D. Name

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Everything got unclear to us in a far away constellation Tau Ceti. Specifically, the Taucetians choose names to their children in a very peculiar manner.

Two young parents `abac` and `bbad` think what name to give to their first-born child. They decided that the name will be the permutation of letters of string s. To keep up with the neighbours, they decided to call the baby so that the name was lexicographically strictly larger than the neighbour's son's name t.

On the other hand, they suspect that a name tax will be introduced shortly. According to it, the Taucetians with lexicographically *larger* names will pay *larger* taxes. That's the reason `abac` and `bbad` want to call the newborn so that the name was lexicographically strictly larger than name t and lexicographically minimum at that.

The lexicographical order of strings is the order we are all used to, the "dictionary" order. Such comparison is used in all modern programming languages to compare strings. Formally, a string p of length n is lexicographically less than string q of length m, if one of the two statements is correct:

- $n < m$, and p is the beginning (prefix) of string q (for example, "`aba`" is less than string "`abaa`"),
- $p_1 = q_1, p_2 = q_2, \ldots, p_{k-1} = q_{k-1}, p_k < q_k$ for some k $(1 \le k \le min(n, m))$, here characters in strings are numbered starting from 1.

Write a program that, given string s and the heighbours' child's name t determines the string that is the result of permutation of letters in s. The string should be lexicographically strictly more than t and also, lexicographically minimum.

Input

The first line contains a non-empty string s $(1 \le |s| \le 5000)$, where $|s|$ is its length. The second line contains a non-empty string t $(1 \le |t| \le 5000)$, where $|t|$ is its length. Both strings consist of lowercase Latin letters.

Output

Print the sought name or `-1` if it doesn't exist.

Examples

input
aad aac
output
aad

input
abad bob
output
daab

input
abc defg

output
-1

input
czaaab abcdef

output
abczaa

Note

In the first sample the given string s is the sought one, consequently, we do not need to change the letter order there.