

## Задача А. Три строки

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

В новом учебном году школа, в которой обучается 7-классница Андреева Анна Андреевна, решилась на беспрецедентные меры по повышению знаний учащихся: каждый преподаватель школы должен помимо основных занятий прочесть хотя бы один годовой спецкурс. Темы спецкурсов преподаватели подобрали самые разные и интересные, а чтобы все учащиеся окончательно побросали свои внешкольные дела и с увлечением слушали новые курсы, их решено было сделать обязательными, а в конце еще и ставить дополнительную оценку в дневник.

Аня тщательно следит, чтобы в ее дневнике появлялись лишь оценки, строго большие четырех, поэтому, выкроив время между посещением бассейна и музыкальной школой, она села делать домашнее задание по спецкурсу «История магии», куда они с подружками ходят по средам, сразу после математики.

На последнем занятии рассказывалось, что все заклинания представляют собой конечные последовательности строчных латинских букв (возможно, пустые). Более того, история магии такая длинная и насыщенная, а маги в стародавние времена трудились столь плодотворно, что в наше время можно считать, что каждая конечная последовательность строчных латинских букв (в том числе и пустая) обязательно является заклинанием. После этого преподаватель рассказал, что в древности магические ордена с одной стороны беспокоились о передаче и сохранении своих знаний, а с другой ни в коем случае не желали делиться своими результатами с кем бы то ни было, вследствие чего в ордене Данных Строк применялся следующий метод шифрования. Чтобы сохранить три заклинания  $A$ ,  $B$  и  $C$ , из них формировали три строки:  $A' = *A*B*$ ,  $B' = *C*A*$ ,  $C' = *B*C*$ , где вместо каждого символа  $*$  подставлялась любая последовательность строчных латинских букв (возможно, пустая). Дополнительно требовалось, чтобы длины всех трех полученных строк были равны некоторому заранее выбранному числу  $N$ . Например, строки  $A = \text{«a»}$ ,  $B = \text{«tb»}$  и  $C = \text{«c»}$  могли быть зашифрованы при помощи строк  $A' = \text{«agtbt»}$ ,  $B' = \text{«icea»}$  и  $C' = \text{«tbhc»}$ .

В качестве домашнего задания преподаватель предложил каждому школьнику попытаться расшифровать по одному результату такой записи. Ане даны три строки  $A'$ ,  $B'$ ,  $C'$ , вероятно полученные из каких-то заклинаний  $A$ ,  $B$  и  $C$  описанными выше преобразованиями.

Поскольку некоторые наборы из трех строк могли быть получены и больше чем из одного варианта исходных заклинаний  $A$ ,  $B$  и  $C$ , Ане необходимо всего лишь найти максимально возможную суммарную длину исходных заклинаний.

### Формат входных данных

Первая строка входного файла содержит единственное целое число  $N$  ( $1 \leq N \leq 2000$ ) — длину каждой из входных строк. В следующих трех строках содержатся строки  $A'$ ,  $B'$ ,  $C'$ , каждая из которых состоит из  $N$  строчных латинских букв.

### Формат выходных данных

Выведите единственное целое число — ответ на поставленную задачу.

## Примеры

стандартный ввод	стандартный вывод
2 ac ba cb	3
4 agtb icea tbhc	4
3 abc cde dea	2

## Задача В. Ахо

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

Это интерактивная задача.

Каждый раз, когда Маргарет и Альфред посещают кафе «У Дональда», они играют в странную игру про угадывание строк.

Её правила таковы:

- Первый игрок пишет строку  $S$  фиксированной длины  $N$ . Также у первого игрока есть строка  $T$ , изначально пустая. Обе строки состоят только из маленьких букв английского алфавита.
- Второй игрок не знает эти строки в течение всей игры. Однако ему разрешено спрашивать про любые две позиции (в обеих строках), правда ли, что символы в них равны. Например, вопрос может выглядеть так “Равны ли второй символ строки  $S$  и пятый символ строки  $T$ ?” Обратите внимание, что можно спрашивать про два символа одной строки.
- Игра состоит из  $M$  раундов. В начале каждого раунда первый игрок добавляет один символ в конец строки  $T$ .
- После добавления символа второй игрок может задать не более пяти вопросов. После этого он должен сказать, какое число подстрок строки  $T$  равно строки  $S$ .

