

## CoPA Questions

For problems 1-3, you should solve them using **CoPA**  
and document the **entire** process (questions and answers) in a .md file.

---

### 1. Question1

One day, Cola developed a problem-solving plan. On the  $k$ -th day, he must complete  $k$  problems; otherwise, he will become lazy.

Cola has now found a question bank, which contains  $n$  sets of problem sets, each with a certain number of problems. However, he is very picky—he will only use each problem set once, and each day he can only use the problems from one problem set. After that, the problem set will be discarded. For each problem set, he does not need to complete all the problems in it.

So the question is, how many days can Cola keep solving problems before he becomes lazy?

Input format:

1. An integer  $n$ , representing how many sets of problem sets there are.
2.  $n$  integers  $a_1, a_2, \dots, a_n$ , which represent how many problems are in each problem set.

Output format:

A single integer representing the answer.

### 2. Question2

Given an integer sequence  $a_i$  of length  $n$ , and another integer sequence  $b_i$  of length  $n$ .

You can perform some modifications, where each time you can increase one  $a_i$  by 1, at a cost of  $b_i$ . You need to ensure that all  $a_i$  are not equal while minimizing the total cost.

You can perform an unlimited number of the following operations before making modifications: swap  $b_i$  and  $b_j$  for any  $1 \leq i, j \leq n$ .

Determine the minimum cost.

Input format:

1. An integer  $n$ .
2.  $n$  integers, where the  $i$ -th number represents  $a_i$ .
3.  $n$  integers, where the  $i$ -th number represents  $b_i$ .

Output format:

A single integer representing the value of the answer.

### 3. Question3

Kosuke is too lazy. He will not give you any legend, just the task:

Fibonacci numbers are defined as follows:

$$f(1) = f(2) = 1$$
$$f(n) = f(n-1) + f(n-2), n \geq 3$$

We denote  $G(n, k)$  as an index of the  $n$ -th Fibonacci number that is divisible by  $k$ . For given  $n$  and  $k$ , compute  $G(n, k)$ .

As this number can be too big, output it by modulo  $10^9 + 7$ .

For example:  $G(3, 2) = 9$  because the 3-rd Fibonacci number that is divisible by 2 is 34.

Input format:

Two integers  $n$  and  $k$  ( $1 \leq n \leq 10^{18}, 1 \leq k \leq 10^5$ ). It is guaranteed that the sum of  $k$  across all test cases does not exceed  $10^6$ .

Output format:

For each test case, output the only number: the value  $G(n, k)$  taken by modulo  $10^9 + 7$ .