Forbidden Anagrams

Valerie Pie is interested in something she learned in her combinatorics class called anagrams. An anagram is a rearrangement of letters to form a new word or phrase. The words and phrases don't need to make sense, but they can. An example anagram is Travis Meade can become Varied Meats. Valerie wants to make anagrams of her name, but she realized that some anagrams contain bad words as substrings. An anagram is forbidden if it contains at least one bad word as a substring. Help Valerie count the number of non-forbidden anagrams of different words for phrases.

Problem

Valerie will give a word or phrase, and a list of forbidden words. Compute the number of non-forbidden anagrams.

Input Specification

The input will begin with a single non-empty string of at most 12 lower-case, Latin characters, **s**, representing the phrase, of which Valarie want the anagram count. The next line will contain a single non-negative integer, **n** (**n** ≤ 20), representing the number of bad words. The remaining **n** lines will contain one non-empty string of at most 12 lower-case Latin characters. Each of these strings represents a bad word.

Output Specification

The output will consist of a single integer, **k**, representing the number of non-forbidden anagrams of **s**.

Sample Input	Sample Output
abad 1 dab	10
same 2 sam me	17
valeriepie 1 peeve	302040

Explanation

In the *first case* when we have the following possibilities

aabd

aadb

abad

abda

adba

baad

bada

bdaa

daab

dbaa

adab and daba are both forbidden

In the *second case* we have the following possibilities

aems
aesm
amse
asem
eams
easm
emas
emsa
esma
maes
mase
msae
msea
saem
seam
sema
smae

I don’t think I should upload a file with all 300k+ possibilites...

Grading Details

Read/Write from/to standard input/output – 10 points

Good comments, whitespace, and variable names – 15 points

No extra input output (e.g. input prompts, “Please enter the number of words”) – 10 points

Use a recursive function to build up words – 15 points

Your program will be tested on 10 test cases – 5 points each

No points will be awarded to programs that do not compile using gcc -std=gnu11 (gnu “eleven”).

Sometimes a requested technique will be given, and solutions without the requested technique will have their maximum points total reduced. For this problem you must use recursion. ***Without this your program will earn at most 50 points!***

Any case that causes your program to return a non-zero error return code will be treated as completely wrong. Additionally any case that takes longer than the maximum allowed time (the max of {5 times my solution, 10 seconds}) will also be treated as wrong.

No partial credit will be awarded for an incorrect case.