

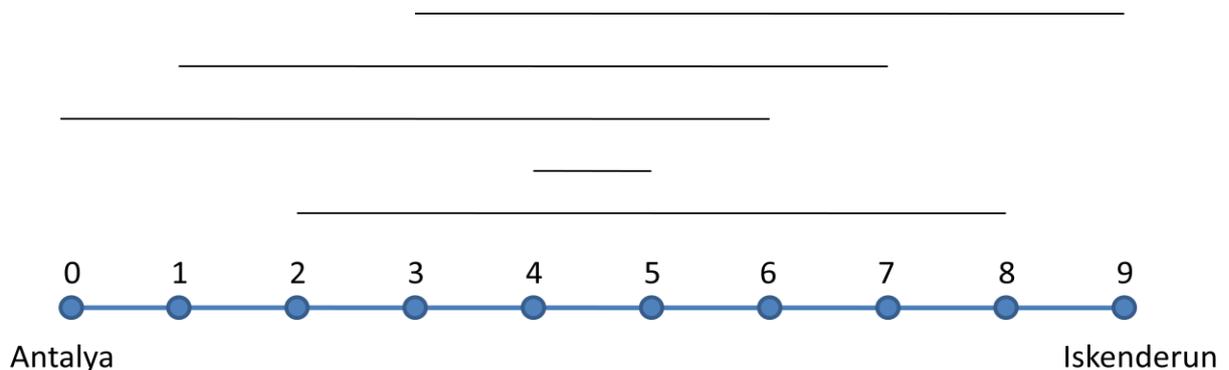
# Mediterranean

Coastal towns in the Mediterranean coast of Turkey are regularly aligned on a straight-line highway, with west-end at Antalya and east-end at Iskenderun. The distance between neighboring towns is 1 km. Antalya is located at kilometer 0 and all other towns are identified with their integer distance to Antalya.

There are  $N$  eastbound passengers (numbered  $1, 2, \dots, N$ ) with known source and destination town pairs. Let's use  $b_i$  and  $e_i$  to respectively denote the source and destination towns for passenger  $i$ . An eastbound bus with departure town  $d$  and arrival town  $a$  can carry passenger  $i$  if and only if  $d \leq b_i < e_i \leq a$ .

Given source and destination town pairs for  $N$  passengers and  $M$  eastbound buses, your task is to find how many passengers each of the buses can carry. To make the task interactive, a *shift* value is introduced so that the result of the current query is also dependent on the previous query (check *input format* section).

## Example



The figure plots the source and destination towns for  $N=5$  passengers, where  $\langle \text{source}, \text{dest} \rangle$  town pairs are:  $\langle 2, 8 \rangle$ ,  $\langle 4, 5 \rangle$ ,  $\langle 0, 6 \rangle$ ,  $\langle 1, 7 \rangle$ ,  $\langle 3, 9 \rangle$ . An eastbound bus with departure town 3 and arrival town 7 can carry only one passenger, while an eastbound bus with departure town 0 and arrival town 6 can carry two passengers.

## Task

Please write a program that finds how many passengers can be carried by each of **M** eastbound buses (queries). Your program should read input from standard input and should write output to the standard output. The input/output format is explained below with a sample.

input file (Standard Input)	output file (Standard Output)
5	1
2 8	2
4 5	4
0 6	1
1 7	1
3 9	
5	
3 7	
-1 5	
-1 7	
-2 1	
3 7	

*Input format:* The first line has a single integer **N**. The next **N** lines contain the itinerary of **N** passengers in the form of <source, dest> town pairs. Next line contains a single integer **M**, the number of queries. Each of the next **M** lines contains two integers *d* and *a* for the next query. The query asks how many passengers a bus with departure and arrival town pairs <*d+shift*, *a+shift*> can carry. The integers on the same lines are space-separated.

*shift:* *shift* is zero at the beginning. At the end of each query *shift* is equal to the result of the current query. In the example, for instance, *shift*=1 after the first query, *shift*=2 after the second query, and *shift*=4 after the third query.

*Output format:* **M** lines each with a single integer. Each integer is the result of the respective query.

## Subtasks

### Subtask 1 (9 points)

$$1 \leq \mathbf{N}, \mathbf{M} \leq 5\,000$$

$$0 \leq b_i < e_i \leq 400\,000$$

$$0 \leq d+shift < a+shift \leq 400\,000$$

**Subtask 2 (23 points)**

$$1 \leq \mathbf{N}, \mathbf{M} \leq 50\,000$$

$$0 \leq b_i < e_i \leq 10^9$$

$$0 \leq d + \text{shift} < a + \text{shift} \leq 10^9$$

**Subtask 3 (32 points)**

$$1 \leq \mathbf{N}, \mathbf{M} \leq 200\,000$$

$$0 \leq b_i < e_i \leq 10^9$$

$$0 \leq d + \text{shift} < a + \text{shift} \leq 10^9$$

**Subtask 4 (36 points)**

$$1 \leq \mathbf{N}, \mathbf{M} \leq 500\,000$$

$$0 \leq b_i < e_i \leq 10^9$$

$$0 \leq d + \text{shift} < a + \text{shift} \leq 10^9$$

**Implementation details**

You have to submit only one file, called `mediterranean.c`, `mediterranean.cpp` or `mediterranean.pas`. The file implements your full program.