

Задача А. Суффиксный массив

Имя входного файла: `array.in`
Имя выходного файла: `array.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Постройте суффиксный массив для заданной строки s , для каждой двух соседних суффиксов найдите длину максимального общего префикса.

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

В первой строке выведите $|s|$ различных чисел — номера первых символов суффиксов строки s так, чтобы соответствующие суффиксы были упорядочены в лексикографически возрастающем порядке. Во второй строке выведите $|s| - 1$ чисел — длины наибольших общих префиксов.

Пример

<code>array.in</code>	<code>array.out</code>
ababb	1 3 5 2 4 2 0 1 1

Задача В. Количество подстрок

Имя входного файла: `count.in`
Имя выходного файла: `count.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Вычислите количество различных подстрок строки s .

Формат входных данных

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

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

<code>count.in</code>	<code>count.out</code>
ababb	11

Задача С. Наибольшая общая подстрока

Имя входного файла: `common.in`
Имя выходного файла: `common.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Найдите наибольшую общую подстроку строк s и t .

Формат входных данных

Первая строка входного файла содержит строку s , вторая — t ($1 \leq |s|, |t| \leq 100,000$). Строки состоят из строчных латинских букв.

Формат выходных данных

Выведите одну строку — наибольшую общую подстроку строк s и t . В случае, если ответ не единственный, выведите минимальный лексикографически.

Пример

<code>common.in</code>	<code>common.out</code>
ababb abacabba	aba

Задача D. Частотность

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

В данный момент астробиологи ведут работу по изучению форм жизни на планете Альфабет. Жизнь там основана на ДНК, составленных из 26 различных нуклеотидов. Таким образом, ДНК каждой формы жизни на планете Альфабет может быть представлена строкой, состоящей из строчных букв английского алфавита. Астробиологи уже получили последовательности ДНК для K форм жизни, суммарная длина этих K последовательностей равняется N . Возможна ситуация, что ДНК некоторых форм жизни совпадают.

Теперь они хотели бы выделить некоторые нити (подстроки) данных ДНК, которые встречаются у различных форм жизни. Обозначим через $L(i)$ (здесь $2 \leq i \leq K$) максимальную длину нити (подстроки), состоящей из последовательных нуклеотидов, которая встречается хотя бы у i форм жизни. Обратите внимание, что $L(i)$ может быть равно 0.

Вычислите значения функции $L(i)$ для всех i от 2 до K .

Формат входных данных

В первой строке записано целое число K , означающее количество форм жизни, для которых была выделена последовательность ДНК. В каждой из последующих K строк записана непустая строка, состоящая из строчных букв английского алфавита. ($2 \leq N \leq 200000$, $2 \leq K \leq N$).

Формат выходных данных

Выходной файл должен содержать $K - 1$ строку со значениями $L(2), L(3), \dots, L(K)$, каждое на отдельной строке.

Система оценки

Тесты оцениваются независимо.

- 30% баллов: $N \leq 10000$
- 40% баллов: $N \leq 100000$
- 30% баллов: $N \leq 200000$

Пример

стандартный ввод	стандартный вывод
6	5
matter	3
animate	2
pattern	2
thermal	1
domain	
teammate	

Замечание

atter встречается в двух строках, mat встречается в трёх строках, ma (или at, или te) встречается в четырёх строках, ma встречается в пяти строках, a встречается во всех шести строках

Задача Е. Периодические префиксы

Имя входного файла: `periodic.in`
Имя выходного файла: `periodic.out`
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

periodic.in	periodic.out
aabaabb	1 2 1 1 1 2 1

Задача F. Подстроки-3

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

Даны 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

Наибольшая общая подстрока.

Пример

стандартный ввод	стандартный вывод
3 abacaba mycabarchive acabistrue	cab

Задача G. Рефрен

Имя входного файла: `refrain.in`
Имя выходного файла: `refrain.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 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

<code>refrain.in</code>	<code>refrain.out</code>
8 3	9
1 2 1 2 1 1 2 1	3
	1 2 1