Маргарет быстро заметила, что Альфред всегда преуспевает в роли второго игрока. Она подозревает наличие стратегии, позволяющей второму игроку выигрывать независимо от  $S$  и  $T$ . А вы так сможете?

### Формат входных данных

При запуске ваша программа должна считывать два целых числа  $N$  и  $M$  ( $1 \leq N, M \leq 20\,000$ ) из стандартного потока ввода.

Далее следуют  $M$  раундов игры. В  $i$ -м раунде вы можете задать **не более пяти** вопросов в формате “<позиция1> <позиция2>”. Описание любой позиции выглядит как “s  $x$ ” где  $1 \leq x \leq N$  (если это  $x$ -й символ строки  $S$ ) или как “t  $y$ ” где  $1 \leq y \leq i$  (если это  $y$ -й символ строки  $T$ ). Ответ программы жюри будет “Yes”, если символы на этих позициях равны и “No” иначе.

### Формат выходных данных

В конце любого раунда вы должны вывести ответ в формате “\$  $k$ ”, где  $k$  равно числу вхождений строки  $S$  в строку  $T$ . После этого в строку  $T$  будет автоматически добавлен новый символ (если это не последний раунд).

Не забудьте делать `flush` после каждого вопроса. После того, как вы вывели все  $m$  чисел, ваша программа должна автоматически завершиться, иначе ваш вердикт может быть каким угодно.

## Пример

стандартный ввод	стандартный вывод
3 7	s 1 s 2
No	s 1 s 3
Yes	s 2 t 1
No	s 1 t 1
Yes	\$ 0
Yes	s 2 t 2
Yes	\$ 0
No	s 3 t 3
No	\$ 1
Yes	s 2 t 4
Yes	s 1 t 4
Yes	\$ 1
	s 1 t 5
	\$ 1
	s 2 t 6
	\$ 1
	s 3 t 7
	\$ 2

## Замечание

В примере, строка  $S$  изначально равна "aba", а строка  $T$  получается добавлением символов "a", "b", "a", "c", "a", "b", "a".

## Задача С. Палиндромы

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Вам дана строка  $s$  из маленьких английских букв. Определим для подстроки *красоту* как количество вхождений подстроки в строку, умноженное на длину подстроки. Для данной строки найдите максимальную красоту среди всех её палиндромных подстрок.

### Формат входных данных

Первая строка содержит строку  $s$ . Длина строки не превосходит  $3 \cdot 10^5$ .

### Формат выходных данных

Выведите одно число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
abacaba	7
www	4

## Задача D. Мультимножество Василия

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

У автора уже закончились истории про Василия, поэтому он просто написал формальную постановку задачи.

У вас есть  $q$  запросов и мультимножество  $A$ , изначально содержащее только число 0. Запросы бывают трёх видов:

- «+  $x$ » — добавить в мультимножество  $A$  число  $x$ .
- «-  $x$ » — удалить одно вхождение числа  $x$  из мультимножества  $A$ . Гарантируется, что хотя бы одно число  $x$  в этот момент присутствует в мультимножестве.
- «?  $x$ » — вам даётся число  $x$ , требуется вычислить  $\max_{y \in A} x \oplus y$ , то есть максимальное значение побитового исключающего ИЛИ (также известно как XOR) числа  $x$  и какого-нибудь числа  $y$  из мультимножества  $A$ .

Мультимножество — это множество, в котором разрешается несколько одинаковых элементов.

### Формат входных данных

В первой строке входных данных содержится число  $q$  ( $1 \leq q \leq 200\,000$ ) — количество запросов, которые требуется обработать Василию.

Каждая из последующих  $q$  строк входных данных содержит один трёх символов «+», «-» или «?» и число  $x_i$  ( $1 \leq x_i \leq 10^9$ ). Гарантируется, что во входных данных встречается хотя бы один запрос «?».

Обратите внимание, что число 0 всегда будет присутствовать в мультимножестве.

### Формат выходных данных

На каждый запрос типа «?» выведите единственное целое число — максимальное значение побитового исключающего ИЛИ для числа  $x_i$  и какого-либо числа из мультимножества  $A$ .

