

Problem A. Strange Device

Time limit: 4 seconds
Memory limit: 512 megabytes

Archaeologists have found a strange device that was probably created by some ancient civilization. The device has a screen that displays two integers: x and y .

After exploring the device the scientists have made a conclusion that the device is kind of a clock. It measures time t passed from some moment in the past, but shows it in some weird way, probably used by the creators of the device. If the time passed is an integer t , the two integers displayed are: $x = ((t + \lfloor \frac{t}{B} \rfloor) \bmod A)$, and $y = (t \bmod B)$. Here $\lfloor x \rfloor$ is the *floor function* — the greatest integer less or equal to x .

The archaeologists have studied the device and found out that its screen wasn't turned on all the time. Actually it was only working during n continuous periods of time, the i -th of them was from the moment l_i to the moment r_i , inclusive. Now the scientists would like to calculate how many distinct pairs (x, y) were shown by the device when its screen was on.

Two pairs (x_1, y_1) and (x_2, y_2) are distinct if $x_1 \neq x_2$ or $y_1 \neq y_2$.

Input

The first line contains three integers n , A , and B ($1 \leq n \leq 10^6$; $1 \leq A, B \leq 10^{18}$).

Each of the following n lines contains two integers l_i and r_i , the beginning and the end of the i -th segment $[l_i, r_i]$ when the device screen was turned on ($0 \leq l_i \leq r_i \leq 10^{18}$; $r_i < l_{i+1}$).

Output

Output the number of distinct pairs (x, y) that were shown on the device screen when it was turned on.

Scoring

Let $S = \sum_{i=1}^n (r_i - l_i + 1)$ and $L = \max_{i=1}^n (r_i - l_i + 1)$.

Subtask 1 (points: 10)

$S \leq 10^6$.

Subtask 2 (points: 5)

$n = 1$.

Subtask 3 (points: 5)

$A \cdot B \leq 10^6$.

Subtask 4 (points: 5)

$B = 1$.

Subtask 5 (points: 5)

$B \leq 3$.

Subtask 6 (points: 20)

$B \leq 10^6$.

Subtask 7 (points: 20)

$L \leq B$.

Subtask 8 (points: 30)

No additional constraint.

Examples

input	output
3 3 3 4 4 7 9 17 18	4
3 5 10 1 20 50 68 89 98	31
2 16 13 2 5 18 18	5

Note

In the first test, the device screen shows the following integers.

t	(x, y)
4	(2, 1)
7	(0, 1)
8	(1, 2)
9	(0, 0)
17	(1, 2)
18	(0, 0)

So there are four distinct pairs $(0, 0)$, $(0, 1)$, $(1, 2)$, $(2, 1)$.

Problem B. Bridges

Time limit: 2 seconds
Memory limit: 512 megabytes

The length of all waterways in St. Petersburg is about 282 km, and the water surface is about 7 % of city area.

Wikipedia

St. Petersburg is located on n islands that are connected by m bridges. Islands are labeled by integers from 1 to n , and bridges — from 1 to m . Every bridge connects two distinct islands. Some bridges were built in Peter the Great era, and some of them were built recently. That's why different bridges have different weight limits. Namely, only the cars with weight not exceeding d_i can drive through the i -th bridge. Sometimes some of the bridges in St. Petersburg are being renovated, but this doesn't necessarily make the bridge stronger, so, some of d_i may either increase or decrease. You develop a product that will help citizens and city guests. Currently, you develop a module that has to answer two types of queries:

1. The weight limit of bridge b_j became equal to r_j .
2. Count the number of islands, that can be reached by the car of weight equal to w_j from island s_j .

Answer all queries of the second type.

Input

The first line contains two integers n and m — the number of islands and bridges in St. Petersburg ($1 \leq n \leq 50\,000$, $0 \leq m \leq 100\,000$).

The i -th of the following m lines contains three integers u_i , v_i , and d_i that describe the bridge connecting islands u_i and v_i , initially its weight limit equals to d_i ($1 \leq u_i, v_i \leq n$; $u_i \neq v_i$; $1 \leq d_i \leq 10^9$).

The following line contains single integer q — the number of queries ($1 \leq q \leq 100\,000$). The following q lines contain queries.

The description of each query starts with an integer t_j ($t_j \in \{1, 2\}$).

If $t_j = 1$, the query is of the first type, then two integers b_j and r_j follow, they mean that the weight limit of the bridge b_j becomes equal to r_j ($1 \leq b_j \leq m$, $1 \leq r_j \leq 10^9$).

