

Задача А. Префикс-функция

Имя входного файла: `prefix-function.in`
Имя выходного файла: `prefix-function.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

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

Требуется для всех i от 1 до N вычислить её префикс-функцию $\pi[i]$.

Формат входного файла

Одна строка длины N , $0 < N \leq 10^6$, состоящая из маленьких латинских букв.

Формат выходного файла

Выведите N чисел — значения префикс-функции для каждой позиции, разделённые пробелом.

Примеры

<code>prefix-function.in</code>	<code>prefix-function.out</code>
<code>abracadabra</code>	<code>0 0 0 1 0 1 0 1 2 3 4</code>

Задача В. Z-функция

Имя входного файла: `z-function.in`
Имя выходного файла: `z-function.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

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

Требуется для всех i от 1 до N вычислить её z-функцию $z[i]$.

Формат входного файла

Одна строка длины N , $0 < N \leq 10^6$, состоящая из маленьких латинских букв.

Формат выходного файла

Выведите N чисел — значения z-функции для каждой позиции, разделённые пробелом.

Примеры

<code>z-function.in</code>	<code>z-function.out</code>
<code>abracadabra</code>	<code>11 0 0 1 0 1 0 4 0 0 1</code>

Задача С. Поиск набора образцов

Имя входного файла: `console.in`
Имя выходного файла: `console.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Напишите программу, которая для каждой строки из заданного набора S проверяет, верно ли, что она содержит как подстроку одну из строк из набора T .

Формат входного файла

Первая строка входного файла содержит натуральное число n — количество строк в наборе T ($n \leq 1000$). Каждая из следующих n строк содержит непустую строку длины не более 80-ти символов.

Оставшаяся часть файла содержит строки из набора S . Каждая строка состоит из ASCII символов с кодами от 32 до 126 включительно. Строка может быть пустой и ее длина не превышает 250-ти символов.

Гарантируется, что размер входного файла не превышает 1 Мбайт.

Формат выходного файла

В выходной файл выведите все строки из набора S (в том порядке, в котором они находятся во входном файле), содержащие как подстроку по крайней мере одну строку из набора T .

Примеры

<code>console.in</code>	<code>console.out</code>
<code>3</code>	<code>sudislavl</code>
<code>gr</code>	<code>group a'</code>
<code>sud</code>	
<code>abc</code>	
<code>lksh</code>	
<code>sudislavl</code>	
<code>kostroma</code>	
<code>summer</code>	
<code>group a'</code>	

Задача D. Палиндромы

Имя входного файла: `palindrome.in`
Имя выходного файла: `palindrome.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Строка называется палиндромом, если она одинаково читается как слева направо, так и справа налево. Например, `abba` — палиндром, а `омах` — нет. Для строки α будем обозначать $\alpha[i..j]$ ее подстроку длины $j - i + 1$ с i -й по j -ю позицию включительно (позиции нумеруются с единицу). Для заданной строки α длины N ($1 \leq N \leq 100\,000$) требуется подсчитать число q пар (i, j) , $1 \leq i < j \leq n$, таких что $\alpha[i..j]$ является палиндромом.

Формат входного файла

Входной файл содержит одну строку α длины N , состоящую из маленьких латинских букв.

Формат выходного файла

В выходной файл выведите искомое число q .

Примеры

<code>palindrome.in</code>	<code>palindrome.out</code>
<code>aaa</code>	3
<code>abba</code>	2
<code>омах</code>	0

Задача Е. Дуэт на клавиатуре

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

В 2030 году Очень Известная Компания выпустила новую клавиатуру. Разработчики решили избавиться от всех ненужных кнопок и оставить только кнопки с первыми A буквами латинского алфавита. Новая клавиатура пользуется большой популярностью, поэтому Петя решил научиться печатать на ней свое любимое слово (оно не содержит букв, отличных от первых A букв латинского алфавита).

Петя считает, что он научился, когда на экране можно будет увидеть его любимое слово целиком (то есть найдется последовательность подряд идущих букв, образующих его любимое слово). Например, если Петино любимое слово — «apple», и на экране написано «pineapple», то любимое слово увидеть можно, а если на экране написано «apple», то нельзя. Петя запустил текстовый редактор, и пытается, совершив как можно меньше нажатий на клавиши, добиться появления своего любимого слова.

У Пети есть друг Вася, который хочет, чтобы Петя, напротив, совершил как можно больше нажатий на клавиши — так он лучше научится. В любые моменты (как до того, как Петя начал набирать текст, так и между нажатиями Пети на клавиши) Вася может отпихивать Петю от клавиатуры и печатать на ней что угодно. При этом ни Петя, ни Вася не могут стирать уже напечатанные символы. Суммарно Вася может сделать не более K нажатий на клавиши (не обязательно подряд), после этого Петя выгонит его из комнаты, и Вася больше никак не будет участвовать в процессе обучения.

