Задача 1.

Лёгкая

**Равные подстроки**

**Ограничение по времени работы: 2 секунды**

Дана строка S=s1s2…sn (то есть символы строки индексируются начиная с 1) и множество запросов вида (l1,r1,l2,r2). Для каждого такого запроса  
нужно ответить, равны ли подстроки sl1…sr1 и sl2…sr2 (т.е. от l1 до r1 включительно и от l2 до r2 включительно).

ВХОДНЫЕ ДАННЫЕ

В первой строке записана строка S, состоящая из строчных латинских букв. Эта строка непустая и имеет длину не более 100000 символов. Во второй строке записано целое число q (1⩽q⩽100000) — количество запросов. В каждой из следующих q строк записаны числа l1,r1,l2,r2 (1⩽l1⩽r1⩽|S|; 1⩽l2⩽r2⩽|S|).

ВЫХОДНЫЕ ДАННЫЕ

Для каждого запроса выведите "+", если соответствующие подстроки равны, и "-" в противном случае.

ПРИМЕР

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| abacaba 4 1 1 7 7 1 3 5 7 3 4 4 5 1 7 1 7 | ++-+ |

Задача 2.

Олимпиадная

**Максимальная общая подстрока**

**Ограничение по времени работы: 1 секунда**

Дана строка A=a1a2…an и строка B=b1b2…bm. Нужно узнать длину их максимальной общей подстроки.

ВХОДНЫЕ ДАННЫЕ

В первых двух строках записаны строки A и B, состоящие из строчных латинских букв. Эти строки непустые и имеют длину не более 30000 символов.

ВЫХОДНЫЕ ДАННЫЕ

В выходной файл выведите длину их максимальной общей подстроки.

ПРИМЕР

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| abacaba acabaca | 5 |
| abc xyz | 0 |

Задача 3.

Средняя

**Циклический сдвиг**

**Ограничение по времени работы: 1 секунда**

Даны две строки S и T. Проверьте, может ли строка T быть получена из строки S циклическим сдвигом и найдите наименьшую величину циклического сдвига вправо, при котором из строки S получается строка T.

Например, если исходная строка была равна "abcde", то при циклическом сдвиге вправо на 1 получается строка "eabcd", при циклическом сдвиге на 2 получается строка "deabc" и т.д.

ВХОДНЫЕ ДАННЫЕ

Программа получает на вход две строки, состоящие из заглавных и строчных латинских букв. Строки имеют одинаковую длину, не превосходящую 100.000.

ВЫХОДНЫЕ ДАННЫЕ

Программа должна вывести одно число — величину минимального циклического сдвига вправо, при котором первая строка переходит во вторую. Если такого сдвига не существует, программа должна вывести число -1.

ПРИМЕР

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| abcde deabc | 2 |
| aaa aaa | 0 |
| XYZ ZYX | -1 |

Задача 4.

Лёгкая

**Z-функция**

Дана непустая строка s, длина которой N не превышает 10^6. Будем считать, что элементы строки нумеруются от 1 до N.

Для каждой позиции i символа в строке определим Z-блок как наибольшую подстроку, которая начинается в этой позиции и совпадает с некоторым началом всей строки s. Значением Z-функции Z(i) будем считать длину этого Z-блока. (Заметим, что для первой позиции строки  Z-блок совпадает со всей строкой, поэтому Z(1)=N. С другой стороны, если s[i]≠s[1], то Z(i)=0).

Z-функция используется в различных алгоритмах обработки строк. В частности, с её помощью можно быстро решать задачу о поиске вхождения одной строки в другую («поиск по образцу»).

Требуется для всех i от 1 до N вычислить Z(i).

## **Входные данные**

Одна строка длины N, 0 < N ≤ 106, состоящая из маленьких латинских букв.

## **Выходные данные**

N чисел, разделенные пробелами: Z(1), Z(2), … Z(N)

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| abracadabra | 11 0 0 1 0 1 0 4 0 0 1 |

Задача 5.

Средняя

**Префикс-функция**

Дана строка s. Найдите сумму значений префикс-функции для всех позиций строки s.

ВХОДНЫЕ ДАННЫЕ

Во входном файле записана единственная строка s (1≤|s|≤150000).

ВЫХОДНЫЕ ДАННЫЕ

В выходной файл выведите одно число -— ответ на задачу.

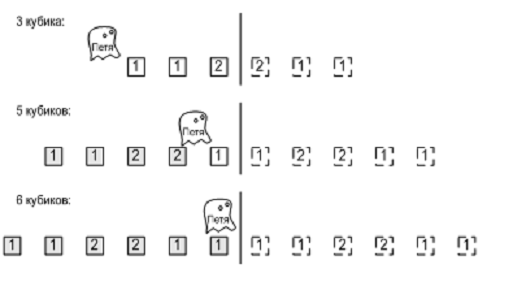
|  |  |
| --- | --- |
| **ввод** | **вывод** |
| abcaabacabc | 11 |

Задача 6.

