

# A. Hockey

time limit per test: 2 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

Petya loves hockey very much. One day, as he was watching a hockey match, he fell asleep. Petya dreamt of being appointed to change a hockey team's name. Thus, Petya was given the original team name  $w$  and the collection of forbidden substrings  $s_1, s_2, \dots, s_n$ . All those strings consist of uppercase and lowercase Latin letters. String  $w$  has the length of  $|w|$ , its characters are numbered from 1 to  $|w|$ .

First Petya should find all the occurrences of forbidden substrings in the  $w$  string. During the search of substrings the case of letter shouldn't be taken into consideration. That is, strings "aBC" and "ABc" are considered equal.

After that Petya should perform the replacement of all letters covered by the occurrences. More formally: a letter in the position  $i$  should be replaced by any other one if for position  $i$  in string  $w$  there exist pair of indices  $l, r$  ( $1 \leq l \leq i \leq r \leq |w|$ ) such that substring  $w[l \dots r]$  is contained in the collection  $s_1, s_2, \dots, s_n$ , when using case insensitive comparison. During the replacement the letter's case should remain the same. Petya is not allowed to replace the letters that aren't covered by any forbidden substring.

Letter *letter* (uppercase or lowercase) is considered lucky for the hockey players. That's why Petya should perform the changes so that the *letter* occurred in the resulting string as many times as possible. Help Petya to find such resulting string. If there are several such strings, find the one that comes first lexicographically.

Note that the process of replacements is not repeated, it occurs only once. That is, if after Petya's replacements the string started to contain new occurrences of bad substrings, Petya pays no attention to them.

## Input

The first line contains the only integer  $n$  ( $1 \leq n \leq 100$ ) — the number of forbidden substrings in the collection. Next  $n$  lines contain these substrings. The next line contains string  $w$ . All those  $n + 1$  lines are non-empty strings consisting of uppercase and lowercase Latin letters whose length does not exceed 100. The last line contains a lowercase letter *letter*.

## Output

Output the only line — Petya's resulting string with the maximum number of letters *letter*. If there are several answers then output the one that comes first lexicographically.

The lexicographical comparison is performed by the standard  $<$  operator in modern programming languages. The line  $a$  is lexicographically smaller than the line  $b$ , if  $a$  is a prefix of  $b$ , or there exists such an  $i$  ( $1 \leq i \leq |a|$ ), that  $a_i < b_i$ , and for any  $j$  ( $1 \leq j < i$ )  $a_j = b_j$ .  $|a|$  stands for the length of string  $a$ .

## Examples

input
3 bers ucky elu PetrLoveLuckyNumbers t
output
PetrLovtTttttNumtttt

input
4 hello

party abefglghjdhfgj IVan petrsmatchwin a
output
petrsmatchwin

input
2 aCa cba abAcaba c
output
abCacba