Problem J

Inverse Common Superstring

Given a set of string $S = \{S_1, S_2, \cdots, S_n\}$, a common superstring R of the set S is a string such that each string in S appears as a substring in R. For example, let S be {"abb", "baab", "bbc"}, then one possible common superstring R of S is "abbbaabbbc" which has the length of 10 characters. Notice that all strings in S appear as substring in R. To verify: "abb" appears in "[abb]baabbbc", "baab" appears in "abb[baab]bbc", and "bbc" appears in "abbbaab[bbc]". The string "abbbaabbbc" is also a common superstring of S; you can verify it by yourself.

Among all possible common superstrings, usually the shortest common superstring is more attractive. It has many real-world application such as sparse matrix compression, DNA sequencing, and many others. In the example above, the shortest common superstring will be "baabbc" with the length of 6 characters. To verify: "aab" appears in "b[aab]bc", "baab" in "[baab]bc", and "bbc" in "baa[bbc]".

Unfortunately, the problem of finding the shortest common superstring is known to be NP-hard, i.e. up to this moment, there is no known polynomial-time algorithm to solve the problem.

The inverse problem of finding the shortest common superstring will be: given a string R, find the set of string S such that R is the shortest common superstring of S. Of course this inverse problem is very easy and trivial! The set S can simply contains a single string which equals to R (notice that a string is also a substring to the string itself).

Now, you are going to solve a more challenging problem. Given a string R, find the lexicographically (alphabetically) smallest string which does NOT appear as a substring in R. To simplify the problem, a string is defined as a non-empty sequence of only lowercase alphabetical character (a-z). For example, let the string R be "icpc", then the lexicographically smallest string which does not appear as substring in R is "a".

String $S = S_1S_2S_3\cdots$ is lexicographically smaller than string $T = T_1T_2T_3\cdots$ if one of the following is true:

- $|S| < |T|$ and $S_i = T_i$ for all $1 \le i \le |S|$, or
- There exists an index i where $S_i < T_i$ and $S_j = T_j$ for all $1 \le j < i$.

Input

The first line contains a string which length between 1 and 1000, inclusive. The given string contains only lowercase alphabetical character (a-z).

Output

The output contains the smallest lexicographical string which is NOT a substring of the input string, in a line. The output string should contain only lowercase alphabetical character.

Sample Input	Output for Sample Input
icpc	a
jakarta	aa