Problem Set 10 Strings 15-295 Spring 2020

A. Sensible Strings

time limit per test: 1 second
memory limit per test: 256 megabytes

For two strings s and t, define $\mathrm{LCP}(s,t)$ to be the length of their longest common prefix. For example, $\mathrm{LCP}(\mathrm{aaa},\mathrm{aab}) = 2$.

Given n string $s_1, s_2, \ldots, s_n$. Find a permutation $p = (p_1, p_2, \ldots, p_n)$ such that $\sum_{i=1}^{n-1} \left(\mathrm{LCP}(s_{p_i}, s_{p_{i+1}})\right)^2$ is maximized.

Input

The first line contains a single integer which is the number of strings n ($1 \le n \le 40000$). For the following n lines, each line contains a single string consisting of lowercase English letters which denotes s_i.

It is guaranteed that $1 \le |s_i| \le 10,000$ for all $1 \le i \le n$ and $\sum_{i=1}^{n} |s_i| \le 200,000$.

Output

Display a single integer which is the maximum value of $\sum_{i=1}^{n-1} \left(\mathrm{LCP}(s_{p_i}, s_{p_{i+1}})\right)^2$.

Example

input	output
4 a b abc bc	2

B. Repetitive Repetitions

time limit per test: 1 second
memory limit per test: 256 megabytes

For a binary string s and a positive integer n, define $\mathrm{REP}(s, n)$ to be the string formed by repeating s a total of n times.

Given a binary string s, output the largest n such that $s = \mathrm{REP}(t, n)$ for some binary string t.

Input
A single line which contains a single non-empty string consisting of 0 and 1 which is the given string s. The length of s is at most 2^{20}.

Output
Display a single integer, which is the largest n such that $s = \mathrm{REP}(t, n)$ for some binary string t.

Example

input	output
10011001100110011001	5

C. Password

time limit per test: 2 seconds
memory limit per test: 256 megabytes

Asterix, Obelix and their temporary buddies Suffix and Prefix has finally found the Harmony temple. However, its doors were firmly locked and even Obelix had no luck opening them.

A little later they found a string s, carved on a rock below the temple's gates. Asterix supposed that that's the password that opens the temple and read the string aloud. However, nothing happened. Then Asterix supposed that a password is some substring t of the string s.

Prefix supposed that the substring t is the beginning of the string s; Suffix supposed that the substring t should be the end of the string s; and Obelix supposed that t should be located somewhere inside the string s, that is, t is neither its beginning, nor its end.

Asterix chose the substring t so as to please all his companions. Besides, from all acceptable variants Asterix chose the longest one (as Asterix loves long strings). When Asterix read the substring t aloud, the temple doors opened.

You know the string s. Find the substring t or determine that such substring does not exist and all that's been written above is just a nice legend.

Input

You are given the string s whose length can vary from 1 to 10^6 (inclusive), consisting of small Latin letters.

Output

Print the string t. If a suitable t string does not exist, then print "`Just a legend`" without the quotes.

Examples

input	output
fixprefixsuffix	fix

input	output
abcdabc	Just a legend

D. Petr#

time limit per test: 2 seconds
memory limit per test: 256 megabytes

Long ago, when Petya was a schoolboy, he was very much interested in the Petr# language grammar. During one lesson Petya got interested in the following question: how many different continuous substrings starting with the s_{begin} and ending with the s_{end} (it is possible $s_{begin} = s_{end}$), the given string t has. Substrings are different if and only if their contents aren't equal, their positions of occurence don't matter. Petya wasn't quite good at math, that's why he couldn't count this number. Help him!

Input

The input file consists of three lines. The first line contains string t. The second and the third lines contain the s_{begin} and s_{end} identificators, correspondingly. All three lines are non-empty strings consisting of lowercase Latin letters. The length of each string doesn't exceed 2000 characters.

Output

Output the only number — the amount of different substrings of t that start with s_{begin} and end with s_{end}.

Examples

input	output
round ro ou	1

input	output
codeforces code forca	0

input	output
abababab a b	4

input	output
aba ab ba	1

Note

In the third sample there are four appropriate different substrings. They are: ab, abab, ababab, abababab.

In the fourth sample identificators intersect.

E. Games on a CD

time limit per test: 4 seconds
memory limit per test: 512 megabytes

Several years ago Tolya had n computer games and at some point of time he decided to burn them to CD. After that he wrote down the names of the games one after another in a circle on the CD **in clockwise order**. The names were distinct, the length of each name was equal to k. The names didn't overlap.

Thus, there is a cyclic string of length $n \cdot k$ written on the CD.

Several years have passed and now Tolya can't remember which games he burned to his CD. He knows that there were g popular games that days. All of the games he burned were among these g games, and **no game was burned more than once**.

You have to restore any valid list of games Tolya could burn to the CD several years ago.

Input

The first line of the input contains two positive integers n and k ($1 \le n \le 10^5$, $1 \le k \le 10^5$) — the amount of games Tolya burned to the CD, and the length of each of the names.

The second line of the input contains one string consisting of lowercase English letters — the string Tolya wrote on the CD, split in arbitrary place. The length of the string is $n \cdot k$. It is guaranteed that the length is not greater than 10^6.

The third line of the input contains one positive integer g ($n \le g \le 10^5$) — the amount of popular games that could be written on the CD. It is guaranteed that the total length of names of all popular games is not greater than $2 \cdot 10^6$.

Each of the next g lines contains a single string — the name of some popular game. Each name consists of lowercase English letters and has length k. It is guaranteed that the names are distinct.

Output

If there is no answer, print "NO" (without quotes).

Otherwise, print two lines. In the first line print "YES" (without quotes). In the second line, print n integers — the games which names were written on the CD. You should print games in the order they could have been written on the CD, it means, **in clockwise order**. You can print games starting from any position. Remember, that no game was burned to the CD more than once. If there are several possible answers, print any of them.

Examples

input	output
3 1 abc 4 b a c d	YES 2 1 3

input	output
4 2 aabbccdd 4 dd ab bc cd	NO

F. String Transformation

time limit per test: 2 seconds

memory limit per test: 256 megabytes

Let s be a string whose length equals n. Its characters are numbered from 0 to $n - 1$, i and j are integers, $0 \le i < j < n$. Let's define function f as follows:

$f(s, i, j) = s[i + 1 \ldots j - 1] + r(s[j \ldots n - 1]) + r(s[0 \ldots i])$.

Here $s[p \ldots q]$ is a substring of string s, that starts in position p and ends in position q (inclusive); "+" is the string concatenation operator; $r(x)$ is a string resulting from writing the characters of the x string in the reverse order. If $j = i + 1$, then the substring $s[i + 1 \ldots j - 1]$ is considered empty.

You are given two strings a and b. Find such values of i and j, that $f(a, i, j) = b$. Number i should be maximally possible. If for this i there exists several valid values of j, choose the minimal j.

Input

The first two input lines are non-empty strings a and b correspondingly. Each string's length does not exceed 10^6 characters. The strings can contain any characters with ASCII codes from 32 to 126 inclusive.

Output

Print two integers i, j — the answer to the problem. If no solution exists, print "`-1 -1`" (without the quotes).

Examples

input	output
Die Polizei untersucht eine Straftat im IT-Bereich. untersucht eine Straftat.hciereB-TI mi ieziloP eiD	11 36

input	output
cbaaaa aaaabc	4 5

input	output
123342 3324212	-1 -1