

Problem A. 奇異裝置

Time limit: 4 seconds
Memory limit: 512 megabytes

考古學家們發現了一個可能是由遠古文明所創造的奇異裝置。這個裝置有一個顯示器，可顯示兩個整數： x 與 y 。

考古學家們在研究了這個裝置後，發現它其實是一種特殊的計時器。它持續累計著由過去的某時刻起經過的時刻個數（以一整數 t 表示）。然而此裝置用了一種相當奇特的方式來做顯示。若目前所累計的時刻數量為 t ，裝置的顯示器將顯示兩個整數： $x = ((t + \lfloor \frac{t}{B} \rfloor) \bmod A)$ 以及 $y = (t \bmod B)$ 。此處的 $\lfloor x \rfloor$ 表示小於等於 x 的最大整數。

這群考古學家還發現，這個奇異裝置的顯示器並非隨時都在運作；實際上，僅有某 n 個時段顯示器會運作。第 i 個時段為由時刻 l_i 至時刻 r_i 的中連續的時刻構成的區段（包含 l_i 及 r_i ）。考古學家們希望得知在此裝置的顯示器運作時，共有多少個不同的數對 (x, y) 會出現在顯示器上。

兩個數對 (x_1, y_1) 和 (x_2, y_2) 不同，定義為 $x_1 \neq x_2$ 或 $y_1 \neq y_2$ 。

Input

第一行包含三個整數 n 、 A 以及 B ($1 \leq n \leq 10^6$; $1 \leq A, B \leq 10^{18}$)。

接下來 n 行的每一行包含二整數 l_i 及 r_i ，表示顯示器有在運作的時間區段 $[l_i, r_i]$ ($0 \leq l_i \leq r_i \leq 10^{18}$; $r_i < l_{i+1}$)。

Output

輸出為一數字，表示顯示器運作期間，出現在顯示器上的不同的數對 (x, y) 總數。

Scoring

令 $S = \sum_{i=1}^n (r_i - l_i + 1)$ 且 $L = \max_{i=1}^n (r_i - l_i + 1)$ 。

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)

無特別限制。

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

在第一個範例中，顯示器於各時刻顯示的數對如下：

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

因此共有 4 個不同的數對 $(0, 0), (0, 1), (1, 2), (2, 1)$ 。

Problem B. 橋梁

Time limit: 2 seconds
Memory limit: 512 megabytes

聖彼得堡水路的總長度大約為 282 km，水面約佔城市總面積的 7 %。

Wikipedia

聖彼得堡位於由 m 座橋梁連接而成的 n 座島嶼上。島嶼用 1 到 n 的整數編號，橋梁用 1 到 m 的整數編號。每座橋連接兩個不同的島嶼。有些橋梁是在彼得大帝時代興建的，有些是近代才造的，也因此不同的橋梁其承重能力也不同。若一輛車要通過編號為 i 的橋梁，則車的重量不能超過 d_i 。有時聖彼得堡的橋梁會進行修繕工程，但這並不表示橋梁會因此提升其承重能力，換句話說，修繕後的橋梁，其承重能力 d_i 可能增加也可能減少。你要開發一個產品來幫助市民以及訪客，目前開發的模組中能進行兩類操作：

1. 將橋梁 b_j 的承重限制改為 r_j 。
2. 計算一輛重量為 w_i 的車，從島嶼 s_j 出發能到達的島嶼個數。

請你回答所有第二類操作的答案。

Input

第一行包含二整數 n 與 m — 分別表示聖彼得堡的島嶼數量及橋梁數量 ($1 \leq n \leq 50\,000$, $0 \leq m \leq 100\,000$)。

接下來 m 行中，第 i 行包含三個整數 u_i 、 v_i 及 d_i ，表示該橋梁所連接的兩座島嶼為 u_i 及 v_i ，且其初始的承重限制為 d_i ($1 \leq u_i, v_i \leq n$; $u_i \neq v_i$; $1 \leq d_i \leq 10^9$)。

接下來這行包含一整數 q — 表示操作的個數 ($1 \leq q \leq 100\,000$)。

接下來的 q 行每行為一個操作。

每個操作的描述由三個整數構成，開頭的整數為 t_j ($t_j \in \{1, 2\}$)。

若 $t_j = 1$ ，此操作即為第一類操作，接下來的兩個整數 b_j 及 r_j 表示橋梁 b_j 的承重限制將調整為 r_j ($1 \leq b_j \leq m$, $1 \leq r_j \leq 10^9$)；

