Popcount Words

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

The popcount word of interval $[l, r]$ is defined as

$$w(l, r) = s_l s_{l+1} \dots s_{r-1} s_r,$$

where $s_i = \text{popcount}(i) \bmod 2$. Here $\text{popcount}(i)$ means the number of ones in binary representation of integer i.

You will be given n intervals $[l_1, r_1], [l_2, r_2], \dots, [l_n, r_n]$. Let's build an extremely long string

$$S = w(l_1, r_1) + w(l_2, r_2) + \cdots + w(l_n, r_n),$$

here "`+`" denotes concatenation of strings.

You will also be given q queries. In the i-th query, you will be given a bit pattern p_i, your task is to report how often does p_i occur as a substring in S. Note that occurrences may overlap.

Input

The input contains only a single case.

The first line of the input contains two integers n and q $(1 \le n, q \le 100\,000)$, denoting the number of intervals and the number of queries.

In the next n lines, the i-th line $(1 \le i \le n)$ contains two integers l_i and r_i $(1 \le l_i \le r_i \le 10^9)$, describing the i-th interval.

In the next q lines, the i-th line $(1 \le i \le q)$ contains a non-empty string p_i consists of characters in {'`0`', '`1`'}, describing the pattern of the i-th query.

It is guaranteed that the total length of all patterns is at most $500\,000$.

Output

For each query, print a single line containing an integer, denoting the number of occurrences of the bit pattern in S.

Example

standard input	standard output
3 5	6
2 6	7
1 3	2
4 8	2
0	1
1	
11	
101	
0011010	

Note

- $w(l_1, r_1) = w(2, 6) =$ "`10100`", $w(l_2, r_2) = w(1, 3) =$ "`110`", $w(l_3, r_3) = w(4, 8) =$ "`10011`".
- $S = w(l_1, r_1) + w(l_2, r_2) + w(l_3, r_3) =$ "`1010011010011`".