### Пример

стандартный ввод	стандартный вывод
10	11
+ 8	10
+ 9	14
+ 11	13
+ 6	
+ 1	
? 3	
- 8	
? 3	
? 8	
? 11	

### Замечание

После первых пяти операций в мультимножестве  $A$  содержатся числа 0, 8, 9, 11, 6 и 1.

Ответом на шестой запрос будет число  $11 = 3 \oplus 8$  максимальное из чисел  $3 \oplus 0 = 3$ ,  $3 \oplus 9 = 10$ ,  $3 \oplus 11 = 8$ ,  $3 \oplus 6 = 5$  и  $3 \oplus 1 = 2$ .

## Задача E. Divljak

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	768 мегабайт

Это история про  $N$  варваров и Тарзана. Варвары занумерованы числами от 1 до  $N$ , и у каждого на табличке написана строка из строчных букв латинского алфавита. Игра состоит из  $Q$  раундов, каждый из которых может быть одного из двух типов:

1. Тарзан показывает слово  $P$  варварам.
2. Тарзан спрашивает варвара номер  $i$ , сколько слов из показанных им до текущего момента таковы, что слово на табличке варвара является их подстрокой.

Варвары не очень любят задачи на строчки, поэтому помогите им ответить на все запросы Тарзана.

### Формат входных данных

Первая строка содержит число варваров  $N$  ( $1 \leq N \leq 10^5$ ).

Каждая из следующих  $N$  строк содержит слово на табличке очередного варвара.

Далее идет число запросов  $Q$  ( $1 \leq Q \leq 10^5$ ).

Каждая из следующих  $Q$  строк описывает очередной запрос. Сперва записано число  $t$  — тип запроса. Если  $t$  равно 1, то далее записано слово  $P$ , которое показывает Тарзан. Иначе  $t$  равно 2 и далее записан номер варвара  $i$  ( $1 \leq i \leq N$ ), которого спрашивает Тарзан.

Суммарная длина слов на табличках варваров  $\leq 2 \cdot 10^6$ .

Суммарная длина показанных Тарзаном слов  $\leq 2 \cdot 10^6$ .

### Примеры

стандартный ввод	стандартный вывод
3 a bc abc 3 1 abca 2 1 2 3	1 1
7 abba bbaa b bbaa abba a ba 7 1 aaabbabbaab 2 7 1 baabaaa 1 aabbbab 2 3 1 aabba 2 3	1 3 4

## Задача F. Подпалиндромы

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Строка называется палиндромом, если она читается одинаково как слева направо, так и справа налево. Например, строки "abba", "kek" являются палиндромами.

Дана строчка. Ее подстрокой называется некоторая непустая последовательность подряд идущих символов. Напишите программу, которая определит, сколько подстрок данной строки является палиндромами.

### Формат входных данных

Вводится одна строка, состоящая из маленьких латинских букв. Длина строки не превышает 100 000 символов.

### Формат выходных данных

Выведите одно число – количество подстрок данной строки, являющихся палиндромами.

### Примеры

стандартный ввод	стандартный вывод
aaa	6
aba	4



## Задача G. Логотип

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В целях своей рекламной кампании большая компания в городе Гдыня хочет разместить в городе свой логотип. Компания хочет потратить весь свой годовой рекламный бюджет на логотип, поэтому он должен быть действительно большим. Один из управляющих решил использовать целые здания как части логотипа.

Логотип состоит из  $n$  вертикальных полос разной длины. Полосы пронумерованы числами от 1 до  $n$  слева направо. Логотип описан перестановкой  $(s_1, s_2, \dots, s_n)$  чисел  $1, 2, \dots, n$ . Полоса под номером  $s_1$  самая короткая, полоса под номером  $s_2$  самая короткая среди оставшихся, и так далее, полоса  $s_n$  самая длинная. Нам не будет важно, какая именно длина каждой полосы.

На главной улице Гдыни  $m$  зданий. Удивительно, но высоты всех  $m$  зданий различны. Встала проблема, найти все позиции где логотип соответствует зданиям.

