
가로등

Input file: **standard input**
Output file: **standard output**
Time limit: **5 seconds**
Memory limit: **512 megabytes**

이노폴리스에서는 자율 주행 택시가 운행하고 있다. 이 도시의 도로망은 $n + 1$ 개의 택시를 탈 수 있는 정류장과 이들을 잇는 n 개의 도로로 이루어져 있다. 도로 하나마다 가로등이 있다. i 번 가로등이 켜져 있다면, 정류장 i 와 정류장 $i + 1$ 를 잇는 도로를 밝힌다. 만약 꺼져 있다면, 이 도로는 어둡다.

보안 문제로, 자율 주행 택시는 가로등이 켜진 도로만 달릴 수 있다. 다른 말로 하면, 택시가 정류장 a 에서 정류장 b 로 가려면 ($a < b$), 정류장 a 과 정류장 $a + 1$ 를 잇는 도로, 정류장 $a + 1$ 과 정류장 $a + 2$ 를 잇는 도로, ..., 정류장 $b - 1$ 과 정류장 b 를 잇는 도로의 가로등이 켜져야 한다.

고장이 나거나, 수리를 하면 가로등이 켜지거나 꺼질 수 있다.

시점 0에서는 가로등들의 초기 상황이 주어진다. $1, 2, \dots, q$ 시간이 지나면 다음 두 상황 중 하나가 벌어질 수 있다.

- “toggle i ” – i 번 가로등이 반전된다. 즉, 이 가로등이 켜져 있었다면 꺼지고, 꺼져 있었다면 켜진다.
- “query $a b$ ” – 자율 주행 택시를 운영하는 곳의 대표가 다음을 궁금해한다. 시점 0에서 시작해서 현재에 이르기 까지 자율 주행 택시가 정류장 a 에서 정류장 b 까지 운행할 수 있었던 총 시간은 얼마일까?

자율 주행 택시를 운영하는 곳이 이 질의를 답하는 것을 도와주자.

Input

첫번째 줄에는 두 정수 n, q 가 주어진다. ($1 \leq n, q \leq 300\,000$) – 이는 가로등의 수와 이벤트의 개수이다.

두번째 줄에는 최초의 가로등 상태를 알려주는 문자열 s 가 주어진다. ($|s| = n$), 만약 가로등 i 가 켜져 있으면 s_i 는 ‘1’이고, 가로등 i 가 꺼져 있으면 s_i 는 ‘0’이다.

다음 q 줄 각각은 이벤트를 설명한다. 이 중 i 번째 줄은 i 시간이 지났을 때 이벤트를 설명한다.

- “toggle i ” ($1 \leq i \leq n$) – 가로등 i 이 반전된다.
- “query $a b$ ” ($1 \leq a < b \leq n + 1$) – 정류장 a 에서 정류장 b 까지 자율 주행 택시가 운행할 수 있었던 총 시간 수를 계산한다.

이벤트 중 최소 하나는 query이다.

Output

각각의 query 이벤트마다 하나의 정수를 출력하고, 이 수는 질의에 대한 답이다.

Scoring

Subtask 1 (points: 20)

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

Subtask 2 (points: 20)

모든 “query $a b$ ” 이벤트에서 $b - a = 1$ 이다.

Subtask 3 (points: 20)

모든 “toggle i ” 이벤트의 결과로 가로등 i 가 켜진다.

Subtask 4 (points: 20)

모든 toggle 이벤트는 모든 query 이벤트 전에 발생한다.

Subtask 5 (points: 20)

추가적인 제약조건이 없다.

Example

standard input	standard 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

예제 입력의 결과는 다음과 같다.

시간	가로등 상태	이벤트	응답
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