Друзья видят, что написано на экране, и знают, сколько клавиш уже нажал каждый из них. Исходя из этого и Петя, и Вася действуют наиболее оптимально.

Напишите программу, которая определит общее количество Петиних нажатий на клавиши, после которого он гарантированно увидит свое любимое слово.

Формат входного файла

В первой строке входного файла содержатся три целых числа: N , A , K — длина любимого слова Пети, количество кнопок на клавиатуре и максимальное количество нажатий кнопок Васей соответственно ($1 \leq N \leq 100\,000$, $1 \leq A \leq 26$, $1 \leq K \leq 10^9$). В следующей строке содержится слово длины N , состоящее из строчных латинских букв — любимое слово Пети. Слово завершает перевод строки.

Формат выходного файла

Выведите одно число — искомое количество нажатий клавиш.

Примеры

<code>sgame.in</code>	<code>sgame.out</code>
2 1 2 aa	2
3 4 3 abc	9
3 2 1 aab	4

Задача F. Нервных просим условие не читать

Имя входного файла: `grob.in`
Имя выходного файла: `grob.out`
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

От такого нахальства Кормен бы в гробу перевернулся!

Д. Э. Кнут

На одной из станций «Веревочного курса» участникам предлагается следующее испытание: они должны встать в одну линию на узкую скамейку и, не слезая с нее (то есть не касаясь земли), развернуться на ней в обратном порядке.

Несмотря на то, что это очень веселый и интересный конкурс, некоторые люди, считающие его отвратительным, утверждают, что на самом деле его идея не нова и берет корни из ЛРШ (Летней Ритуальной Школы). В ЛРШ–1910 уверовали, что для закрепления знаний алгоритмов на строках, необходимо собираться ночью и танцевать на гробе, в котором лежит мумия Дейкстры.

Ученики ЛРШ–1910 верили, что для достижения нужного эффекта совершенно необходимо было надеть сюртуки, выдаваемые в предыдущих сменах. Так как в 1910 году ЛРШ проводилась в 27-й раз, у всех учеников была возможность надеть один из 26 сюртуков, выданных в прошлые разы. Забавно, но за 26 лет в ЛРШ ни разу не повторились с цветами выдаваемых сюртуков. Поэтому к моменту начала ритуала все лршата были одеты в сюртуки одного из 26 цветов.

Ритуальный танец выглядел следующим образом. Все участники вставали на гроб в один ряд. На земле оставался только ведущий, руководивший танцем. Время от времени он называл два числа l и r — порядковые номера каких-то танцующих. После этого все участники ритуала с порядковыми номерами с l -й по r -й включительно должны были встать в обратном порядке, то есть на позиции l после этого должен был оказаться тот, кто был на r -й позиции, на $(l + 1)$ -й позиции — участник с $(r - 1)$ -й позиции и так далее.

Легенда гласит, что во время ЛРШ-1910 за этим ритуалом наблюдали духи Кнута и Кормена. Время от времени забавы ради они выбирали позиции l и r . Затем они находили самое большое k , такое что для всех i от 0 до $k - 1$ включительно цвета сюртуков лршат на позициях $l + i$ и $r + i$ совпадали. Потом они подлетали к гробу и шептали полученное k на ухо мумии. Узнав о таком нахальстве, мумия Дейкстры в ярости переворачивалась в гробу ровно k раз.

Сейчас на дворе 2010-й год, и противники конкурса на скамейке хотят привести неопровержимые доказательства того, что этот конкурс отвратителен, и Дейкстре бы не понравился. Но для того, чтобы это доказать, им надо сперва проанализировать его переворачивания в ЛРШ-1910.

Помогите им и скажите, сколько раз переворачивалась мумия Дейкстры во время ритуала.

Формат входного файла

В первой строке входного файла содержится информация о начальном расположении участников ритуала на гробе — строка длины n ($1 \leq n \leq 1000\,000$), состоящая из строчных латинских букв: i -й символ описывает цвет сюртука участника на позиции i .

Во второй строке записано единственное число m — количество событий во время ритуала ($0 \leq m \leq 10\,000$).

В следующих m строках содержится информация о произошедших событиях. В каждой из m строк содержится три числа t , l и r , описывающих событие ($t \in \{1, 2\}$, $1 \leq l \leq r \leq n$). Первое число t описывает тип произошедшего события. Если $t = 1$, то в этот момент ведущий называет числа l и r и лршата на позициях с l -й по r -ю встают в обратном порядке. Если $t = 2$, духи Кнута и Кормена выбирают числа l и r и нашептывают мумии соответствующее число k .

Формат выходного файла

Для каждого события типа $t = 2$ выведите на отдельной строке, сколько раз мумия Дейкстры перевернется в результате невинной забавы духов Кнута и Кормена.

Примеры

grob.in	grob.out
abacaba	2
4	1
1 3 6	
1 2 7	
2 1 2	
2 2 5	