

Single substitutions

In the land of Redsublia, they always write words with exactly one letter changed. All words are 6 letters long, made up of lower-case characters from a-z. So if the word was abcdef, they might write gbcdef or azcdef.

Of course, this means that sometimes when they write a word, it might be ambiguous.

In school, they are given some words to learn, and then tested on them. Of course, in the tests, the words are written with one letter changed for another. Sometimes, they are tested with words that don't exist, or are ambiguous. They are asked to find the lexicographically smallest word that matches out of the words that they have learned so far.

Bugsy is struggling at school. Please help him.

Input

The first line contains N, the number of test cases.

The next line contains L the number of lines in the input for this test-case.

Each line will either add a word to the dictionary

W <word>

Or suggest a word to look up in the dictionary so far.

Q <query>

Output

The first line should contain:

Test <test case>

Then for each line of input that is a query, print the (lexicographically smallest) word. If there are no possible words, print 0.

Example

Input:

```
1
6
W abcdef
Q bacdef
Q abcdef
W abadef
Q abcdeg
Q abgdef
```

Output:

Test 1
0
0
abcdef
abedef

Notes on example

In the first query, bacdef does not match abcdef as 2 characters have changed.

In the second query, abcdef does not match abcdef as 0 characters have changed.

In the third query, abcdeg matches abcdef as 1 character has changed - the last.

In the fourth query, abgdef matches both abcdef and abedef. abedef is lexicographically smaller.

Constraints

$1 \leq N \leq 10$

$1 \leq L \leq 70000$

words and queries are all 6 characters long, using lower-case a-z.