

We are observing the movement of  $N$  trucks on the road. The road can be represented as a number line. The points on the line denoting cities are integers. The cities are denoted with the number of the corresponding point on the line.

All trucks are moving with **the same speed** and **no single truck is standing still at any given moment**. Each truck takes 1 minute to pass the distance between two adjacent cities.

You are given the **route** of each truck. All of the trucks start their route **at the same initial moment**.

The route is given as an array of  $k$  cities:  $A_1, A_2, \dots, A_k$ . The truck starts from city  $A_1$  and drives to city  $A_2$ , then turns and drives to city  $A_3$  and so on. Given the fact that the truck turns, it will hold:

$$A_1 < A_2 > A_3 < A_4 > \dots \text{ or } A_1 > A_2 < A_3 > A_4 < \dots$$

The time it takes for the truck to turn is negligible.

One possible route is 2, 5, 1, 7. The truck is at city number 2 initially, 3 minutes after departure it arrives to city number 5. It turns and continues towards the city number 1 in which it arrives 7 minutes after departure. It turns again and drives towards city number 7 in which it arrives at moment 13.

After the truck completes his route, aliens appear and take it away in their space rocket.

For some pairs of trucks, we want to know the number of times they encountered each other on the road. In other words, how many times they appeared to be on the same position (the position they encountered doesn't need to be an integer; i.e. they could've encountered at position 2.5).

Write a programme that will, for a given number of trucks  $N$  and their routes and for  $M$  pairs of trucks, determine the number of encounters for each pair.

**Please note:** each pair of trucks we want to know the number of encounters for, it will hold:

- they won't be at the same place in the moment when one of them (or both) are being taken away by aliens
- they won't be at the same place in the initial moment or in the moment when one of them (or both) are turning

The upper statement **won't hold** for all pairs of trucks, but **only the pairs** we want to know the number of encounters for.

### INPUT

The first line of input contains the integers  $N$  and  $M$  ( $1 \leq N \leq 10^5$ ,  $1 \leq M \leq 10^5$ ), the number of trucks and the number of pairs of trucks we want to know the number of encounters for.

The  $i^{\text{th}}$  of the following  $N$  lines contains the description of the route of the  $i^{\text{th}}$  truck.

The first integer in the line,  $K_i$  ( $2 \leq K_i \leq 3 \cdot 10^5$ ) represents the number of cities on the truck's route. After it  $K_i$  integers follow,  $A_j$  ( $1 \leq A_j \leq 10^9$ ), the ordinal numbers of the cities on the truck's route given in the order which the truck visits them.

The sum of routes of all the trucks won't exceed  $3 \cdot 10^5$ .

Each of the following  $M$  lines contains two integers  $(a_i, b_i)$ , the ordinal numbers of the trucks we want to know the number of encounters for.

### OUTPUT

Output  $M$  lines. The  $i^{\text{th}}$  line must contain the number of encounters of the  $i^{\text{th}}$  pair of trucks from the input.

### SCORING

In the test cases worth 50% of total points, it will hold  $N \leq 10^2$ ,  $K_i \leq 10^3$ ,  $M \leq 10^3$ .

**SAMPLE TESTS**

<p><b>input</b></p> <p>3 3 3 1 3 1 2 2 1 3 3 1 3 1 2 2 3 3 1</p> <p><b>output</b></p> <p>1 0 2</p>	<p><b>input</b></p> <p>2 1 4 1 6 3 6 7 3 4 2 6 5 6 1 1 2</p> <p><b>output</b></p> <p>3</p>	<p><b>input</b></p> <p>3 4 3 1 4 2 4 3 4 2 4 3 4 1 3 1 2 2 3 3 1 1 3</p> <p><b>output</b></p> <p>2 1 2 2</p>
--	--	--