If $t_j = 2$, the query is of the second type, then two integers s_j and w_j follow, that describe a car of weight w_j located on island s_j ($1 \leq s_j \leq n$, $1 \leq w_j \leq 10^9$).

Output

For each query of the second type print the answer on a separate line.

Scoring

Subtask 1 (points: 13)

$n \leq 1\,000$, $m \leq 1\,000$, $q \leq 10\,000$.

Subtask 2 (points: 16)

Islands and bridges form a chain, $m = n - 1$, $u_i = i$, $v_i = i + 1$ ($1 \leq i \leq m$).

Subtask 3 (points: 17)

Islands and bridges form a complete binary tree, $n = 2^k - 1$, $m = n - 1$, $u_i = \lfloor \frac{i+1}{2} \rfloor$, $v_i = i + 1$ ($1 \leq k \leq 15$, $1 \leq i \leq m$).

Subtask 4 (points: 14)

All t_j equal to 2.

Subtask 5 (points: 13)

Islands and bridges form a tree, $m = n - 1$.

Subtask 6 (points: 27)

No additional constraint.

Examples

input	output
3 4 1 2 5 2 3 2 3 1 4 2 3 8 5 2 1 5 1 4 1 2 2 5 1 1 1 2 3 2	3 2 3
7 8 1 2 5 1 6 5 2 3 5 2 7 5 3 4 5 4 5 5 5 6 5 6 7 5 12 2 1 6 1 1 1 2 1 2 1 2 3 2 2 2 1 5 2 1 3 1 2 2 4 2 4 2 1 8 1 2 1 1 2 1 3	1 7 7 5 7 7 4

Note

Green lines represent bridges that the car from the query can drive through. Green vertices represent islands that are reachable by this car. Arrow points to the island, where the car is located initially.

