

Problem B. Change a Password

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 512 megabytes

Password authentication is used in a lot of facilities. The office of JAG also uses password authentication. A password is required to enter their office. A password is a string of N digits ‘0’-‘9’. This password is changed on a regular basis. Taro, a staff of the security division of JAG, decided to use the following rules to generate a new password from an old one.

1. The new password consists of the same number N of digits to the original one and each digit appears at most once in the new password. It can have a leading zero. (Note that an old password may contain same digits twice or more.)
2. The new password maximizes the difference from the old password within constraints described above. (Definition of the difference between two passwords is described below.)
3. If there are two or more candidates, the one which has the minimum value when it is read as an integer will be selected.

The difference between two passwords is defined by $\min(|a - b|, 10^N - |a - b|)$, where a and b are the integers represented by the two passwords. For example, the difference between “11” and “42” is 31, and the difference between “987” and “012” is 25.

Taro would like to use a computer to calculate a new password correctly, but he is not good at programming. Therefore, he asked you to write a program. Your task is to write a program that generates a new password from an old password.

Input

The input consists of a single test case. The first line of the input contains a string S which denotes the old password. You can assume that the length of S is no less than 1 and no greater than 10. Note that old password S may contain same digits twice or more, and may have leading zeros.

Output

Print the new password in a line.

Examples

standard input	standard output
201	701
512	012
99999	49876
765876346	265874931