

## 2642 Spelling Be

It's a simple requirement your company has, really—every document should be spell-checked before it's sent out to a customer. Unfortunately, while word processing documents are easily spell-checked, your employees have not been checking email every time they send out a message. So you've come up with a little improvement. You are going to write a program that will check email on its way out. You will spell-check each message, and if you find any spelling errors, it will be returned to the sender for correction.

When you announced this plan, one of your coworkers fell off their chair laughing, saying that you couldn't possibly anticipate every name, technical acronym, and other terms that might appear in an email. Undaunted, however, you are going to test-run your code with an online dictionary and some sample emails you have collected.

### Input

The input consists of two sections, the dictionary and the emails. The first line of input specifies the number of words in the dictionary, followed by that many lines, with one word per line. There is no whitespace before, after, or in any words, although there may be apostrophes or hyphens in the words, which are considered part of the word (i.e. “bobs” is different than “bob's”). There will be no duplicate words. All words will be in lower case.

Following that are the emails. The first line of this section has the number of emails in the input. Following that line begins the first email. It has been preprocessed, so it consists of one word per line, with no punctuation (other than apostrophes and hyphens) or whitespace, and all words are in lower case.

The last word in the email is followed by a line with only ‘-1’. Each email will have at least one word.

### Output

For each email, you must either print:

Email *X* is spelled correctly.

where *X* begins with 1 and counts up. Or, if a word is found that is not in the dictionary, print out:

Email *X* is not spelled correctly.

followed by a list of unknown words in the order that you find them, one per line. If an unknown word is found multiple times, it should be printed multiple times.

There are no spaces between datasets. Following the output for the final dataset, print a line stating ‘End of Output’

### Sample Input

```
6
alice
am
bitterly
i
leaving
```

you  
1  
dear  
bob  
i  
am  
leaving  
you  
bitterly  
alice  
-1

### Sample Output

Email 1 is not spelled correctly.  
dear  
bob  
End of Output