

Problem E. Infinite Parenthesis Sequence

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 256 megabytes

BaoBao has just found a sequence $A = a_0, a_1, \dots, a_{n-1}$ of length n in his left pocket. Each element a_i in this sequence is either a left parenthesis '(' or a right parenthesis ')'. As BaoBao dislikes short sequences, he decides to make the sequence infinitely long!

Let's denote b_i as the element in the i -th position of the infinite parenthesis sequence B . As B is an infinite sequence, i can be positive, zero, or even negative! To derive B from A , one can use the following equations:

$$\begin{cases} b_i = a_i & \text{if } 0 \leq i < n \\ b_i = b_{i-n} & \text{if } i \geq n \\ b_i = b_{i+n} & \text{if } i < 0 \end{cases}$$

As BaoBao is bored, he also crafts a generator to generate an infinite number of parenthesis sequences from sequence B ! Denote B^k ($k \geq 1$) as the k -th infinite sequence generated by the generator and b_i^k as the element in the i -th position of sequence B^k . For completeness, we define $B^0 = B$. One can derive B^k from B^{k-1} using the following equations:

$$\begin{cases} b_i^k = b_{i+1}^{k-1} & \text{if } b_i^{k-1} = '(' \\ b_i^k = b_{i-1}^{k-1} & \text{if } b_i^{k-1} = ')' \end{cases}$$

To obtain a deeper insight of the sequence, BaoBao would like to calculate the number of left parenthesis '(' in the continuous subsequence $b_l^k, b_{l+1}^k, b_{l+2}^k, \dots, b_{r-1}^k, b_r^k$ of B^k . Please write a program to help him calculate the answer.

Input

There are multiple test cases. The first line of the input contains an integer T , indicating the number of test cases. For each test case:

The first line contains a string s ($1 \leq |s| \leq 10^5$, $s_i \in \{ '(', ') ' \}$) indicating the sequence A . The i -th character s_i in s indicates the value of a_{i-1} .

The second line contains an integer q ($1 \leq q \leq 10^5$), indicating the number of queries.

For the following q lines, each line contains three integers k, l and r ($0 \leq k \leq 10^9$, $-10^9 \leq l \leq r \leq 10^9$), indicating a query.

It's guaranteed that neither the sum of $|s|$ nor the sum of q of all test cases will exceed 10^6 .

Output

For each query output one line containing one integer, indicating the number of left parenthesis '(' in the continuous subsequence $b_l^k, b_{l+1}^k, b_{l+2}^k, \dots, b_{r-1}^k, b_r^k$ of B^k .

Example

standard input	standard output
3	3
(())	3
3	0
0 -3 2	4
1 -2 3	1
2 0 0	1
))((7345
3	623
0 -3 4	45
2 1 3	3
3 -4 -1	
))((
4	
1234 -5678 9012	
123 -456 789	
12 -34 56	
1 -2 3	

Note

In the following explanation, the value of b_0^k is marked in ***bold and italics***.
For the first sample test case, we have $B^0 = \dots(())(())(())\dots$, $B^1 = \dots(())(())(())(())\dots$ and $B^2 = \dots(())(())(())(())(())\dots$, so the answer is 3, 3 and 0.
For the second sample test case, we have $B^0 = \dots))(())(())(())(())\dots$, $B^1 = \dots(())(())(())(())(())(())\dots$, $B^2 = \dots(())(())(())(())(())(())(())\dots$ and $B^3 = \dots(())(())(())(())(())(())(())(())\dots$, so the answer is 4, 1 and 1.