Помогите компании и найдите все непрерывные отрезки последовательности домов, которые соответствуют логотипу. Непрерывный отрезок домов соответствует логотипу, если здание под номером  $s_1$  в этом отрезке самое низкое, здание под номером  $s_2$  самое низкое среди оставшихся на отрезке, и так далее. Например, последовательность зданий с высотами 5, 10, 4 соответствует логотипу, описанному перестановкой  $(3, 1, 2)$ , так как здание номер 3 (высоты 4) самое низкое, здание 1 (высоты 5), самое низкое среди оставшихся двух и здание номер 2 самое высокое.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $2 \leq n \leq m \leq 1\,000\,000$ ).

Вторая строка содержит  $n$  целых чисел  $s_i$ , которые описывают перестановку из чисел  $1, 2, \dots, n$ , это значит, что  $1 \leq s_i \leq n$  и  $s_i \neq s_j$  при  $i \neq j$ .

Третья строка содержит  $m$  целых чисел  $h_i$  — высоты зданий ( $1 \leq h_i \leq 10^9$ ). Все  $h_i$  различны.

### Формат выходных данных

В первую строку выведите целое число  $k$  — число соответствий. Во вторую строку выведите  $k$  целых чисел — номера домов, которые соответствуют полоске номер 1 логотипа в каждом из  $k$  соответствий. Числа во второй строке требуется выводить в возрастающем порядке. Если  $k = 0$ , выведите второй строкой пустую.

### Пример

стандартный ввод	стандартный вывод
5 10	2
2 1 5 3 4	2 6
5 6 3 8 12 7 1 10 11 9	

### Замечание

Последовательности 6, 3, 8, 12, 7 и 7, 1, 10, 11, 9 соответствуют логотипу, описанному перестановкой  $(2, 1, 5, 3, 4)$ . В частности, в первой последовательности здание номер 2 (высоты 3) самое низкое, здание номер 1 (высоты 6) самое низкое среди оставшихся, здание номер 5 (высоты 7) следующее по высоте, и так далее.

## Задача Н. Словарь

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 1024 мегабайта

*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.

### Пример

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

## Задача I. Под-бор

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 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 входного файла 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.

### Пример

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 J. Периодические префиксы

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

Дана строка  $s$ , найдите для каждого ее префикса максимальное  $k_i$ , такое, что префикс длины  $i$  является конкатенацией  $k_i$  одинаковых строк.

### Формат входных данных

Первая строка входного файла содержит строку  $s$  ( $1 \leq |s| \leq 400\,000$ ). Строка состоит из строчных латинских букв.

### Формат выходных данных

Выведите  $|s|$  чисел: для всех длин от 1 до  $|s|$  выведите максимальное  $k_i$ , такое, что префикс длины  $i$  является конкатенацией  $k_i$  одинаковых строк.

### Пример

стандартный ввод	стандартный вывод
aabaabb	1 2 1 1 1 2 1

## Задача К. Рефрен

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Рассмотрим последовательность  $n$  целых чисел от 1 до  $m$ . Подпоследовательность подряд идущих чисел называется рефреном, если произведение ее длины на количество вхождений в последовательность максимально.

По заданной последовательности требуется найти ее рефрен.

### Формат входных данных

Первая строка входного файла содержит два целых числа:  $n$  и  $m$  ( $1 \leq n \leq 150\,000$ ,  $1 \leq m \leq 10$ ).

Вторая строка содержит  $n$  целых чисел от 1 до  $m$ .

### Формат выходных данных

Первая строка выходного файла должна содержать произведение длины рефрена на количество ее вхождений. Вторая строка должна содержать длину рефрена. Третья строка должна содержать последовательность которая является рефреном.

### Пример

стандартный ввод	стандартный вывод
9 3	9
1 2 1 2 1 3 1 2 1	9
	1 2 1 2 1 3 1 2 1

## Задача L. Задачка на строчечки

Имя входного файла: inputik.txt  
Имя выходного файла: outputik.txt  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайтов

