Problem E. Expected Characters

Source file name:	strings.c, strings.cpp, strings.java
Input:	Standard
Output:	Standard
Author(s):	Juan Pablo Marín Rosas

A string S is traditionally a variable where a sequence of characters is stored. There are several operations you can do to strings, one of those operations that is of importance to us is rotation: a string $T = uv$ where u and v are also strings is said to be a rotation of S if $S = vu$. For example, if $S = abc$ the string bca is a rotation of abc, where $u = bc$ and $v = a$.

As you can see if a string S has N characters in the sequence then there will be N rotations. For example, if $S = abc$ the 3 rotations are: abc, bca, cab. Toby is a very curious person and after realizing this fact, he found that an $N \times N$ matrix containing all the rotations of S can be created. The Matrix Toby creates is as follows: For the $k - th$ row in the matrix take the substring of S that starts on 0 and ends on $k - 1$ as v, and the substring that starts on k and ends on $N - 1$ as u then the string $T = uv$ is a rotation of S. An example of this matrix using $S = abc$ is:

$$abc$$
$$bca$$
$$cab$$

Toby has challenged you to find a quick way to determine given the string S and a Q number of queries where each query asks for the expected character in position i, j of the matrix built of the rotations of S.

Input

The input consists of several test cases. The first line of each test case contains the strings S and the number of queries Q separated by a space. Each of the next Q lines contains the numbers i, and j of the query that needs to be answered. You may assume that S will have no more than 10^6 characters and that $1 \le Q \le 10^6$. S contains only lowercase characters and no white spaces. The end of the test cases is given by the end of file (EOF).

Output

For each test case your program should print exactly Q lines, each line containing a single character which is the answer to the query.

Example

Input	Output
abc 3	a
1 1	c
2 2	b
3 3	a
bca 3	b
3 1	c
1 1	i
2 1	a
acmicpc 5	c
2 3	c
4 5	c
3 3	
1 5	
4 2	