

Problem A. Strange Device

Time limit: 4 seconds
Memory limit: 512 megabytes

Các nhà khảo cổ tìm thấy một thiết bị kỳ lạ có lẽ là của nền văn minh cổ đại tạo ra. Thiết bị có màn hình hiển thị hai số nguyên x và y .

Sau khi khám phá thiết bị này, các nhà khoa học đã đưa ra kết luận rằng thiết bị này là một loại đồng hồ. Nó xác định thời gian t so với một thời điểm nào đó trong quá khứ, nhưng hiển thị thời gian theo một cách kỳ lạ, có lẽ được sử dụng bởi những người tạo ra thiết bị này. Nếu thời gian đó là một số nguyên t , hai số nguyên được hiển thị là $x = ((t + \lfloor \frac{t}{B} \rfloor) \bmod A)$, và $y = (t \bmod B)$. Ở đây $\lfloor x \rfloor$ là hàm làm tròn xuống — là số nguyên lớn nhất nhỏ hơn hoặc bằng x .

Các nhà khảo cổ đã nghiên cứu thiết bị và phát hiện ra rằng màn hình của nó không được bật mọi lúc trong quá khứ. Thực chất, nó chỉ hoạt động trong n khoảng thời gian liên tục trong quá khứ, khoảng thứ i là từ thời điểm l_i đến thời điểm r_i , bao gồm cả hai đầu mút. Bây giờ các nhà khoa học muốn tính toán xem có bao nhiêu cặp phân biệt (x, y) được thiết bị hiển thị khi màn hình được bật.

Hai cặp (x_1, y_1) và (x_2, y_2) là phân biệt nếu $x_1 \neq x_2$ hoặc $y_1 \neq y_2$.

Input

Dòng đầu tiên chứa ba số nguyên n , A và B ($1 \leq n \leq 10^6$; $1 \leq A, B \leq 10^{18}$).

Mỗi dòng trong số n dòng tiếp theo chứa hai số nguyên l_i và r_i là thời điểm bắt đầu và kết thúc của đoạn $[l_i, r_i]$ khi thiết bị hoạt động trong quá khứ ($0 \leq l_i \leq r_i \leq 10^{18}$, $r_i < l_{i+1}$).

Output

In ra số lượng cặp phân biệt (x, y) được hiển thị trên thiết bị khi nó được bật trong quá khứ.

Scoring

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

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)

Không có ràng buộc gì thêm.

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

Trong test ví dụ đầu tiên, màn hình thiết bị hiển thị các số nguyên sau trong quá khứ.

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

Vì vậy có bốn cặp phân biệt $(0, 0)$, $(0, 1)$, $(1, 2)$, $(2, 1)$.

Problem B. Bridges

Time limit: 2 seconds
Memory limit: 512 megabytes

St. Petersburg nằm trên n hòn đảo được kết nối bởi m cây cầu. Các hòn đảo được gán nhãn bởi các số nguyên từ 1 đến n và các cây cầu – từ 1 đến m . Mỗi cây cầu nối hai hòn đảo khác nhau. Một số cây cầu được xây dựng trong thời đại Peter, và một số khác được xây dựng gần đây. Đó là lý do tại sao những cây cầu khác nhau có giới hạn trọng lượng khác nhau. Cụ thể, chỉ những chiếc xe có trọng lượng không vượt quá d_i mới có thể đi qua cây cầu i . Đôi khi, một số cây cầu ở St. Petersburg đang được cải tạo, nhưng điều này không nhất thiết sẽ làm cho cây cầu chắc chắn hơn, vì vậy, một số giá trị d_i có thể tăng hoặc giảm. Bạn phát triển một sản phẩm nhằm hỗ trợ cho cư dân và du khách của thành phố. Hiện tại, bạn phát triển một mô-đun phải thực hiện hai loại truy vấn:

1. Thay đổi giới hạn trọng lượng của cầu b_j thành r_j .
2. Đếm số lượng hòn đảo có thể đi tới được với chiếc xe có trọng lượng bằng w_j từ đảo s_j .

