Absent

Computer Science Society
Programming Contest
Spring 2009

Given a string of symbols, it's natural to look it over and see what substrings are present. In this problem, you are given a string and asked to consider what substrings are absent. Of course, a given string has finite length and therefore only finitely many substrings, so there are always infinitely many strings that don't appear as substrings of a given string. We'll seek to find the lexicographically least string that is absent from the given string.

Input Format

Each line of input contains a string x over the alphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$. x may be the empty string, as shown in the second line of the input sample below, or a nonempty string.

Output Format

For each input string x, find and output the lexicographically least string s over the alphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ such that s is not a substring of x; i.e. s is *absent* from x. Since the empty string is a substring of every string, your output s is necessarily nonempty. Recall that a string is lexicographically less than another string if it is shorter or is the same length and alphabetically less; e.g. $\mathtt{b} < \mathtt{aa}$, $\mathtt{abc} < \mathtt{acb}$. Format each line of output to show s and x, as shown in the output sample below.

Input Sample

```
bcabacbaa

aaabacbbcacc
```

Output Sample

```
bb is absent from bcabacbaa
a is absent from
aac is absent from aaabacbbcacc
```