LAEncoding

Petr wishes to send a message to his friend ACRush. Unfortunately, due to repeated use of his keyboard some of the keys of his keyboard do not work. ACRush and Petr during their next meeting invent a method to send the message by using the limited alphabet (working keys).

Given an alphabet set, the first character in the ASCII ordering of the alphabet set is called special LA character. LA encodable string for an alphabet set, can be of any length but the special LA character cannot appear as the starting letter of it. The message sent by Petr, is a string of uppercase characters (`A-Z`) and is a LA encodable string for the alphabet set `A-Z`.

To encode the message, Petr first writes down all possible LA encodable strings for alphabet set `A-Z` in lexicographical order, and finds the position, say `*i* ', of his message, which is to be sent to ACRush. To send the message he must use only working keys. So he encodes the message to be sent. For this purpose, he writes down all possible LA encodable strings for the limited alphabet set (working keys) in lexicographical order and selects the string in this list that is in the position `*i* ' which he found earlier. As this encoded string (found at positon `*i* ') contains only the working keys, he can send it without any problem and can be decoded easily.

You will be given the message to be sent by Petr, and the list of non-working keys, your task is to print the encoded string which will be sent to ACRush.

Constraints:

- Length of message sent by Petr is between 1 and 12 both inclusive.
- Number of non-working keys will range from 1 to 24 both inclusive.
- String of lesser length comes first in lexicographical order. If the two strings are of same length, they are written in alphabetical order.

Input

First line contains number of test cases *t* . Each test case contains a line with two strings separated by space(s). First string is the message to be sent by Petr. Second string is the list of non-working characters (not separated by

spaces).

Output

Output a line for each test case, containing the encoded form of the string to be sent by Petr.

Sample Input

```
3
Z Z
CMBWJOJ BCMJOW
CMBWJOJ ADEZLXY
```

Sample Output

```
BA
QXXXHLN
UGMKSKM
```
