

Задача А. Множественный поиск

Имя входного файла: `search4.in`
Имя выходного файла: `search4.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Дано множество строк S и строка t . Требуется для каждой строки $p \in S$ определить, встречается ли она в t как подстрока.

Формат входных данных

Первая строка входного файла содержит целое число n — мощность S ($1 \leq n \leq 10^6$). Следующие n строк содержат по одной строке из S . Сумма длин всех строк из S не превосходит 10^6 . Последняя строка входного файла содержит t ($1 \leq t \leq 10^6$). Все строки состоят из прописных латинских букв.

Формат выходных данных

Для каждой строки из S выведите «YES», если она встречается в t и «NO» в противном случае. Строки нумеруются в порядке появления во входном файле.

Примеры

<code>search4.in</code>	<code>search4.out</code>
3	YES
abc	NO
abcdr	YES
abcde	
xabcdef	

Задача В. Под-бор

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Бором называется подвешенное дерево, на каждом из рёбер которого написано по символу, причём символы, написанные на рёбрах, выходящих из общей вершины-родителя, различны. Будем называть направление от родителя к детям “вниз”. Назовем *вхождением* строки s в бор такую вершину бора, от которой можно пройти несколько шагов вниз таким образом, что встретившиеся символы образуют строку s .

Даны бор и несколько строк, найдите сумму количеств вхождений этих строк в этот бор.

Формат входных данных

В первой строке ввода записано единственное число n , ($1 \leq n \leq 100\,000$) — количество вершин бора. В следующих n строках описаны вершины бора. В $(i + 1)$ -й строке описаны дети i -й вершины: число k_i ее детей, затем k_i пар из номера вершины-ребёнка и символа, написанного на соответствующем ребре. Номер родителя всегда меньше номера ребёнка; корнем бора является вершина номер 1.

В $(n + 2)$ -й строке записано количество m ($1 \leq m \leq 100\,000$) строк для поиска. В следующих m строках перечислены сами строки. Входные строки непусты, а их суммарная длина не превышает 100 000 символов.

Все символы, написанные на рёбрах, а также все символы, составляющие строки — маленькие латинские буквы.

Формат выходных данных

Выведите одно число — сумму количеств вхождений.

Примеры

стандартный ввод	стандартный вывод
7 2 2 a 4 b 2 3 a 6 b 0 1 5 b 1 7 b 0 0 4 b bb bbb bb	9

Задача С. Динамический поиск подстроки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 6 секунд
Ограничение по памяти: 256 мегабайт

Словарь — это множество слов. Вы должны уметь обрабатывать запросы трех типов:

- «+ word» — добавить слово **word** в словарь, если оно в нем не присутствует.
- «- word» — удалить слово **word** из словаря, если оно там присутствует.
- «? text» — вычислить суммарное количество вхождений всех слов из словаря в текст **text**, при этом, если слово входит в текст несколько раз, то необходимо учесть каждое вхождение.

Гарантируется, что любое слово или текст являются непустыми строками, состоящими из букв **a**, **b** и **c**, суммарная длина которых не превосходит L . Однако, для упрощения задачи перед выполнением каждого запроса необходимо поступить следующим образом: пусть x обозначает ответ на последний запрос **?**, или 0, если таких запросов еще не было. Тогда необходимо очередную строку (**word** или **text**) циклически сдвинуть x раз. Напомним, что циклическим сдвигом строки $s = s_0s_1 \dots s_{|s|}$ называется строка $s' = s_1 \dots s_{|s|}s_0$.

Формат входных данных

В первой строке дано одно число Q — число запросов. В следующих Q строках находятся запросы. Суммарная длина строк во всех запросах не превосходит L ($L \leq 5\,000\,000$)

Формат выходных данных

Для каждого запроса «**?**» выведите одно число — ответ на него.

Примеры

стандартный ввод	стандартный вывод
11	0
+ a	6
+ a	5
- a	7
- ab	
? abca	
+ ab	
+ a	
? abaaabb	
? aaabbab	
+ baa	
? babaca	

Задача D. Вирусы

Имя входного файла: `virus.in`
Имя выходного файла: `virus.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Комитет По Исследованию Бинарных Вирусов обнаружил, что некоторые последовательности единиц и нулей являются кодами вирусов. Комитет изолировал набор кодов вирусов. Последовательность из единиц и нулей называется безопасной, если никакой ее подотрезок (т.е. последовательность из соседних элементов) не является кодом вируса. Сейчас цель комитета состоит в том, чтобы установить, существует ли бесконечная безопасная последовательность из единиц и нулей.