若 $t_j = 2$ ，此操作即為第二類操作，接下來的兩個整數 s_j 及 w_j 表示有一重量為 w_j 的車將由島嶼 s_j 出發 ($1 \leq s_j \leq n$, $1 \leq w_j \leq 10^9$)。

Output

對每一個第二類操作，於個別的一行輸出其答案。

Scoring

Subtask 1 (points: 13)

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

Subtask 2 (points: 16)

島嶼和橋梁形成一個鏈 (chain), $m = n - 1$, $u_i = i$, $v_i = i + 1$ ($1 \leq i \leq m$)。

Subtask 3 (points: 17)

島嶼和橋梁形成一個完全二元樹 (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)

所有的 t_j 皆為 2.

Subtask 5 (points: 13)

島嶼和橋梁形成一個樹 (tree), $m = n - 1$.

Subtask 6 (points: 27)

無特別限制。

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

綠線表示當前操作中的汽車可以通過的橋梁。綠色頂點代表這輛車可以到達的島嶼。箭頭指向汽車最初所在的島嶼。

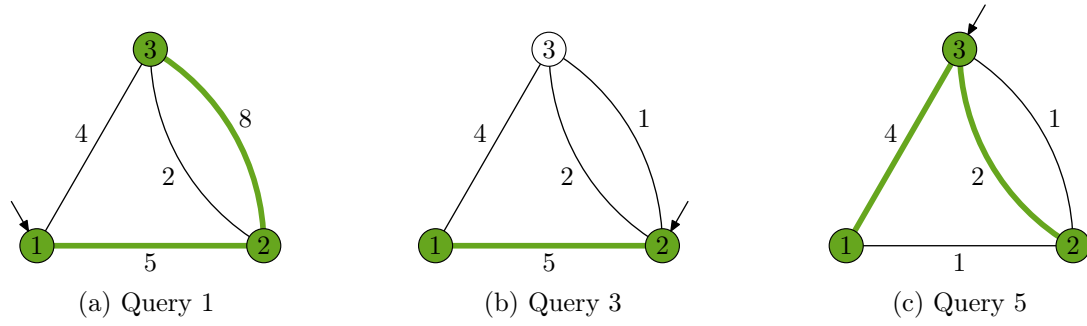


Рис. 1: Picture for the first test

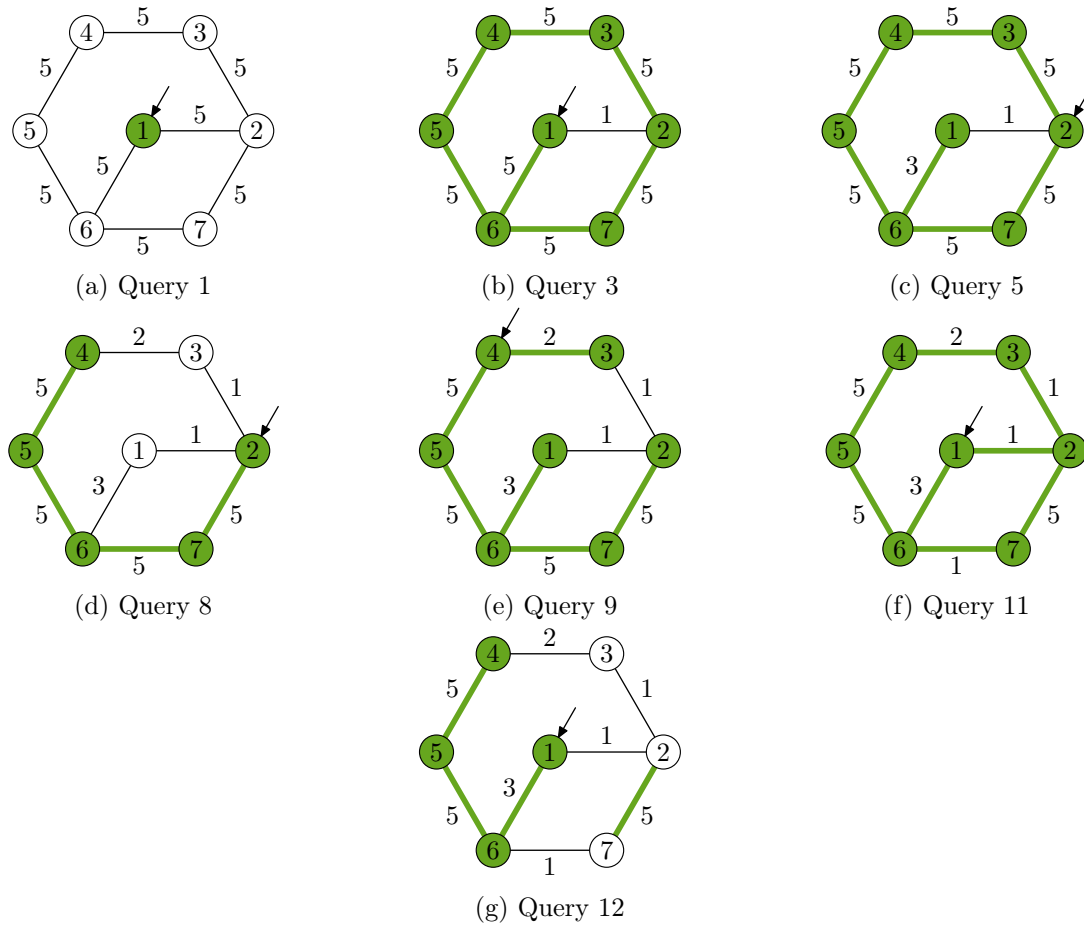


Рис. 2: Picture for the second test

Problem C. 路燈

Time limit: 5 seconds
Memory limit: 512 megabytes

一輛自動駕駛計程車正在 Innopolis 鎮的街道上行駛。這條街道上有 $n + 1$ 個計程車停靠站。這些停靠站將街道分隔為 n 個路段，每一路段都有一個路燈。當第 i 個路燈亮起，它會照亮第 i 到第 $i + 1$ 個停靠站之間的路段；否則，這個路段將是暗的。

考量到安全性，這輛自動駕駛計程車只能行駛在被照亮的路段。換言之，這輛計程車可以從停靠站 a 行駛至停靠站 b ($a < b$) 的話，表示停靠站 a 至 $a + 1$ 、 $a + 1$ 至 $a + 2$ 、 \dots 、 $b - 1$ 至 b 這些停靠站間的路段全都被照亮的。

