

Задача А. Суффиксный массив

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

Постройте суффиксный массив для заданной строки s .

Формат входных данных

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

Формат выходных данных

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

Примеры

<code>array.in</code>	<code>array.out</code>
ababb	1 3 5 2 4
abacaba	7 5 1 3 6 2 4

Задача В. Башни

Имя входного файла: `towers.in`
Имя выходного файла: `towers.out`
Ограничение по времени: 10 секунды
Ограничение по памяти: 256 мегабайт

Задано число n и последовательность из n чисел. Требуется рассмотреть все возможные циклические сдвиги заданной последовательности, отсортировать их в лексикографическом порядке, и вывести сумму наибольших общих префиксов соседних в этом порядке сдвигов.

Формат входных данных

Входной файл содержит не более 200 тестовых примеров. Каждый тестовый пример состоит из двух строк. Первая из них содержит целое число $1 \leq n \leq 50000$ — количество магических башен. Вторая строка содержит n чисел в интервале от 0 до 100 — заданную последовательность.

После последнего тестового примера вместо числа n идет 0.

Формат выходных данных

Для каждого тестового примера выведите одно число — искомую сумму.

Примеры

towers.in	towers.out
11 12 8 18 18 8 18 18 8 15 15 8 0	13

Задача С. Циклические сдвиги

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

k -м *циклическим сдвигом* строки S называется строка, полученная перестановкой k первых символов строки S в конец строки.

Рассмотрим все различные циклические сдвиги строки S и отсортируем их по возрастанию. Требуется вычислить i -ю строчку этого массива.

Например, для строки `abacabac` существует четыре различных циклических сдвига: нулевой (`abacabac`), первый (`bacabaca`), второй (`acabacab`) и третий (`cabacaba`). После сортировки по возрастанию получится такой массив: `abacabac`, `acabacab`, `bacabaca`, `cabacaba`.

Формат входных данных

В первой строке входного файла записана строка S , длиной не более 100 000 символов с ASCII-кодами от 32 до 126. Во второй строке содержится единственное целое число k ($1 \leq k \leq 100\,000$).

Формат выходных данных

В выходной файл выведите k -й по возрастанию циклический сдвиг строки S , или слово IMPOSSIBLE, если такого сдвига не существует.

Примеры

<code>shifts.in</code>	<code>shifts.out</code>
<code>abacabac</code> 4	<code>cabacaba</code>

Задача D. Контрольное списывание

Имя входного файла: `kthsubstr.in`
Имя выходного файла: `kthsubstr.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Сегодня на уроке преподаватель Массивов Автомат Укконеви́ч рассказывал своим ученикам про строки, суффиксные структуры и всё такое. Например, он рассказал им, как сравнить две строки A и B лексикографически. Если одна из них является префиксом другой, то более короткая будет лексикографически меньше, иначе необходимо сравнить символы стоящие на первой позиции, в которой они отличаются. Строка с меньшим по номеру в алфавите символом на данной позиции и будет лексикографически меньше.

Чтобы проверить понимание учениками нового материала, Автомат Укконеви́ч дал им следующее задание: найти k -ю лексикографически непустую уникальную подстроку строки S .

Так как учитель знает, что Михаил В. и Роман Б. очень любят списывать у известного в узких кругах Максима И., каждый школьник получил своё число k и вынужден был обратиться к вам за помощью.

Формат входных данных

В первой строке входного файла находится строка S ($|S| \leq 10^5$). Вторая строка содержит число k ($1 \leq k \leq 10^{18}$) — порядковый номер запрашиваемой подстроки.

Формат выходных данных

Если ответ существует, выведите искомую подстроку строки S . В противном случае выведите её лексикографически максимальную подстроку.

