Cyclic Vigenere Cipher (cipher)

The Vigenere Cipher is a well known cipher in cryptography. The encryption of a string s, with the help of a key-string t, both having the same lengths, is done by the following way:
Let e be the final string that will be obtained by the encryption of string s using the key t. The string e will have the same length as the previous 2 strings, and will respect the following property:

$$e_i = \mathrm{chr}\Big(\big(\mathrm{ord}(s_i) + \mathrm{ord}(t_i)\big) \mod 26\Big)$$

where $\mathrm{ord}(c)$ is the ordinal number of the character c in the english alphabet ($\mathrm{ord}(\texttt{'a'}) = 0$, $\mathrm{ord}(\texttt{'z'}) = 25$ and so on), and $\mathrm{chr}(i)$ is the function that returns the character with the ordinal number i in the english alphabet.

Briefly, the encryption process can be described as an iterative addition, character by character, modulo 26, of the string t to the string s.

Let's define a modification over the Vigenere Cipher, which we will refer to as: “Cyclic Vigenere”, the following way:

The encryption of string s with key-string t is possible only when the length of string t divides the length of string s. The result of the encryption is string e, defined as:

$$e_i = \mathrm{chr}\Big(\big(\mathrm{ord}(s_i) + \mathrm{ord}(t_{i \mod |t|})\big) \mod 26\Big)$$

where $|t|$ denotes the length of the t string, $i \mod |t|$ is the remainder of i by $|t|$, and the indexing of the strings starts at 0.

Briefly, the encryption process can be described as an iterative addition, character by character, modulo 26, of the repetition of the string t to the string s.

In this task, K key-strings are given, and N strings that need to be encrypted. For any string s_i from the N strings, print the index of the key-string t_j that generates the minimal possible lexicographic encryption e of the string s_i. If there are multiple answers, the one with the smallest index must be printed. If there is no answer, print -1.

> ☞ Among the attachments of this task you may find a template file `cipher.*` with a sample incomplete implementation.

Input

The first line of input contains N and K.

The next K lines contain the key-strings $t_0, t_1, \ldots, t_{K-1}$.

The next N lines contain the strings $s_0, s_1, \ldots, s_{N-1}$.

Output

Output N lines. The i-th of them must contain the index of the key-string t_j that generates the minimal possible lexicographic encryption e of the string s_i. The indexes are considered to start from 1.

If there are multiple answers for any of the strings, print the minimal possible index of a key-string t_j.

If there is no such key-string, print -1.

Constraints

- $1 \le N \le 2 \cdot 10^5$.
- $1 \le K \le 2 \cdot 10^5$.
- The total lengths of all strings and key-strings is at most $4 \cdot 10^5$.
- Each string contains only lowercase English letters.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points) Examples.

– **Subtask 2** (20 points) The total length of all strings and key-strings multiplied by K is at most 10^6.

– **Subtask 3** (50 points) The length of each key-string divides the length of each string.

– **Subtask 4** (30 points) No additional limitations.

Examples

input	output

`3 6`
`are`
`mere`
`ana`
`oare`
`cinestie`
`nimeni`
`anaana`
`plebi`
`nusestie`
	`3`
	`-1`
	`4`

Explanation

In the sample, for the first string `anaana`, the key-string `are` produces `aeeaee`, the key-string `ana` produces `aaaaaa`, and the key-string `nimeni` produces `nvmeai`, the lexicographically minimal encryption result is `aaaaaa`, which is produced by the third key-string, so the answer is 3.

For the second string `plebi`, there is no suitable key-string, since it has length 5, which is not divisible by any of the lengths of the key-strings, thus the answer is -1.

For the third string `nusestie`, the key-string `mere` produces `zyjiexzi`, the key-string `oare`, produces `bujigtzi`, and the key-string `cinestie` produces `pcfikmqi`, the lexicographically minimal encryption result is `bujigtzi`, which is produced by the fourth key-string, so the answer is 4.