - the answer to the problem.

Note: Reading from a file and printing into a file is done by adding an operator for that. You can use the freopen operator by turning on the cstdio library and adding the following two lines to the beginning of your main function: freopen ("xor.in", "r", stdin); freopen ("xor.out", "w", stdout);

Constraints

$1 \leq N \leq 100\ 000$

$0 \leq K \leq 100\ 000$

$1 \leq A_i \leq 100\ 000$

Time limit: 0.5 sec
Memory limit: 256 MB

Sample tests

Input (xor.in)	Output (xor.out)
3 3 1 2 3	1
6 1 1 4 1 5 3 2	2

Xor

SEASON 7 – ROUND SIX

Clarifications

In the first sample the only pair is i = 1, j = 2, because 1 ⊕ 2 = 3 = K.

In the second sample the two pairs are i = 2, j = 4 (because 4 ⊕ 5 = 1 = K) and i = 5, j = 6 (because 2 ⊕ 3 = 1 = K), so the answer is 2.

You can read more about bitwise xor operation here:

https://en.wikipedia.org/wiki/Bitwise_operation#XOR.