Problem D
Integer Division
Time limit: 2 seconds

In `C++` division with positive integers always rounds down. Because of this, sometimes when two integers are divided by the same divisor they become equal even though they were originally not equal. For example in `C++`, `5/4` and `7/4` are both equal to `1`, but $5 \neq 7$.

Given a list of nonnegative integers and a divisor, how many pairs of distinct entries in the list are there that give the same result when both are divided by the divisor in `C++`?

Input

The first line of input contains two integers n ($1 \leq n \leq 200\,000$), the number of elements in the list, and d ($1 \leq d \leq 10^9$), the divisor.

The second line of input contains n integers $a_1, \ldots, a_n$ ($0 \leq a_i \leq 10^9$), where a_i is the i^{th} element of the list.

Output

Display a single integer indicating the number of distinct pairs of indices (i, j) with $1 \leq i < j \leq n$ such that $a_i/d = a_j/d$ when using integer division in `C++`. Note that the numbers in the list are not necessarily distinct (i.e. it is possible that $a_i = a_j$ for some indices $i \neq j$).

Sample Input 1	Sample Output 1
5 4 4 5 6 7 8	6

Sample Input 2	Sample Output 2
5 1 4 5 6 7 8	0

Sample Input 3	Sample Output 3
6 1 1 2 1 2 1 2	6

This page is intentionally left blank.