Примеры

<code>kthsubstr.in</code>	<code>kthsubstr.out</code>
abacaba 10	acab
abracadabra 10000000000000000000	racadabra

Задача Е. Рефрен

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

Примеры

refrain.in	refrain.out
9 3	9
1 2 1 2 1 3 1 2 1	9
	1 2 1 2 1 3 1 2 1

Задача F. Двоичный суффиксный массив

Имя входного файла: `binary.in`
Имя выходного файла: `binary.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Двоичное слово — это слово, состоящее только из символов 0 и 1.

Рассмотрим двоичное слово w длины n . Суффиксным массивом слова w называется массива $a[1..n]$ такой, $w[a[i]..n]$ — i в лексикографическом порядке суффикс строки w . Например, пусть $w = "001011"$, тогда суффиксы строки w упорядочены следующим образом: "001011", "01011", "011", "1", "1011", "11", а суффиксный массив равен (1, 2, 4, 6, 3, 5). Вам дан суффиксный массив a двоичного слова w . Восстановите w .

Формат входных данных

Первая строка входных данных содержит число n — длина w ($1 \leq n \leq 300\,000$). Вторая строка содержит n различных целых чисел от 1 до n — суффиксный массив двоичного слова w .

Формат выходных данных

Выведите такое двоичное слово w , что его суффиксный массив совпадает с заданным во входных данных массивом. Если таких слов несколько, то разрешается вывести любое из них. Если не существует такого слова w , выведите "Error".

Примеры

binary.in	binary.out
6	001011
1 2 4 6 3 5	

Задача G. Юнг и ковёр

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

Элоди Юнг — талантливая актриса; вы могли видеть её в таких фильмах как “Боги Египта”, “.I. Джо: Бросок кобры ” или “Девушка с татуировкой дракона”. Недавно Элоди купила восхитительный коврик, состоящий из n сшитых подряд кусочков ткани, каждый из которых покрашен в какой-то из 26 цветов. Для простоты будем считать этот коврик диаграммой Ахо длины n (каждому цвету соответствует один из символов латинского алфавита).

Элоди хочет вырезать из этого длинного коврика маленький коврик для своей мамы (мама Элоди — француженка и ценит уютные предметы декора). Для этого она выберет какой-то непрерывный непустой подотрезок коврика, вырежет его — это и будет подарок для её мамы.

Но это еще не все. Из оставшихся частей она хочет сшить точно такой же коврик. Для этого она собирается вырезать из левой и правой оставшихся частей по кусочку ткани и сшить их вместе в таком же порядке — это и будет коврик для неё. При этом Элоди хочет, чтобы эти коврики были **абсолютно** одинаковыми — тем самым она покажет маме свою любовь к ней. Ей стало интересно, сколькими способами можно вырезать коврик.

Формально говоря, она хочет выяснить, сколько существует таких пар индексов i, j , ($1 \leq i \leq j \leq n$), для которых найдутся такие i_1, j_1, i_2, j_2 ($1 \leq i_1 \leq j_1 < i, j < i_2 \leq j_2 \leq n$), что $s[i..j] = s[i_1..j_1] + s[i_2..j_2]$. ($s[l..r]$ — это поддиаграммы диаграммы s , с l -го символа по r -й символ включительно).

Сама Элоди не в состоянии посчитать это число, поэтому она попросил вас помочь ей.

Формат входных данных

В первой и единственной строке входного файла дана диаграмма Ахо s ($1 \leq |s| \leq 100\,000$).

Формат выходных данных

В первой и единственной строке выходного файла выведите одно число — количество способов вырезать коврик.

Примеры

стандартный ввод	стандартный вывод
aaaa	1
abababb	3

Замечание

В первом примере существует единственная подходящая пара индексов (2, 3). Во втором примере существуют три подходящие пары индексов (3, 4), (5, 6) и (4, 6).

Задача Н. Частотность

Имя входного файла: `frequent.in`
Имя выходного файла: `frequent.out`
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. ($2 \leq N \leq 200\,000$, $2 \leq K \leq N$).

Формат выходных данных

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

Примеры

<code>frequent.in</code>	<code>frequent.out</code>
6	5
matter	3
animate	2
pattern	2
thermal	1
domain	
teammate	