

Problem A. Broken Keyboard

Time limit 1000 ms

Mem limit 262144 kB

Recently Polycarp noticed that some of the buttons of his keyboard are malfunctioning. For simplicity, we assume that Polycarp's keyboard contains 26 buttons (one for each letter of the Latin alphabet). Each button is either working fine or malfunctioning.

To check which buttons need replacement, Polycarp pressed some buttons in sequence, and a string s appeared on the screen. When Polycarp presses a button with character c , one of the following events happened:

- if the button was working correctly, a character c appeared at the end of the string Polycarp was typing;
- if the button was malfunctioning, **two** characters c appeared at the end of the string.

For example, suppose the buttons corresponding to characters a and c are working correctly, and the button corresponding to b is malfunctioning. If Polycarp presses the buttons in the order a, b, a, c, a, b, a , then the string he is typing changes as follows: $a \rightarrow abb \rightarrow abba \rightarrow abbac \rightarrow abbaca \rightarrow abbacabb \rightarrow abbacabba$.

You are given a string s which appeared on the screen after Polycarp pressed some buttons. Help Polycarp to determine which buttons are working correctly for sure (that is, this string could not appear on the screen if any of these buttons was malfunctioning).

You may assume that the buttons don't start malfunctioning when Polycarp types the string: each button either works correctly throughout the whole process, or malfunctions throughout the whole process.

Input

The first line contains one integer t ($1 \leq t \leq 100$) — the number of test cases in the input.

Then the test cases follow. Each test case is represented by one line containing a string s consisting of no less than 1 and no more than 500 lowercase Latin letters.

Output

For each test case, print one line containing a string *res*. The string *res* should contain all characters which correspond to buttons that work correctly **in alphabetical order, without any separators or repetitions**. If all buttons may malfunction, *res* should be empty.

Examples

Input	Output
4 a zzaaz ccff cbddbb	a z bc