在發生了一些意外的故障或修復後，街道上的路燈可能是亮起，也可能是熄滅的。

給定所有的路燈於時刻 0 的狀態，接下來的 q 個時刻 $(1, 2, \dots, q)$ 每個時刻會發生下列二事件其中之一：

- “toggle i ” — 切換第 i 個路燈的狀態。也就是說，若原來這盞路燈是亮起的，將變為熄滅；若原來是熄滅的，則變為亮起。
- “query $a\ b$ ” — 自駕計程車部門的負責人想知道，由時刻 0 起至當前時刻，共有多少小時 (連續兩個時刻間隔恰為一小時) 這輛計程車可以由停靠站 a 行駛至停靠站 b 。

請你協助自駕計程車部門的負責人，回答這些問題。

Input

第一行包含兩個整數 n 和 q ($1 \leq n, q \leq 300\,000$) — 表示路燈的數量與事件數。

第二行包含一個字串 s ，表示路燈的初始狀態 ($|s| = n$)， s_i 為 ‘1’ 表示第 i 個路燈是亮起的； s_i 為 ‘0’ 表示第 i 個路燈是熄滅的。

接下來 q 行每行描述一個時刻發生的事件。第 i 行描述時刻 i 所發生的事件。

- “toggle i ” ($1 \leq i \leq n$) — 切換第 i 個路燈的狀態。
- “query $a\ b$ ” ($1 \leq a < b \leq n + 1$) — 計算由時刻 0 起至當前時刻，共有幾個小時這輛計程車能從停靠站 a 行駛至停靠站 b (連續兩時刻的間隔恰為 1 小時)。

q 個事件中至少有一個是 query。

Output

對於每個 query 事件，輸出一個整數，表示該問題的答案。

Scoring

Subtask 1 (points: 20)

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

Subtask 2 (points: 20)

對所有的 “query $a\ b$ ” 事件， $b - a = 1$.

Subtask 3 (points: 20)

所有的 “toggle i ” 事件皆表示「亮起第 i 盞路燈」。換言之，“toggle i ” 事件發生後，第 i 盞路燈一定是亮的。

Subtask 4 (points: 20)

所有 `toggle` 事件發生於第一個 `query` 事件之前。

Subtask 5 (points: 20)

無特別限制。

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