Hãy trả lời tất cả các truy vấn loại thứ hai.

Input

Dòng đầu tiên chứa hai số nguyên n và m — số lượng hòn đảo và số cây cầu ở St. Petersburg ($1 \leq n \leq 50\,000$, $0 \leq m \leq 100\,000$).

Dòng thứ i trong m dòng tiếp theo chứa ba số nguyên u_i , v_i , và d_i mô tả cây cầu nối hai hòn đảo u_i và v_i , có giới hạn trọng lượng ban đầu là d_i ($1 \leq u_i, v_i \leq n$; $u_i \neq v_i$; $1 \leq d_i \leq 10^9$).

Dòng tiếp theo chứa một số nguyên q — số lượng truy vấn ($1 \leq q \leq 100\,000$). Tiếp theo là q dòng chứa các truy vấn.

Mỗi truy vấn bắt đầu bằng một số nguyên t_j ($t_j \in \{1, 2\}$).

Nếu $t_j = 1$, truy vấn là loại một, tiếp theo là hai số nguyên b_j và r_j , nghĩa là giới hạn trọng lượng của cây cầu b_j sẽ thay đổi thành r_j ($1 \leq b_j \leq m$, $1 \leq r_j \leq 10^9$).

Nếu $t_j = 2$, truy vấn là loại hai, tiếp theo là hai số nguyên s_j và w_j mô tả chiếc xe có trọng lượng w_j ở hòn đảo s_j ($1 \leq s_j \leq n$, $1 \leq w_j \leq 10^9$).

Output

Đối với mỗi truy vấn loại thứ hai, in câu trả lời trên một dòng riêng biệt.

Scoring

Subtask 1 (points: 13)

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

Subtask 2 (points: 16)

Các hòn đảo và cây cầu tạo thành một chuỗi, $m = n - 1$, $u_i = i$, $v_i = i + 1$ ($1 \leq i \leq m$).

Subtask 3 (points: 17)

Các hòn đảo và cây cầu tạo thành một cây nhị phân hoàn chỉnh, $n = 2^k - 1$, $m = n - 1$, $u_i = \lfloor \frac{i+1}{2} \rfloor$, $v_i = i + 1$ ($1 \leq k \leq 15$, $1 \leq i \leq m$).

Subtask 4 (points: 14)

Tất cả t_j bằng 2.

Subtask 5 (points: 13)

Các hòn đảo và cây cầu tạo thành một cây, $m = n - 1$.

Subtask 6 (points: 27)

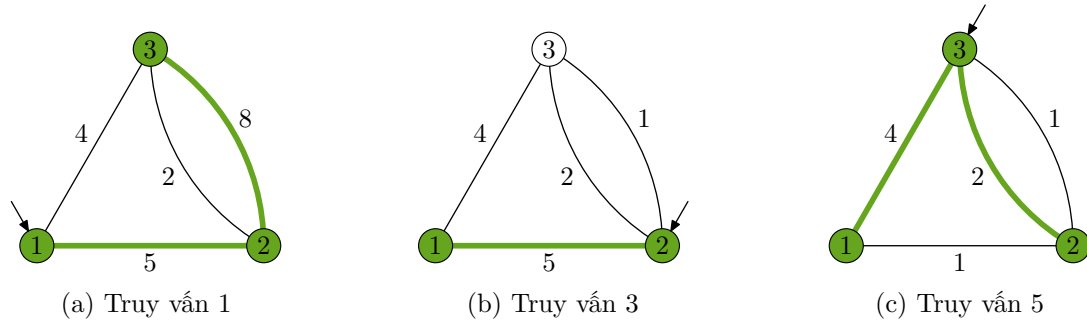
Không có thêm ràng buộc nào.