В стандартном поточике вводика или файлик  
е  
кой. На следующей строчечке программо  
ь а і  
н к; эти словечки мы назовём словарик п  
е о о н р  
н в utputik.txt N строчечек. В i-ой ма u  
д о о . й t  
о к - количюсик (сколько штучеч с д і  
х у к е т В ё k  
с б и о ем через пробельчик для к р а т .  
и л к т ) о ш t  
х й л а в всех вхожденъечек к ч а ч х  
м и а е з о а в е и t  
ё к ф с к ртированном про э ж х ч п с  
в с и , и о я т д о к р е в  
о н в ч й л с рочечек нач д о о же о л а  
з и о а т т и о й г д г к ш  
а т и е к ч о с ничек. н ч о е п р о а  
на л о ь а и а к с н р a  
л и в н н в х де с ястю е т в ь о м N п  
ы р е е . р х е г м , р  
м х а е н и ю св и к и с к е д н и о о ч р о o  
и к п д к у ч ж e a ч a г  
ю к и о и к ь н e н д o x c и в и к ч e д к м к р  
у ь д : х с e м a в a  
р н o к с к e d н и e т и д e в ы в a к ч и ь н с o м  
o e в o и т ч д c m  
т л ы л в a к и p a в o л c з и i и k ч e ч o p k o л o  
o a в e a л e ч  
к м с и ч o к ь л o x c e n и t c e в ы в a н ж л o d ж д к  
к н у a  
, e e ч и ч o т o п й н т p a d н a t c a n и t c e в ы в a ю  
к ж щ н  
o х e t з и y k ч e в o l c o p - х a k ч e ч o p t c N х и a  
в й  
o k y б х и k c н и т a л х и k ь н e л a m з и y k ч e ч o p t c т e d

## Формат входных данных

В стандартном поточике вводика или файлике `inputik.txt` ваша программочка найдёт строчечку из маленьких латинских буковок, которую мы назовём исходненькой. На следующей строчечке программочка найдёт числище  $N$  ( $1 \leq N \leq 1\,000\,000$ ), а в следующих  $N$  строчечках — по словечку из тех же маленьких латинских буковок; эти словечки мы назовём словариком. Суммарненькая суммочка длинниц словечек из словарика не превосходит  $1\,000\,000$ .

## Формат выходных данных

Ваша программочка должна вывести на стандартный поточичек выводика или в файл `outputik.txt`  $N$  строчечек. В  $i$ -ой строчечке программочка должна вывести несколько чиселок: первое чиселко — количюсик (сколько штучечек) вхожденьчек строчечки  $i$  из словарика в исходненькой, затем через пробельчик для каждого вхожденьчика выведите индексики началиков всех вхожденьчик этой строчечки в исходненькую в отсортированном порядочке. Индексики всех строчечек начинаются с единичек. Няшечки-преподавашечки гарантируют, что колчюсик вхожденьчик не превосходит  $1\,000\,000$ .

## Пример

inputik.txt	outputik.txt
abrachkacadabrachka	2 1 12
4	1 9
abrachka	2 1 12
cadabrachka	0
ab	
marazmik	

## Задача М. Ключ к шифру (3 балла)

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Сейчас Эркюль Пуаро занят разоблачением международного преступного синдиката, занимающегося контрабандой предметов искусства. Полиция, сотрудничающая с Пуаро, перехватила зашифрованное письмо, содержащее информацию о месте и времени предстоящей сделки, на которой будет присутствовать и глава синдиката. Чтобы сорвать сделку и задержать главу синдиката, необходимо расшифровать перехваченное письмо.

Эркюль знает, что ключ к шифру вычисляется из строки  $s$ . Обозначим за  $f(w)$  длину максимального суффикса  $w$ , не равного  $w$ , который является и префиксом  $w$ . Например,  $f(abc) = 0$ ,  $f(abab) = 2$ ,  $f(aaa) = 2$ . Тогда ключом является максимум по всем  $t$ , являющимся подстроками  $s$ , величины  $(|t| + f(t)^2)$ . Помогите Эркюлю вычислить ключ.

### Формат входных данных

В единственной строке дана строка  $s$ , состоящая из строчных латинских букв ( $1 \leq |s| \leq 500\,000$ ).

### Формат выходных данных

Выведите единственное целое число — искомый ключ к шифру.

### Пример

стандартный ввод	стандартный вывод
ababaab	14



## Задача N. Удаление

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 128 мегабайт

Дано  $n$  различных строк  $s_i$ . Требуется найти количество таких пар  $(i, j)$ , что  $i \neq j$ , а так же  $s_j$  можно получить из  $s_i$  каждый раз удаляя только первый или второй символ из  $s_i$ .

Например из строки `abcxux` можно получить строку `сux`: `abcxux`  $\rightarrow$  `асxux`  $\rightarrow$  `сxux`  $\rightarrow$  `сux`.

