Competitive Programming Contest: Binary Search

A. Worry about Marks

5 seconds, 256 megabytes

In a regular school, Yamcha is a student who's really worried about how he did on his final exam, which was prepared by the professional Dr. Tea. This wasn't just any test—it was a world-class final exam, and Yamcha was dying to know how he stacked up against the other N students who took it. He's curious about how many students scored lower than him and how many scored higher. Plus, he wants to know the best mark just below his and the lowest mark just above his.

But there's a twist. When Dr. Tea was reading out everyone's marks, he realized he had forgotten to grade Yamcha's test! As Dr. Tea scrambles to find and grade Yamcha's test, Yamcha starts wondering about all the different marks he might get. He comes up with Q marks he thinks are possible and wants to figure out where he would stand with each of those marks.

Yamcha needs your help to figure out the details for each mark. Can you help him understand his performance by finding out the number of students he did better than, the number of students he did poorer than, the highest mark among the students who scored less than him, and the lowest mark among those who scored more than him?

Input

The first line of input contains a single integer N ($1 \leq N \leq 10^6$) — the number of students taking the exam except Yamcha.

The second line of input contains N real numbers with 2 decimal places $m_1, m_2, \ldots, m_n$ ($0.00 \leq m_i \leq 100.00$), where m_i is the mark of the student i.

The third line of input contains a single integer Q ($1 \leq Q \leq 2 \cdot 10^5$) — the number of possible marks of Yamcha.

The fourth line of input contains Q real numbers with 2 decimal places $q_1, q_2, \ldots, q_n$ ($0.00 \leq q_i \leq 100.00$), where q_i is the ith possible mark of Yamcha.

Output

For each possible mark, output four numbers in one line, separated by a space.

The first two numbers are integers that represent the number of students scored lower than Yamcha and the number of students scored higher than Yamcha.

The last two numbers are real numbers with 2 decimal places, which represent the highest mark among the students who scored less than Yamcha and the lowest mark among those who scored more than Yamcha. Whenever it is impossible to find any of these numbers, replace that number with a hyphen '-'.

Scoring

Group	Points	Constraints
1	4	$N \leq 1000$
2	6	No additional constraints

input

```
3
1.00 3.00 4.00
3
100.00 1.50 3.00
```

output

```
3 0 4.00 -
1 2 1.00 3.00
1 1 1.00 4.00
```

If Yamcha gets 100.00 marks,

- All the 3 other students scored less than him.
- None of the students scored higher than him.
- The highest mark that is less than his score is 4.00.
- None of the marks is higher than Yamcha's mark.

B. Flappy Bird Pro Max

1 second, 256 megabytes

Jack stumbled upon a super cool game called Flappy Bird Pro Max, which was a bit different from the original game everyone loved. In this version, the bird couldn't just fly up or down whenever Jack wanted because to get that ability, he had to pay RM 1000! Thinking that was way too much for just a game feature, Jack figured out a smart way to play without spending his money. He found out that if his bird accidentally hit a pipe, he could watch a short ad, and the bird would come back to life, skipping the pipe it hit. So, Jack's new mission was to finish the game by watching as few ads as possible.

The tricky game had a canvas height of h and n pairs of pipes, where Jack had to guide the bird through each pair to complete a level. Each pair i had one pipe coming down from the top and another going up from the bottom, with sizes of t_i and b_i, respectively. Jack realized that if he chose the best starting height for his bird, he could avoid hitting the pipes as much and wouldn't have to watch so many ads.

As Jack's best friend, he needs your help to figure out the **minimum** number of ads he needs to watch to finish a level. You also need to help him find out how many distinct starting heights will let him watch that number of ads.

Input

The first line of the input contains two integers h ($3 \leq h \leq 500002$) and n ($1 \leq n \leq 2 \cdot 10^5$) — the height of the game canvas and the number of pipe pairs in the level.

Each of the following n lines contains two integers t_i and b_i ($1 \leq t_i, b_i < h$) — the size of the pipe from the top and bottom for the pair i.