Examples

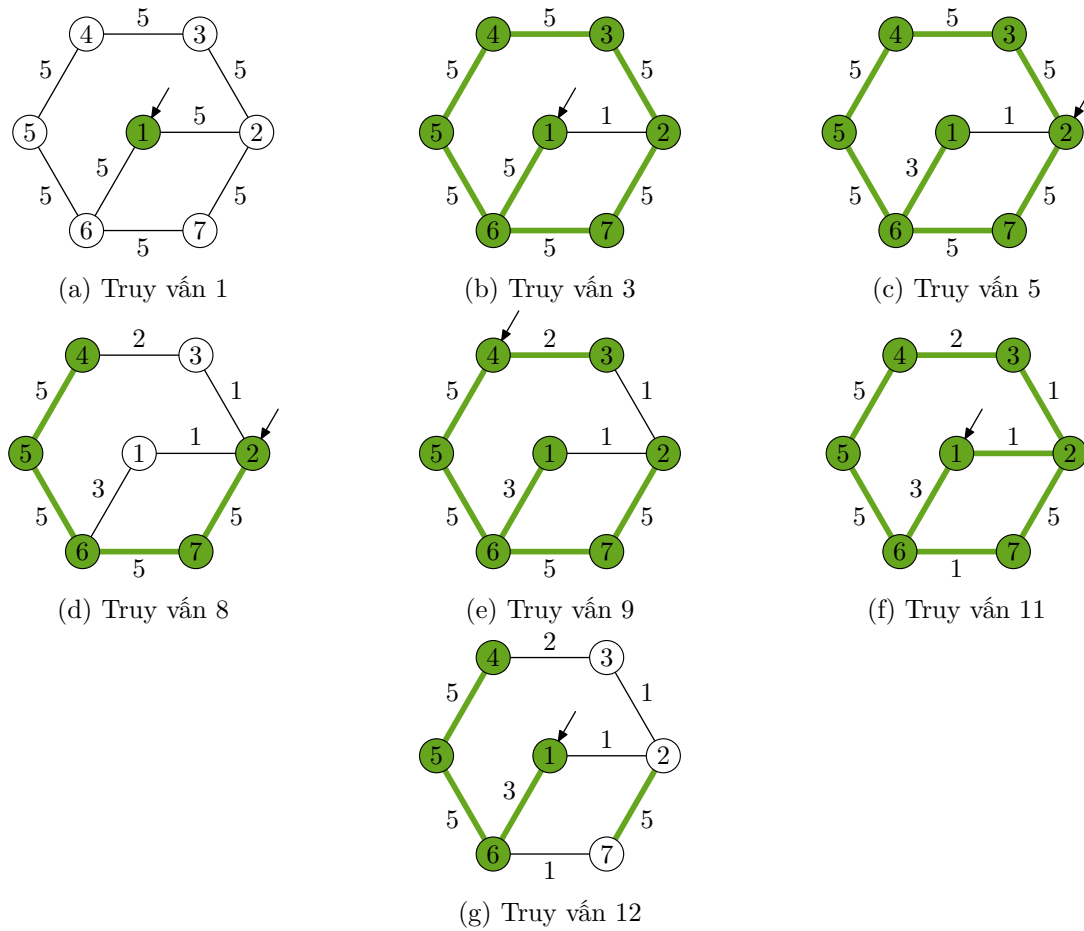
input	output
3 4 1 2 5 2 3 2 3 1 4 2 3 8 5 2 1 5 1 4 1 2 2 5 1 1 1 2 3 2	3 2 3
7 8 1 2 5 1 6 5 2 3 5 2 7 5 3 4 5 4 5 5 5 6 5 6 7 5 12 2 1 6 1 1 1 2 1 2 1 2 3 2 2 2 1 5 2 1 3 1 2 2 4 2 4 2 1 8 1 2 1 1 2 1 3	1 7 7 5 7 7 4

Note

Các đường màu xanh lá cây thể hiện cho những cây cầu mà chiếc xe ở truy vấn có thể đi qua. Các đỉnh màu xanh biểu diễn cho những hòn đảo có thể đi đến được bằng chiếc xe này. Mũi tên chỉ vào hòn đảo mô tả nơi chiếc xe được đặt ban đầu.