Формат входных данных

Первая строка входного файла `virus.in` содержит одно целое число N , равное количеству всех вирусных кодов. Каждая из следующих n строк содержит непустое слово, составленное из символов 0 и 1 — код вируса. Суммарная длина всех слов не превосходит 30000.

Формат выходных данных

Первая и единственная строка выходного файла должна содержать слово:

- **ТАК** — если бесконечная, безопасная последовательность из нулей и единиц существует;
- **NIE** — в противном случае.

Примеры

<code>virus.in</code>	<code>virus.out</code>
3 01 11 00000	NIE
3 011 11 0000	ТАК

Задача Е. Пароли

Имя входного файла: `cofnsubstrings.in`
Имя выходного файла: `cofnsubstrings.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Павлу очень часто нужно придумывать различные новые секретные пароли. Каждый пароль должен иметь длину ровно l символов и может состоять только из строчных латинских букв.

Для того, чтобы не забывать придуманные пароли, Павел решил прятать в них некоторые ключевые слова. Для этого он подготовил множество из n ключевых слов, которые должны встречаться в пароле как подстроки. При этом он решил, что в придуманном пароле должны встречаться ровно s из ключевых слов, каждое возможно несколько раз.

Павел просит вас посчитать сколько всего существует различных подходящих паролей.

Формат входных данных

Первая строка содержит единственное целое число n ($1 \leq n \leq 6$) — количество ключевых слов.

Следующие n строк содержат по одному непустому ключевому слову в каждой строке. Каждое слово состоит из не более чем 50 символов.

Последняя строка содержит два целых числа s и l ($0 \leq s \leq n, 1 \leq l \leq 50$) — количество требуемых слов из множества и длина пароля, соответственно.

Формат выходных данных

Выведите количество паролей по модулю 1 000 000 009.

Примеры

<code>cofnsubstrings.in</code>	<code>cofnsubstrings.out</code>
4 a aa aaa aaaa 2 3	50
1 abcdefgh 0 7	31810104

Задача F. Склейка строк

Имя входного файла: `joined-string.in`
Имя выходного файла: `joined-string.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Вам дан набор слов. Найдите наиболее короткую строку, которая содержит все строки набора как подстроки. Если существует несколько возможных ответов, выведите первый в лексикографическом порядке.

Формат входных данных

В первой строке содержится число n ($1 \leq n \leq 12$) — количество строк в наборе.

Следующие n строк являются строками набора. Гарантируется, что они состоят только из прописных латинских букв, они не пустые, и что их длина не превосходит 50.

Формат выходных данных

Выведите одну строку, являющаяся ответом.

Примеры

<code>joined-string.in</code>	<code>joined-string.out</code>
2 BAB ABA	ABAB
4 ABABA AKAKA AKABAS ABAKA	ABABAKAKABAS
6 AAA BBB CCC ABC BCA CAB	AAABBBACABCCC
8 OFG SDOFGJTILM KBWNF YAAPO AWX VSEAWX DOFGJTIL YAA	KBWNFSDOFGJTILMVSEAWXYAAPO
2 STRING RING	STRING

Задача G. Подозрительные строки

Имя входного файла:	<code>strings.in</code>
Имя выходного файла:	<code>strings.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Вы работаете в компании, специализирующейся в технологиях, связанных с интернетом, и ваш текущий проект — спам-фильтр. Фильтр определяет, содержит ли строка спам, используя *словарь спам-слов*. Если в строке содержится хотя бы одно слово из этого словаря как подстрока, фильтр считает, что исходная строка подозрительна.

Вы стали решать более интересную задачу: сколько существует различных подозрительных строк длины n , состоящих из строчных букв латинского алфавита для данного словаря спам-слов. Найдите ответ по модулю 10 000.

Формат входных данных

В первой строке содержатся два числа n и k ($1 \leq n < 2^{31}, 1 \leq k \leq 10$) — требуемая длина слов и количество слов в словаре спам-слов соответственно.

Следующие k строк являются строками словаря. Гарантируется, что они состоят из строчных латинских букв, они не пустые, и их длина не превосходит 10 символов.

Формат выходных данных

Выведите ответ по модулю 10 000.

Примеры

strings.in	strings.out
1 1 x	1
2 2 ab bb	2
5 2 ab bb	6350
5 2 aab bba	4054
5 9 xxxxxx xxx x уухуу xxхуxxх у ух ху zzzzzzzzzz	8752
2147483647 10 aaaaaaaaaa bbbbbbbbbb cccccccccc dddddddddd eeeeeeeeee ffffffffff gggggggggg hhhhhhhhhh xxxxxxxxxx zzzzzzzzzz	5040