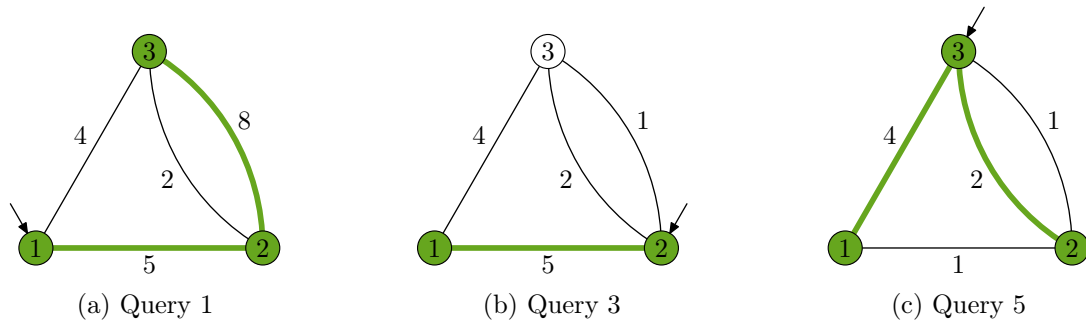


Рис. 1: Picture for the first test

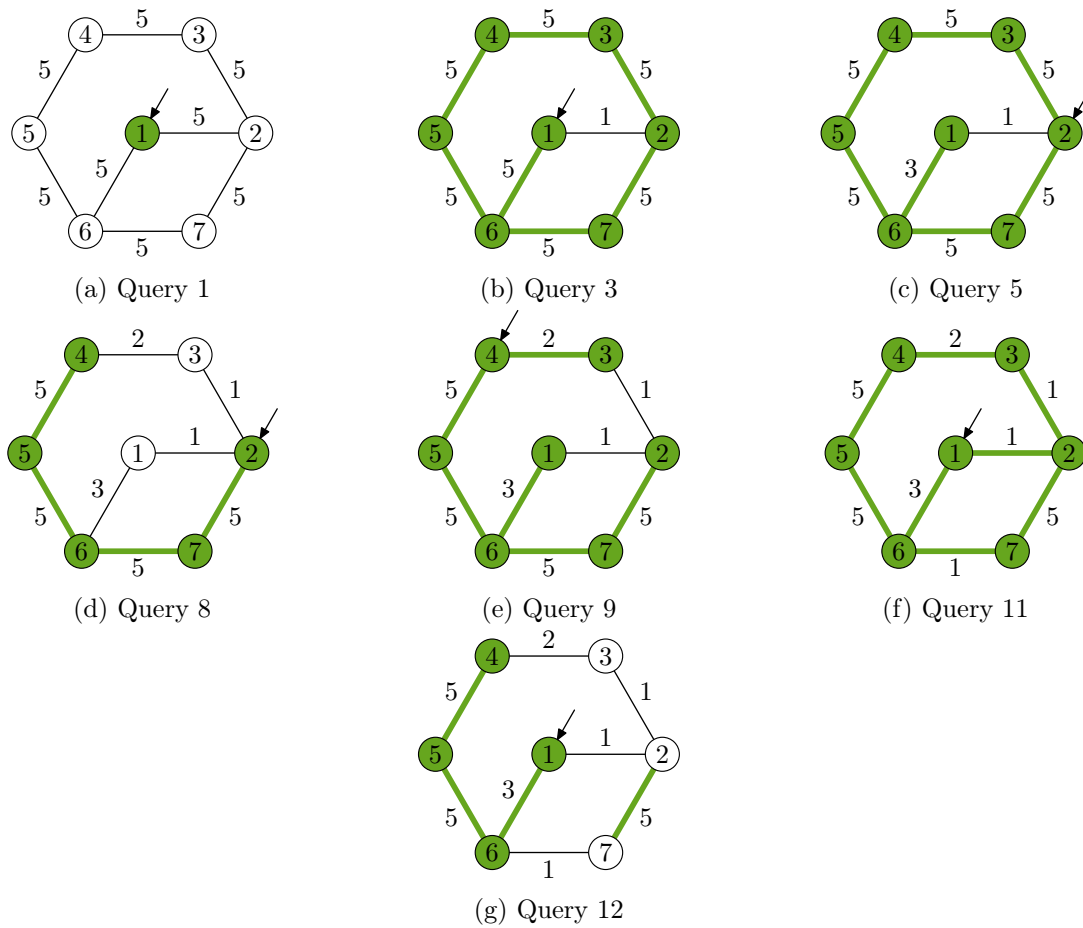


Рис. 2: Picture for the second test

Problem C. Street Lamps

Time limit: 5 seconds
Memory limit: 512 megabytes

There is a self-driving taxi in Innopolis that drives on a long street. The street consists of $n + 1$ taxi stops and n segments connecting adjacent stops. There is a street lamp on each segment. The i -th lamp illuminates the segment connecting stop i and $i + 1$, if the lamp is on. Otherwise, it's dark on the segment.

For the purpose of safety the self-driving taxi can only drive on the segments that are illuminated. In other words, the taxi can drive from the stop a to the stop b ($a < b$), if the segments between a and $a + 1$, $a + 1$ and $a + 2$, $\dots$, $b - 1$ and b are illuminated.

After breakdowns or repairs the street lamps can turn on or turn off. You are given the initial state of the lamps at the moment 0. After that in the end of hours $1, 2, \dots, q$ events take place. Exactly one event takes place in the end of each hour, there are two types of events:

- “toggle i ” — the i -th lamp changes its state: if the lamp was on, it turns off, if the lamp was off, it turns on.
- “query $a\ b$ ” — the head of the self-driving taxi department wonders, what is the total time in hours from 0 up to the current time when the taxi was able to drive from stop a to stop b .

Help the head of self-driving taxi department to answer the questions.

Input

The first line contains two integers n and q ($1 \leq n, q \leq 300\,000$) — the number of street lamps and number of events.

The second line contains a string s that describes the initial state of the lamps ($|s| = n$), s_i is ‘1’ if the i -th lamp is on, and s_i is ‘0’ if the i -th lamp is off.

Each of the following q lines describes events. The i -th of these lines describes an event that takes place in the end of the hour i .

- “toggle i ” ($1 \leq i \leq n$) — the i -th lamp changes its state.
- “query $a\ b$ ” ($1 \leq a < b \leq n + 1$) — calculate the number of hours until the current moment when the taxi was able to drive from stop a to stop b .

At least one of the events is **query**.

Output

For each **query** event print a single integer: the answer to the question.

Scoring

Subtask 1 (points: 20)

$n \leq 100$, $q \leq 100$.

Subtask 2 (points: 20)

For all “query $a\ b$ ” events $b - a = 1$.

Subtask 3 (points: 20)

For all “toggle i ” events the i -th lamp is turning on.

Subtask 4 (points: 20)

All `toggle` events happen before all `query` events.

Subtask 5 (points: 20)

No additional constraint.

Example

input	output
5 7	1
11011	2
query 1 2	0
query 1 2	0
query 1 6	1
query 3 4	2
toggle 3	
query 3 4	
query 1 6	

Note

In the sample test:

Hour	Lamp states	Query	Answer
1	11011		
		query 1 2	1
2	11011		
		query 1 2	1 and 2
3	11011		
		query 1 6	None
4	11011		
		query 3 4	None
5	11011		
		toggle 3	
6	11111		
		query 3 4	6
7	11111		
		query 1 6	6 and 7