**Uncle Grandpa and the dictionary**

**Note:** This problem will be graded for 1%.

Here comes the second problem in the challenge!

“You are given a list of words, each consists of only letters and . You have to process queries. For each query, you will be given a word , and you have to count how many words in the dictionary has as a prefix.”

You think this problem is hard? Haha yes it is because this is the graded problem, yet our dear Grandpa solved it in just 10080 mins. Can you do faster?

## Input

The first line contains two integers – the number of words in the dictionary and the number of queries.

Next lines, each contain a string is a word in the dictionary. It’s guaranteed that the string will only consists of letters and . The total length of all the given words is no more than

Next lines, each contain a string represent a query. It’s guaranteed that the string will only consists of letters and . The total length of all the query words is no more than

## Output

For each query, output a single integer is the number of words in the dictionary that has the corresponding query word as its prefix

## Examples

|  |  |
| --- | --- |
| Input (dictionary1.in) | Output (dictionary1.out) |
| 5 2  ababa  baaa  abaaa  abb  bab  aba  ba | 2  2 |

## Explanation:

For the 1st query, there are exactly 2 words in the dictionary has as its prefix, they are and .

For the 2nd query, there are exactly 2 words in the dictionary has as its prefix, they are and .

## Important Note: For the purpose of learning, you are highly encouraged to implement a Binary Tree to solve this problem. We could have made non-intended solutions impossible by tweaking the way the input is given, but we decided not to do it so that the problem is as “realistic” as possible. You are strictly forbidden from using Hash table and all its variants, API Tree and all its variants. (Constructing your own tree is fine).

## Note:

1. A skeleton file has been given to help you. You should not create a new file or rename the file provided. You should develop your program using this skeleton file.
2. You are free to define your own helper methods and classes (or remove existing ones) if it is suitable but you must put all the new classes, if any, in the same skeleton file provided

## Skeleton File

You can find the skeleton file Dictionary.java in the lab package.