Средняя

**Кубики**

Привидение Петя любит играть со своими кубиками. Он любит выкладывать их в ряд и разглядывать свое творение. Однако недавно друзья решили подшутить над Петей и поставили в его игровой комнате зеркало. Ведь всем известно, что привидения не отражаются в зеркале! А кубики отражаются. Теперь Петя видит перед собой N цветных кубиков, но не знает, какие из этих кубиков настоящие, а какие — всего лишь отражение в зеркале. Помогите Пете! Выясните, сколько кубиков может быть у Пети. Петя видит отражение всех кубиков в зеркале и часть кубиков, которая находится перед ним. Часть кубиков может быть позади Пети, их он не видит.



## **Входные данные**

Первая строка входного файла содержит число N ( 1 ≤ N ≤ 1000000 ) и количество различных цветов, в которые могут быть раскрашены кубики — M ( 1 ≤ M ≤ 1000000 ). Следующая строка содержит N целых чисел от 1 до M— цвета кубиков.

## **Выходные данные**

Выведите в выходной файл все такие K , что у Пети может быть K кубиков

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 6 2  1 1 2 2 1 1 | 3 5 6 |

Задача 7.

Сложная

**Подпалиндромы**

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

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

Входные данные

Вводится одна строка, состоящая из маленьких латинских букв. Длина строки не превышает 100000 символов.

Выходные данные

Выведите одно число — количество подстрок данной строки, являющихся палиндромами

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| aaa | 6 |
| aba | 4 |

Задача 8.

Сложная

**Type Printer**

Вам нужно напечатать N слов на **Movable Type Printer. Movable Type Printers** — это старые принтеры, для работы которых требуется ставить маленькие металлические кусочки (каждый из кусочков содержит одну букву) в определенном порядке, образуя таким образом слова. Потом все они вдавливаются в лист бумаги. Таким образом печатается одно слово. Ваш принтер позволяет делать следующие операции:

* Добавить букву в конец слова, находящегося сейчас на принтере.
* Удалить последнюю букву из слова, находящегося сейчас на принтере. Эту операцию можно делать, только если слово содержит хотя бы одну букву.
* Напечатать слово, находящееся на принтере (при этом слово никуда не исчезает, можно печатать его ещё раз и ещё раз).

Изначально на принтере содержится пустое слово. В конце печати на принтере можно оставить непустое слово. Слова, которые вам даны, вы можете печатать в произвольном порядке. Каждая из трёх операций занимает одну единицу времени. Вам нужно найти последовательность операций, которая печатает данные N слов за минимальное время. Если минимальных последовательностей несколько, выведите любую.

## **Входные данные**

Ваша программа должна считать следующие входные данные:

* На первой строке число N (1 <= N <= 25000)
* На следующих N  строках слова, состоящие из маленьких букв латинского алфавита. Длина каждого слова не превышает 20. Все слова различны.

## **Выходные данные**

Ваша программа должна вывести следующие данные:

* На первой строке M — число операций.
* На следующих M строках по одному символу — описание операций. Каждая операция описывается одним символом:
  + Добавление символа обозначается собственно символом.
  + Удаление символа обозначается символом «**-**» (минус, ASCII-код 45).
  + Операция «напечатать текущее слово» обозначается символом «**P**» (заглавная латинская буква P).

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 3  print  the  poem | 20  t  h  e  P  -  -  -  p  o  e  m  P  -  -  -  r  i  n  t  P |

Задача 9.

Олимпиадная

**K-ая строка**

Реализуйте структуру данных, которая поддерживает следующие операции:

* добавить в словарь строку S;
* найти в словаре k-ю строку в лексикографическом порядке.

Изначально словарь пуст.

ВХОДНЫЕ ДАННЫЕ

Первая строка входного файла содержит натуральное число N "---  
количество команд (N≤105). Последующие N строк содержат по  
одной команде каждая. Команда записывается либо в виде числа k, либо в виде строки S, которая  
может состоять только из строчных латинских букв.  
Гарантируется, что при запросе k-й строки она существует.  
Также гарантируется, что сумма длин всех добавляемых строк не превышает 105.

ВЫХОДНЫЕ ДАННЫЕ

Для каждого числового запроса k выходной файл должен содержать k-ю  
в~лексикографическом порядке строчку из словаря на момент запроса.  
Гарантируется, что суммарная длина строк в выходном файле не превышает 105.

|  |  |
| --- | --- |
| **ввод** | **вывод** |
| 7 pushkin lermontov tolstoy gogol gorkiy 5 1 | tolstoy gogol |