### Формат входных данных

Первая строка содержит число  $n$  ( $2 \leq n \leq 200000$ ) — количество строк.

$i$ -я из следующих  $n$  строк содержит строку  $s_i$  ( $1 \leq |S|$ ). Гарантируется, что  $s_i$  состоит из строчных букв латинского алфавита.

Так же гарантируется, что  $|s_1| + \dots + |s_n| \leq 10^6$ .

### Примеры

стандартный ввод	стандартный вывод
3 abcxux сux abc	1
6 b a abc c d ab	5

## Задача О. Цензура

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Посчитайте, сколько строк над алфавитом из  $n$  символов длины  $m$  не содержат ни одной подстроки из заданного множества “запрещенных” строк.

### Формат входных данных

В первой строке написаны целые числа  $n$  ( $1 \leq n \leq 100$ ) — количество символов в алфавите,  $m$  ( $1 \leq m \leq 100$ ) — длина искомых строк и  $p$  ( $0 \leq p \leq 10$ ) — количество “запрещенных” подстрок. Следующая строка содержит  $n$  символов с кодами больше 32 — буквы алфавита. Далее идет  $p$  “запрещенных” строк, длины которых не превосходят  $\min(m, 10)$  символов. Строки целиком состоят из символов алфавита.

### Формат выходных данных

В первой строке выведите ответ на задачу по модулю 1000000007.

### Пример

стандартный ввод	стандартный вывод
2 3 1 ab bb	5

## Задача Р. Подстроки

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дано  $K$  строк из маленьких латинских букв. Требуется найти их наибольшую общую подстроку.

### Формат входных данных

В первой строке число  $K$  ( $1 \leq K \leq 10$ ).

В следующих  $K$  строках — собственно  $K$  строк (длины строк от 1 до 10 000).

### Формат выходных данных

Выведите наибольшую общую подстроку.

### Пример

стандартный ввод	стандартный вывод
3 abacaba mucabarchive acabistrue	cab

## Задача Q. Суффиксный массив (2 балла)

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 мегабайт

Постройте суффиксный массив для заданной строки  $s$ .

### Формат входных данных

Первая строка входного файла содержит строку  $s$  ( $1 \leq |s| \leq 5\,000\,000$ ). Строка состоит из строчных латинских букв.

### Формат выходных данных

Выведите  $|s|$  различных чисел — номера первых символов суффиксов строки  $s$  так, чтобы соответствующие суффиксы были упорядочены в лексикографически возрастающем порядке.

### Пример

стандартный ввод	стандартный вывод
ababb	1 3 5 2 4

## Задача R. ALCS

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 1024 мегабайта

Даны две строки:  $A$  длины  $n$  и  $B$  длины  $m$ . Для каждой подстроки  $A$  найдите длину её LCS со строкой  $B$ .

### Формат входных данных

В первой строке содержится целое число  $n$  ( $1 \leq n \leq 5000$ ). Во второй строке задана строка  $A$  из маленьких букв латинского алфавита длины  $n$ .

В третьей строке содержится целое число  $m$  ( $1 \leq m \leq 5000$ ). В четвёртой строке задана строка  $B$  из маленьких букв латинского алфавита длины  $m$ .

### Формат выходных данных

Выведите  $n$  строк. В  $l$ -й строке должны быть записаны  $n - l + 1$  целых чисел  $ans_{lr}$ , где  $ans_{lr}$  — ответ на задачу для подстроки  $[l, r]$ .

### Примеры

стандартный ввод	стандартный вывод
7 abacaba	1 1 2 2 3 3 3 1 2 2 3 3 3
4 baca	1 1 2 2 0 1 1 2 1 1 2 1 2 1
6 amogus	0 0 1 1 1 1 0 1 1 1 1
9 qwertyuio	1 1 1 1 0 1 1 1 1 0

## Задача S. Задачка на подстроочки

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 4 секунды  
Ограничение по памяти: 1024 мегабайта

Филипп очень любит задачки на строчки. Он уже решил все известные ему задачки, но этого ему было мало. Поэтому Филипп решил придумать свою собственную задачку.

