

Содержание

Must have	2
Задача 5A. Суффиксный массив [0.2 sec, 256 mb]	2
Задача 5B. LCP для суффиксного массива [0.3 sec, 256 mb]	3
Обязательные задачи	4
Задача 5C. Циклические сдвиги [0.3 sec, 256 mb]	4
Задача 5D. Свобода выбора [1 sec, 256 mb]	5
Задача 5E. Суффиксный массив [1 sec, 256 mb]	6
Дополнительные задачи	7
Задача 5F. Подстроки-3 [0.2 sec, 256 mb]	7
Задача 5G. Рефрен [1.5 sec, 256 mb]	8

У вас не получается читать/выводить данные, открывать файлы?
Воспользуйтесь примерами (c++) (python).

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) **переопределение стандартного аллокатора** ускорит вашу программу.

Обратите внимание на GNU C++ компиляторы с суффиксом inc, они позволяют пользоваться **дополнительной библиотекой**. Под ними можно сдать **вот это**.

Must have

Задача 5А. Суффиксный массив [0.2 sec, 256 mb]

Данна строка, требуется построить суффиксный массив для этой строки. Суффиксный массив — лексикографически отсортированный массив всех суффиксов строки. Каждый суффикс задается целым числом — позицией начала.

Строка s лексикографически меньше строки t , если есть такое i , что $s_i < t_i$ и $s_j = t_j$ для всех $j < i$. Или, если такого i не существует и строка s короче строки t .

Здесь s_i — код i -го символа строки s .

Формат входных данных

Файл состоит из единственной строки. Эта строка — **английский литературный текст**. Длина текста не превосходит 10^5 . Коды всех символов в тексте от 32 до 127.

Формат выходных данных

Выведите N чисел — суффиксный массив данной строки.

Пример

stdin	stdout
99 bottles of beer.	14 3 11 19 2 1 15 4 16 17 9 13 8 12 5 18 10 7 6

Подсказка по решению

Это очень простая задача. Простая именно потому, что текст литературный.

Задача 5B. LCP для суффиксного массива [0.3 сек, 256 mb]

Дана строка длины N и отсортированный массив суффиксов этой строки (т.е. суффиксный массив), вам нужно вычислить LCP. При сортировке строка a считается меньше строки aa . LCP — наибольший общий префикс двух последовательных суффиксов в суффиксном массиве.

Формат входных данных

В первой строке число N ($1 \leq N \leq 10^5$). На второй строке файла дана N строчных латинских букв. В третьей строке N чисел от 1 до N — суффиксный массив (числом i кодируется суффикс, начинающийся с i -го символа).

Формат выходных данных

Выведите $N - 1$ число — значения LCP.

Пример

stdin	stdout
5 сасао 2 4 1 3 5	1 0 2 0

Замечание

Суффиксный массив для строки `сасао`:

асао
ао
сасао
сао
о

Подсказка по решению

24-й тест — антихеш тест.

Можно сдать за $\mathcal{O}(n \log n)$, можно за $\mathcal{O}(n)$.

Обязательные задачи

Задача 5С. Циклические сдвиги [0.3 сек, 256 mb]

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.

Пример

stdin	stdout
abacabac 4	cabacaba
abacabac 5	IMPOSSIBLE

Замечание

Эта задача всё ещё проще чем “построить суффмассив за $\mathcal{O}(n \log n)$ ”.

Кстати, в ней нет антихештестов. Это значит, что хеши по модулю 2^{64} работают также хорошо, как по простому, близкому к 2^{64} ;-)

Задача 5D. Свобода выбора [1 sec, 256 mb]

Даны две строки, состоящих из заглавных латинских букв. Нужно найти их наибольшую общую подстроку. Полное условие можно посмотреть на [тимусе](#).

Формат входных данных

На первой строке число n ($1 \leq n \leq 10^5$).

На второй и третьей строках находятся по n заглавных английских букв.

Формат выходных данных

Максимальную по длине общую подстроку. Если оптимальных ответов несколько, выведите любой.

Примеры

stdin	stdout
28 VOTEFORTHEGREATALBANIAFORYOU CHOOSETHEGREATALBANIANFUTURE	THEGREATALBANIA

Замечание

19-й – антихеш тест

Разбиралась на лекции. Здесь всё ещё не нужны суффмассивы.

Задача 5Е. Суффиксный массив [1 сек, 256 mb]

Постройте суффиксный массив для заданной строки s .

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 такую перестановку начальных позиций суффиксов, чтобы соответствующие суффиксы были упорядочены в лексикографически возрастающем порядке.

Примеры

stdin	stdout
ababb	1 3 5 2 4

Замечание

ababb
abb
b
babb
bb

Подсказка по решению

А вот и он!

Дополнительные задачи

Задача 5F. Подстроки-3 [0.2 sec, 256 mb]

Даны K строк из маленьких латинских букв. Найдите 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.

Примеры

stdin	stdout
3 abacaba myscabarchive acabistrue	cab

Замечание

Нужно написать хороший код, чтобы пройти ТЛ.

В частности вам точно понадобится рукописная хеш-таблица с открытой адресацией.

Задача 5G. Рефрен [1.5 sec, 256 mb]

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

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

stdin	stdout
9 3	9
1 2 1 2 1 3 1 2 1	3
	1 2 1

Замечание

Эту задачу обязательно сдавать суффмассивом.

Даже если больше вам по душе деревья и автоматы.