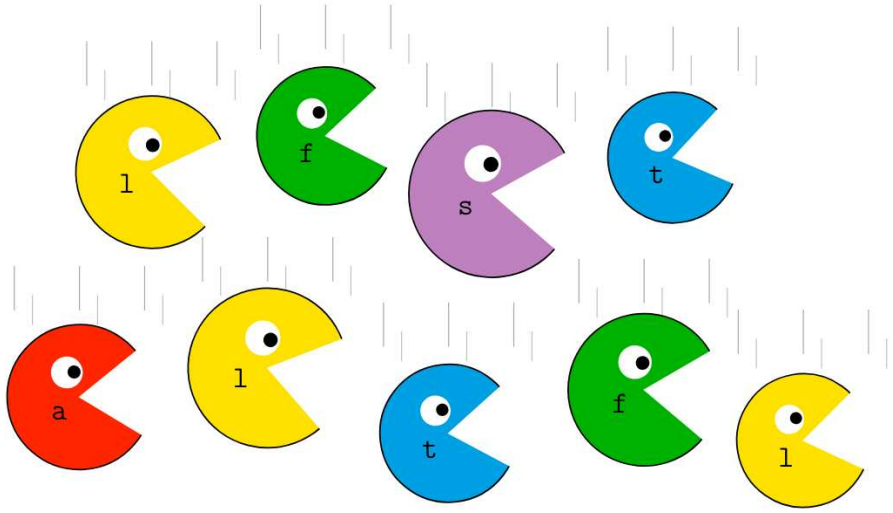
L LFS



You are designing the new videogame Live Fight Simulator. A level is defined by a string s of length n consisting of lowercase English letters, where each letter represents a type of enemy that appears in sequence.

To analyze the gameplay balance, you need to measure how repetitive different parts of the level are. You will consider q specific contiguous segments $s[l, r]$ of the level, with $1 \leq l \leq r \leq n$.

For each of these queries, you want to compute the length of the LFS (Longest Frequent Substring). Formally, for any string t:

- let $f(t)$ be the maximum frequency of any substring in t;
- let $|\text{LFS}(t)|$ be the maximum length of a substring of t that appears exactly $f(t)$ times.

For each query (l, r), you must compute $|\text{LFS}(s[l, r])|$, which represents the maximum length among the most repetitive patterns of enemy spawns in that part of the level.

A string x is a substring of a string y if x can be obtained from y by the deletion of several (possibly, zero or all) characters from the beginning and several (possibly, zero or all) characters from the end.

Input

The first line contains two integers n, q ($1 \leq n \leq 5 \cdot 10^5$, $1 \leq q \leq 5 \cdot 10^5$) — the length of the sequence and the number of level segments to analyze.

The second line contains a string s of length n consisting of lowercase English letters.

Each of the next q lines contains two integers l, r ($1 \leq l \leq r \leq n$), representing a query on the substring $s[l, r]$.

Output

Print q lines. The i-th line must contain a single integer: the value of $|\text{LFS}(s[l, r])|$ for the pair (l, r).

Samples

Sample input 1	Sample output 1
5 4 ababa 1 1 1 5 1 4 2 5	1 1 2 2

Explanation of sample 1.
In the first example:

- In the first query, the substring is $t = s[1, 1] =$ "a". The maximum frequency of a substring inside "a" is 1, reached by "a" itself, whose length is 1. Therefore, the answer is 1.
- In the second query, the substring is $t = s[1, 5] =$ "ababa". The maximum frequency of a substring inside "ababa" is 3, reached by "a", whose length is 1. Therefore, the answer is 1.
- In the third query, the substring is $t = s[1, 4] =$ "abab". The maximum frequency of a substring inside "abab" is 2, reached by "a", "b" and "ab". Among these strings, the one with maximum length is "ab", whose length is 2. Therefore, the answer is 2.
- In the fourth query, the substring is $t = s[2, 5] =$ "baba". The maximum frequency of a substring inside "baba" is 2, reached by "a", "b" and "ba". Among these strings, the one with maximum length is "ba", whose length is 2. Therefore, the answer is 2.

Sample input 2	Sample output 2
10 1 aaaaaaaaaa 1 10	1

Sample input 3	Sample output 3
17 5 deabcabcabdeadede 1 8 1 10 4 9 2 16 1 17	3 2 3 1 2