Problem G. ABACABA Matching

Input file:	`matching.in`
Output file:	`matching.out`
Time limit:	2 seconds
Memory limit:	256 megabytes

Consider a permutation of lowercase English letters: $P = \{p_1, p_2, \ldots, p_{26}\}$. Using P, you can generate the following sequence of strings:

$$\begin{aligned} S_1 &= p_1 \\ S_2 &= S_1 + p_2 + S_1 \\ S_3 &= S_2 + p_3 + S_2 \\ &\vdots \\ S_{26} &= S_{25} + p_{26} + S_{25} \end{aligned}$$

It is easy to show that the length of S_{26} is $2^{26} - 1$ letters. The beginning of S_{26} looks like $p_1p_2p_1p_3p_1p_2p_1 \ldots$.

You are given a string T consisting of lowercase English letters. For a fixed permutation P, you can obtain S_{26} and then substitute some of the letters in T by other letters so that the resulting string becomes a substring of S_{26}. Your goal is to minimize the number of letters that you must replace in T by choosing the appropriate permutation P.

Input

The only line of input contains the string T ($1 \le |T| \le 20\,000$) consisting of lowercase English letters.

Output

On the first line, print the minimal number of letters that should be replaced. On the second line, print the position in string S_{26} where the resulting substring starts (indices start from 1). On the third line, print the permutation P.

Examples

matching.in	matching.out
baca	0 2 abcdefghijklmnopqrstuvwxyz
bcdbaaac	3 2 cbdaefghijklmnopqrstuvwxyz