Problem B

Energy optimization

Time limit: 1 second
Memory limit: 512 megabytes

In a research project at the OLP Technology Center, Luna is assigned the task of optimizing energy usage in a system consisting of n electronic devices. Each device has a specific initial energy level, represented by a non-negative integer. Let a_i denote the initial energy level of the i-th device. These devices are quite special — when adjusting energy levels, one device can be used to update another, and the resulting values depend on bitwise operations. Specifically, Luna may perform the following operations any number of times:

- Choose two distinct indices i and j.
- Calculate $x = (a_i \text{ AND } a_j)$; $y = (a_i \text{ OR } a_j)$.
- Update the energy level of device i: $a_i \leftarrow x$.
- Update the energy level of device j: $a_j \leftarrow y$.

After performing any sequence of these operations, Luna must select k devices and compute the sum of the squares of their energy levels. His goal is to maximize this sum.

Input

- The first line contains two integers n and k ($1 \le k \le n \le 10^5$), the number of devices and the number of devices to be selected.
- The second line contains n non-negative integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 2^{30}$), representing the initial energy levels of the devices.

Output

Print a single integer, the maximum possible sum of squares of the chosen devices' energy levels after performing the operations. Since the result can be very large, output it modulo $10^9 + 7$.

Sample Input	Sample Output
3 2 3 8 2	125
3 3 1 3 5	51

Explanation

In Example 1, we can apply the operation between 3 and 8, obtaining 11 and 0. Note that applying operations involving 0 and 2 still gives 0 and 2. Hence, the final result is $11^2 + 2^2 = 125$, which is the maximum value we can get.

In Example 2, we apply the operation between 3 and 5, obtaining 7 and 1. Applying operations between 1 and 1 yields no change, so the total becomes $7^2 + 1^2 + 1^2 = 51$, which is the maximum value we can get.