



## Problem A. Color Numbers

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            **1 second**  
Memory limit:         **512 megabytes**

You are given an array of non-negative integers  $a_1, a_2, \dots, a_n$  and an integer  $k$ . Two indices  $i, j$  are called *inconsistent* if both of the two conditions hold:

1.  $a_i \text{ AND } a_j = a_i$ ,
2. Binary representation of  $(a_i \text{ XOR } a_j)$  have at least  $k$  bits set to 1.

Here AND stands for bitwise and operation, XOR stands for bitwise exclusive-OR operation.

A *consistent coloring* of  $a_1, \dots, a_n$  in  $m$  colors is an array of  $n$  integers  $c_1, \dots, c_n$  ( $1 \leq c_i \leq m$ ) such that there is no pair of inconsistent indices  $i, j$  with  $c_i = c_j$ .

Your task is to find the smallest possible number of colors in a consistent coloring of  $a_1, \dots, a_n$ .

### Input

In the first line you are given two integers  $n, k$  ( $1 \leq n, k \leq 5 \cdot 10^5$ ).

In the next line you are given  $n$  integers  $a_i$  ( $0 \leq a_i < 2^{22}$ ).

### Output

Print one integer — the smallest number of colors in a consistent coloring.

### Example

standard input	standard output
4 1 1 2 4 6	2

### Note

One possible consistent coloring in two colors is 1, 1, 1, 2. Since indices 2 and 4 are inconsistent, one color is not enough.