

Increasing Subword

Timelimit: 1 sec

Problem description

We say that a string is non-decreasing if it does not contain two consecutive characters c_1 and c_2 where $c_1 > c_2$. Our strings will consist of lowercase letters and the less-than relation is just the alphabetical order. For example, *hello* is not non-decreasing because it contains the consecutive characters *h* and *e*. The string *ello* on the other hand is indeed non-decreasing. The task here is to find the longest non-decreasing substring of a given word.

Input

A number n ($n \leq 100$) followed by n lines containing strings over the alphabet $\{a, \dots, z\}$ of length at most 100.

Output

Output for each string the longest substring that is non-decreasing. If there are several substrings of maximal length output the left most.

Sample input/output

Input	Output
4	
abracadabra	abr
hello	ello
abcacd	abc
valpollicella	cell