Hình 1: Hình ảnh cho ví dụ thứ nhất



Hình 2: Hình ảnh cho ví dụ thứ hai

Problem C. Street Lamps

Time limit: 5 seconds
Memory limit: 512 megabytes

Có một chiếc xe taxi tự lái trên một đường phố dài ở Innopolis. Đường phố này có $n + 1$ điểm dừng taxi và n đoạn đường nối các điểm dừng liền nhau. Trên mỗi đoạn đường có 1 cây đèn đường. Nếu cây đèn thứ i bật sáng, nó chiếu sáng cho đoạn đường nối điểm dừng thứ i và điểm dừng thứ $i + 1$. Ngược lại, đoạn đường này bị tối.

Để đảm bảo an toàn, xe taxi tự lái chỉ có thể chạy trên các đoạn đường được chiếu sáng. Hay nói cách khác, xe taxi có thể chạy từ điểm dừng a tới điểm dừng b ($a < b$), nếu như các đoạn đường nằm giữa a và $a + 1$, $a + 1$ và $a + 2$, \dots , $b - 1$ và b được chiếu sáng tại thời điểm đấy.

Sau khi bị hỏng hoặc sửa chữa, các cây đèn đường có thể bật hoặc tắt. Bạn được cho trạng thái ban đầu của các cây đèn đường tại thời điểm 0. Sau đó, có các sự kiện diễn ra vào thời điểm cuối của các giờ $1, 2, \dots, q$. Có chính xác một sự kiện diễn ra vào thời điểm cuối của mỗi giờ, có hai loại sự kiện:

- “toggle i ” — cây đèn thứ i chuyển trạng thái: nếu nó đang bật, thì nó sẽ được tắt; nếu nó đang tắt, thì nó sẽ được bật.
- “query $a b$ ” — trưởng phòng taxi tự lái muốn biết tổng số giờ từ thời điểm 0 tới thời điểm hiện tại mà chiếc xe taxi có khả năng lái từ điểm dừng a tới điểm dừng b .

Hãy giúp trưởng phòng taxi tự lái trả lời các câu hỏi này.

Input

Dòng đầu tiên chứa hai số nguyên n và q ($1 \leq n, q \leq 300\,000$) — số lượng cây đèn và số lượng sự kiện.

Dòng thứ hai chứa một xâu kí tự s mô tả các trạng thái ban đầu của các cây đèn ($|s| = n$), s_i là ‘1’ nếu như cây đèn thứ i đang bật, và s_i là ‘0’ nếu cây đèn thứ i đang tắt.

Mỗi dòng trong q dòng tiếp theo mô tả các sự kiện. Dòng thứ i mô tả một sự kiện sẽ diễn ra sau giờ i .

- “toggle i ” ($1 \leq i \leq n$) — cây đèn thứ i đổi trạng thái.
- “query $a b$ ” ($1 \leq a < b \leq n + 1$) — tính số lượng giờ cho đến thời điểm hiện tại mà taxi có thể lái từ điểm dừng a đến điểm dừng b .

Ít nhất một trong các sự kiện là query.

Output

Với mỗi sự kiện query, in ra một số nguyên là đáp án cho câu hỏi.

Scoring

Subtask 1 (points: 20)

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

Subtask 2 (points: 20)

Tất cả các sự kiện “query $a b$ ”, $b - a = 1$.

Subtask 3 (points: 20)

Tất cả các sự kiện “toggle i ”, cây đèn thứ i được bật. Nói cách khác, sau sự kiện “toggle i ”, cây đèn thứ i được bật.

Subtask 4 (points: 20)

Tất cả các sự kiện `toggle` diễn ra trước các sự kiện `query`.

Subtask 5 (points: 20)

Không có điều kiện ràng buộc gì thêm.

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

Trong ví dụ test này

Giờ	Trạng thái đèn	Câu hỏi	Câu trả lời
1	11011		
2	11011	query 1 2	1
3	11011	query 1 2	1 và 2
4	11011	query 1 6	Không có
5	11011	query 3 4	Không có
6	11111	toggle 3	
7	11111	query 3 4	6
		query 1 6	6 và 7