



## Problem E

(Program filename: E.CPP or E.PAS or E.DPR or E.java)

### Puzzle Out

The scientific committee members of the 26<sup>th</sup> ACM/ICPC, who design the contest problems, use the following encryption algorithm to communicate the problem drafts securely through the Internet. To encrypt a text, all occurrences of each letter is replaced with another letter (possibly itself), such that no two letters are encrypted to the same letter. Both original and encrypted texts consist of only upper-case letters and blanks. Blanks are not encrypted and are repeated exactly in the encrypted text. As an example, the string GSRH RH GSV URIHG HZNNKOV is the encrypted form of THIS IS THE FIRST SAMPLE according to the encryption table ( $A \rightarrow Z, B \rightarrow Y, C \rightarrow X, E \rightarrow Z \rightarrow A$ ).

A recipient of a problem draft has lost the encryption table, but he has a dictionary which includes all the possible words appearing in the problems. You are to help him set up a decryption table to enable him restore the original problem draft from the encrypted one. Given a dictionary of the original words used in the text, and the encrypted text, we want to find the right encryption table such that after decrypting the given encrypted text back to the original one, all words can be found in the dictionary.

### Input (filename: E.IN)

The first part of the input file is a dictionary of English words common to all test cases. The first line of the file is  $d$  ( $1 \leq d \leq 50000$ ); the number of words in the dictionary, followed by  $d$  lines each containing a word in the dictionary. The words in the dictionary are sorted in alphabetical order and all are in uppercase. Each word has at most 20 characters, but you can assume that sum of the length of all words in the dictionary is no more than 350,000. The next line contains a single integer  $t$  ( $1 \leq t \leq 10$ ), the number of test cases, followed by the input data for each test case. Each line has a string containing only uppercase letters and blank. You may assume that no line break is occurred in the middle of a word and there may be arbitrary number of blank characters at the end of each line. Maximum length of input lines is 80.

### Output (filename: E.OUT)

The output file contains exactly  $t$  lines, each corresponding to a test case. Each line should contain a single string of 26 characters which is the encryption of the string ABCDEFGHIJKLMNOPQRSTUVWXYZ according to the encryption table used in the test case. Letters in the output string should be in uppercase. It is possible that some letters do not appear in the encrypted text at all. In this case, put a \* mark in place of those letters not appearing in the decrypted version of the input text. If the test case has no solution, the output line should contain #NO SOLUTION#. If there is more than one possible encryption table for a test case, the output line should contain #More than one solution#.

## Sample Input

14  
BE  
CHANGE  
FIRST  
IN  
IS  
MUST  
SAMPLE  
SEE  
THE  
THIS  
TO  
WISH  
WORLD  
YOU  
4

GSRH RH GSV URIHG HZNRKV  
IZM BMVU SP UGP  
RGIANP IZM KPVG UZ VPP  
FA UGP KZWCQ  
XYZ ABCDEFG  
XYZY ABD

## Sample Output

Z\*\*\*VU\*SR\*\*ON\*\*K\*IHG\*\*\*\*\*  
TSRQP\*NGF\*\*CBAZ\*\*WVUM\*K\*I\*  
#No solution#  
#More than one solution#