J – Repetitions

Time limit: 10 s *Memory limit: 512 MiB*

Bob is an aspiring avant-garde writer. He disdains the use of spaces, punctuation, capital letters and the like; hence, his stories are nothing but long strings of lowercase letters of the English alphabet. Critics have also noted that his style is marked by a certain fondness for *repetitions*, in the sense that it sometimes happens that two instances of the same substring appear in his story twice in a row, without any other intervening characters.

Bob has submitted his latest masterpiece, a string which happens to be n characters long, to q different literary magazines in the hopes that at least one of them might be willing to publish it. The response was more favourable than he had dared to hope. The editors of all q magazines have expressed willingness to publish some part (i.e. a substring) of his story, but on the condition that he identify the longest repetition (i.e. a shorter substring appearing twice in a row) within that part of the story. The editors intend to remove that part to prevent the story from being too boring. Now Bob needs your help to answer these queries from the editors.

Write a program that, given a string of n letters, $s[1]s[2]\ldots s[n]$, answers q queries of the form "given a_i and b_i, how long is the longest string t for which tt appears as a substring of $s[a_i]s[a_i+1]\ldots s[b_i-1]s[b_i]$, and where does the leftmost such occurrence begin?"

Input data

The first line contains two integers, n and q. The second line contains the string s, which is n characters long; all these characters are lowercase letters of the English alphabet. The remaining q lines describe the queries; the i-th of these lines contains the integers a_i and b_i, separated by a space.

Input limits

- $1 \leq n \leq 10^6$
- $1 \leq q \leq 100$
- $1 \leq a_i \leq b_i \leq n$ for each $i = 1, 2, \ldots, q$

Output data

Output q lines; the i-th of these lines must contain two space-separated integers ℓ_i and c_i. ℓ_i should be the length of the longest string t for which tt appears as a substring in $s[a_i]s[a_i+1]\ldots s[b_i-1]s[b_i]$, and c_i should be the index at which the leftmost repetition of this length begins, i.e. the smallest integer such that $a_i \leq c_i$, $c_i + 2\ell_i - 1 \leq b_i$ and $s[c_i]\ldots s[c_i+\ell_i-1] = s[c_i+\ell_i]\ldots s[c_i+2\ell_i-1]$. (If $\ell_i = 0$, then $c_i = a_i$ by definition.)

Example

Input	Output
10 4 cabaabaaca 4 8 1 9 5 9 8 10	1 4 3 2 1 7 0 8

Comment

The four queries in the above example refer to the substrings **a**abaa, c**aba**abaac, ab**a**ac, and aca; the part shown in bold is the substring referred to by the result of that query (a substring of length ℓ_i, beginning at index c_i). In the last query there is no repetition, so $\ell_4 = 0$.