It is guaranteed that there is a gap between each pair of pipes ($t_i + b_i < h$).

Output

Output two integers separated by a single space on a single line. The first number is the minimum number of ads Jack needs to watch to complete the level, and the second is the number of starting heights on which that can be achieved.

Scoring

Group	Points	Constraints
1	6	$h \le 10000$
2	9	No additional constraints

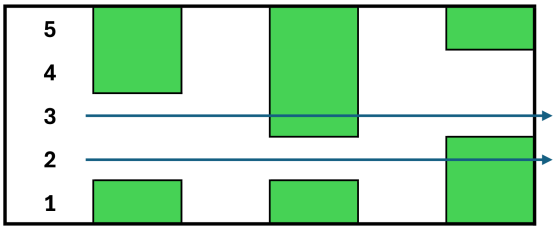
input

```
5 3
2 1
3 1
1 2
```

output

```
1 2
```

To pass the level, Jack can choose the starting height of 2 or 3 to watch the ad only once.



C. Come for the Class, Stay for the Demo

1 second, 256 megabytes

In Universiti Yalama, the coordinator of the Data Structure course, Dr. Alexander Great notices that the way they teach isn't the best. Right now, they have to split the lecturers and demonstrators into different rooms, which makes things unnecessarily complicated. Dr. Alexander had a big dream: he wanted to make a world-class big lecture hall where everyone could learn together. He believes in making learning fun by creating problems for students to solve, saying, "We need to make problems and solve problems together!"

In his big plan, there are N lecturers, each with their own laptop. When the class starts, each lecturer i needs P_i minutes to set up his laptop, after which the lecturer creates and explains his first problem. After that, he produces a new problem every Q_i minutes.

At the same time, there are M lab demonstrators acting as teaching assistants, who also have their own preparation time. Each demonstrator i requires R_i minutes to get ready and look over the problems, after which the demonstrator solves the first problem. Then, she solves another problem every S_i minutes.

All lecturers and demonstrators are very busy. As soon as all the problems are created, the lecturers leave, and then the demonstrators will enter the hall to start solving the problems. Once they are done, they too leave to rest and end the class immediately.

There is a student named Julius who only comes to the class to see his favorite demonstrator in action. He wants to ensure he enters the hall just when the demonstrators do. Even though Julius knows the total time allocated for the class, he doesn't know the number of problems the lecturers are planning to have.

Can you help him calculate the time when the demonstrators arrive?

Input

The first line of the input contains a single integer T ($1 \le T \le 10^9$) — the total time allocated for the class in minutes.

The second line of the input contains a single integer N ($1 \le N \le 100$) — the number of lecturers.

Each of the following N lines contains two integers P_i and Q_i ($1 \le P_i, Q_i \le 10^9$) — the number of minutes of the initial preparation and problem creation time of the lecturer i.

The next line of the input contains a single integer M ($1 \le M \le 100$) — the number of lab demonstrators.

Each of the following M lines contains two integers R_i and S_i ($1 \le R_i, S_i \le 10^9$) — the number of minutes of the initial preparation and problem-solving time of the demonstrator i.

Output

Output a single integer — the number of minutes between the start of the class and the arrival of the demonstrators.

Scoring

Group	Points	Constraints
1	16	$T \le 2 \cdot 10^5$
2	9	No additional constraints

input

```
12
1
3 1
1
5 1
```

output

```
5
```

input

```
20
2
3 2
1 3
3
3 1
4 1
5 1
```

output

```
13
```

In the first example, the lecturer plans to prepare three problems for the class.

- The lecturer prepares the first problem 3 minutes after the class is started.
- The lecturer prepares the second problem 4 minutes after the class is started.
- The lecturer prepares the third problem 5 minutes after the class is started.
- The lecturer leaves immediately, and the demonstrator enters the class.
- The demonstrator solves the first problem 10 minutes after the class is started.
- The demonstrator solves the second problem 11 minutes after the class is started.
- The demonstrator solves the third problem 12 minutes after the class is started.
- The class ends in 12 minutes.