DMOPC '20 Contest 5 P6 - Top Row

Time limit: 4.0s	**Memory limit:** 512M
Java: 8.0s	Java: 1G
PyPy 3: 8.0s	PyPy 3: 1G

After several weeks of training, you've finally managed to increase your typing speed to a staggering 40 WPM. Accordingly, you've also found another modified version of TypeRacer to intensify your training. In this game, you start off with a string S of lowercase characters. Every game is split into Q rounds. In the i^{th} round 2 integers l_i and r_i are given, and your task is to type out all of the distinct non-empty substrings of the substring of S from index l_i to r_i once.

To do this, you are provided with an empty screen at the start of the round. In one keystroke, you may add a lowercase character to the end of the current string on the screen, or you may press enter to submit the current string on the screen, simultaneously clearing the screen of all characters. The round is completed when you have submitted every distinct non-empty substring of the substring of S from index l_i to r_i once. Since you still strive to be as fast as possible, please compute the minimum number of keystrokes needed to complete each round.

Of course, the game wants you to complete the rounds in the order which they are given (supposedly there is extra merit in doing so), so you will not be given the parameters l_i and r_i directly. Instead, you will be given l_i' and r_i', and you may decrypt these values using the formula $l_i = l_i' \oplus \text{lastAns}$ and $r_i = r_i' \oplus \text{lastAns}$, where lastAns represents the answer to the previous round and $\oplus$ denotes the bitwise XOR operator. If there is no previous round, then $\text{lastAns} = 0$.

Constraints

$1 \le |S| \le 10^5$

$1 \le Q \le 5 \times 10^5$

$1 \le l_i \le r_i \le |S|$

S only contains lowercase characters.

Subtask 1 [5%]

$1 \le |S|, Q \le 400$

Subtask 2 [15%]

$\sum_{i=1}^{Q} (r_i - l_i) \le 10^6$

Subtask 3 [30%]

All characters of S are chosen independently and uniformly at random from the set of lowercase characters.

Subtask 4 [50%]

No additional constraints.

Input Specification

The first line contains the string S.

The second line contains the integer Q, the number of rounds.

The next Q lines each contain 2 integers l_i' and r_i', representing the encrypted l_i and r_i values.

Output Specification

Output Q lines, each containing the minimum number of keystrokes needed to complete the corresponding round.

Sample Input

```
araaraoneesan
5
4 6
6 5
29 26
61 50
31 19
```

Sample Input (Unencrypted)

```
araaraoneesan
5
4 6
8 11
1 6
6 9
1 13
```

Sample Output

```
14
28
59
30
522
```

Explanation

For your convenience, an unencrypted version of the sample input is provided above.

The first round is played on the substring of S from index 4 to 6, which is `ara`. One possible strategy is as follows:

1. Type `a` and press enter to submit in 2 keystrokes.
2. Type `ra` and press enter to submit in 3 keystrokes.
3. Type `ara` and press enter to submit in 4 keystrokes.
4. Type `r` and press enter to submit in 2 keystrokes.
5. Type `ar` and press enter to submit in 3 keystrokes.

Thus, this round was completed with $2 + 3 + 4 + 2 + 3 = 14$ keystrokes in total, and it can be shown that this is the minimum number of keystrokes possible. Note that you do not have to submit `a` twice, since every distinct substring only needs to be submitted once.