

C. Memory for Arrays

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

You get to work and turn on the computer. You start coding and give little thought to the RAM role in the whole process. In this problem your task is to solve one of the problems you encounter in your computer routine.

We'll consider the RAM as a sequence of cells that can contain data. Some cells already contain some data, some are empty. The empty cells form the so-called *memory clusters*. Thus, a memory cluster is a sequence of some consecutive empty memory cells.

You have exactly n memory clusters, the i -th cluster consists of a_i cells. You need to find memory for m arrays in your program. The j -th array takes 2^{b_j} consecutive memory cells. There possibly isn't enough memory for all m arrays, so your task is to determine what maximum number of arrays can be located in the available memory clusters. Of course, the arrays cannot be divided between the memory clusters. Also, no cell can belong to two arrays.

Input

The first line of the input contains two integers n and m ($1 \leq n, m \leq 10^6$). The next line contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$). The next line contains m integers b_1, b_2, \dots, b_m ($1 \leq 2^{b_i} \leq 10^9$).

Output

Print a single integer — the answer to the problem.

Examples

input
5 3 8 4 3 2 2 3 2 2
output
2

input
10 6 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0
output
6

Note

In the first example you are given memory clusters with sizes 8, 4, 3, 2, 2 and arrays with sizes 8, 4, 4. There are few ways to obtain an answer equals 2: you can locate array with size 8 to the cluster with size 8, and one of the arrays with size 4 to the cluster with size 4. Another way is to locate two arrays with size 4 to the one cluster with size 8.

In the second example you are given 10 memory clusters with size 1 and 6 arrays with size 1. You can choose any 6 clusters and locate all given arrays to them.