Для этого он взял строку  $t$  и набор из  $n$  строк  $s_1, s_2, s_3, \dots, s_n$ . У Филиппа есть  $m$  запросов, в  $i$ -м из них Филипп хочет взять подстроку строки  $t$  с  $l_i$ -го по  $r_i$ -й символ, и посчитать число её подстрок, которые совпадают с какой-то строкой из набора. Более формально, Филипп хочет посчитать число пар позиций  $a, b$ , таких что  $l_i \leq a \leq b \leq r_i$ , и подстрока строки  $t$  с  $a$ -го по  $b$ -й символ совпадает с некоторой строкой  $s_j$  из набора.

Подстрокой строки  $t$  с  $a$ -го по  $b$ -й символ называется строка, полученная из  $t$  путём удаления  $a - 1$  символа из начала и  $|t| - b$  символами из конца, где  $|t|$  обозначает длину строки  $t$ .

Филипп уже решил эту задачу, а сможете ли вы?

### Формат входных данных

В первой строке два целых положительных числа  $n$  и  $m$  ( $1 \leq n, m \leq 500\,000$ ) — число строк в наборе и количество запросов.

Во второй строке дана единственная строка  $t$ , состоящая из строчных букв английского алфавита ( $1 \leq |t| \leq 5 \cdot 10^6$ ).

В следующих  $n$  строках описываются строки из набора. В  $i$ -й из них дана единственная строка  $s_i$ , состоящая из строчных букв английского алфавита. Обозначим за  $S$  суммарную длину всех строк из набора. Гарантируется, что  $S \leq 10^6$ , а так же что все строки  $s_i$  различны.

В следующих строках вводятся запросы. В  $i$ -й из них даны два целых положительных числа  $l_i$  и  $r_i$  ( $1 \leq l_i \leq r_i \leq |t|$ ) — левая и правая граница подстроки  $t$  из  $i$ -го запроса.

### Формат выходных данных

В единственной строке выведите  $m$  целых чисел,  $i$ -е из них должно быть равно ответу на  $i$ -й запрос.

### Примеры

стандартный ввод	стандартный вывод
3 5 abacaba aba a ac 1 7 1 3 2 7 2 5 4 5	7 3 5 3 1
4 4 abcdca ab ca bcd openolympiad 1 5 2 2 2 6 1 6	2 0 2 3

## Замечание

В первом примере в первом запросе требуется у всей строки посчитать число подстрок, которые входят в набор. Со строкой «aba» совпадают подстроки [1, 3] и [4, 6]. Со строкой «a» совпадают подстроки [1, 1], [3, 3], [5, 5], [7, 7]. Со строкой «ac» совпадает подстрока [3, 4]. Всего получается, что 7 подстрок строки «abasaba» совпадают со строками из набора.

Во втором запросе от исходной строки берется подстрока с 1 по 3 позицию, это строка «aba». В неё строка «aba» входит 1 раз, строка «a» входит 2 раза и строка «ac» не входит ни одного раза как подстрока.

В третьем запросе от исходной строки берется подстрока с 2 по 7 позицию, это строка «basaba». В неё строка «aba» входит 1 раз, строка «a» входит 3 раза и строка «ac» входит 1 раз как подстрока.

## Система оценки

Тесты к этой задаче состоят из 9 групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов некоторых из предыдущих групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Доп. ограничения				Необх. группы	Комментарий
		$n$	$m$	$ t $	$S$		
0	0	–	–	–	–	–	Тесты из условия.
1	10	$n \leq 100$	$m \leq 100$	$ t  \leq 100$	$S \leq 10\,000$	0	
2	12	$n \leq 100$	$m \leq 500$	$ t  \leq 5000$	–	0, 1	
3	7	$n \leq 5000$	–	$ t  \leq 5000$	–	0, 1, 2	
4	8	$n \leq 100$	–	$ t  \leq 50\,000$	–	0, 1, 2	
5	12	–	–	$ t  \leq 100\,000$	$S \leq 100\,000$	0, 1	
6	8	–	–	$ t  \leq 250\,000$	$S \leq 100\,000$	0, 1, 5	
7	7	–	–	$ t  \leq 500\,000$	$S \leq 100\,000$	0, 1, 5, 6	
8	7	–	–	$ t  \leq 750\,000$	$S \leq 100\,000$	0, 1, 5, 6, 7	
9	29	–	–	–	–	0 – 8	<b>Offline-проверка.</b>