Problem E. Amazing Pairs

Time limit: 1 second
Memory limit: 256 megabytes
input: `standard input`
output: `standard output`

You are given two integer n, k, along with a sequence $a_1, a_2, \cdots, a_n$.

A pair $\langle i, j \rangle$ $(i < j)$ if considered as **Amazing Pairs** if $a_i \oplus a_j = k$. Your task is to calculate the count of **Amazing Pairs**. Formally, calculate $\sum_{i=1}^{n} \sum_{j>i}^{n} [a_j \oplus a_j = k]$ where the sign $[a = b] = 1$ if $a = b$, otherwise it's 0.

Here the operator $\oplus$ means **Exclusive Or**(异或). It is a logical operation that is true **if and only if** its arguments differ (one is true, the other is false).

It's a bitwise(按位的) operation. For example, here's the way to calculate $9 \oplus 13$: $1001_{(2)} \oplus 1101_{(2)} = 0100_{(2)} = 4_{(10)}$. So the answer for $9 \oplus 13 = 4$.

Input

The first line contains two integers n, k $(1 \le n, k \le 10^5)$.
The second line contains n integers $a_1, a_2, \cdots, a_n$ $(1 \le a_i \le 10^5)$, indicating the given sequence.

Output

Print an integer — the count of pairs.
Your answer maybe too large, you'd better use `long long` dat type to carry it.

Examples

standard input	standard output
2 3 1 2	1
6 1 5 1 2 3 4 1	2

Note

In example 1, there are only 2 numbers meet $1 \oplus 2 = 3$.
In example 2, the 2 pairs are $(i, j) = (1, 5), (3, 4)$.
By the way, you can use the operator `^` in C/C++:

```c
// main.c
#include <stdio.h>
int main() {
    printf("%d\n", 9 ^ 13); // 4;

    long long ans = 1000000000 ^ 9999999999LL;
    printf("ans = %lld\n", ans);

    return 0;
}
```