Maximum Permutation

As a biologist who studies DNA, Dan finds *string processing* techniques incredibly helpful in his research. Today, he encounters the following problem.

Given two strings w and s consisting of lowercase English letters, find the permutation of w which occurs the maximum number of times as a substring in s.

If more than one permutation occurs the maximum number of times, print the lexicographically smallest one. Print -1 if no permutation of w occurs as a substring of s.

Complete the function `maximumPermutation` which takes two strings w and s and returns a string denoting the permutation of w occurring the maximum number of times as a substring in s, or the string "`-1`" if no permutation of w occurs as a substring of s.

Input Format

The first line contains a single integer t denoting the number of test cases.

The first line of each test case contains a single string w. The second line of each test case contains a single string s.

Constraints

- $1 \leq t \leq 3$
- $1 \leq |w| \leq 20000$
- $1 \leq |s| \leq 200000$
- $|w| \leq |s|$
- All strings in the input consists of lowercase English letters.

Subtasks

- For $\sim 42\%$ of the maximum score, $|w| \leq 1000$

Output Format

For each test case, print one line containing the required string as mentioned in the statement.

Sample Input 0

```
2
aab
aababaa
xabd
absdasfsatsads
```

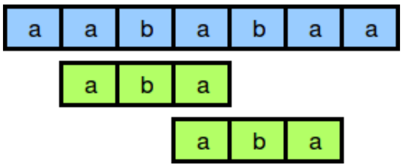
Sample Output 0

```
aba
-1
```

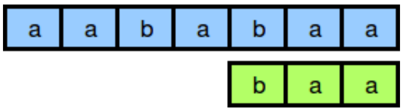
Explanation 0

In the first test case: `aab` has 3 permutations: `aba`, `aab` and `baa`.

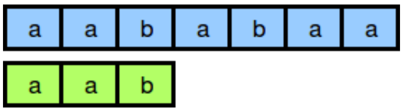
`aba` occurs twice in s:



`baa` occurs just once in s:



`aab` also occurs just once:



`aba` occurs the most number of times in s.