

Содержание

Задачи	2
Задача 4А. Обмен [0.2 sec, 256 mb]	2
Задача 4В. Словарь [0.7 sec, 256 mb]	3
Задача 4С. Поиск набора образцов [0.1 sec, 256 mb]	4
Бонусные задачи	5
Задача 4D. Подстроки-3 [0.1 sec, 256 mb]	5
Задача 4Е. Подстроки [8 sec, 256 mb]	6
Задача 4F. Общая подпоследовательность [6 sec, 256 mb]	7

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.

Задачи

Задача 4А. Обмен [0.2 sec, 256 mb]

Пусть все натуральные числа исходно организованы в список в естественном порядке. Разрешается выполнить следующую операцию: $swap(a, b)$. Эта операция возвращает в качестве результата расстояние в текущем списке между числами a и b и меняет их местами.

Задана последовательность операций $swap$. Требуется вывести в выходной файл результат всех этих операций.

Формат входных данных

Первая строка входного файла содержит число n ($1 \leq n \leq 200\,000$) — количество операций. Каждая из следующих n строк содержит по два числа в диапазоне от 1 до 10^9 — аргументы операций $swap$.

Формат выходных данных

Для каждой операции во входном файле выведите ее результат.

Пример

stdin	stdout
4	3
1 4	1
1 3	4
4 5	2
1 4	

Задача 4В. Словарь [0.7 sec, 256 mb]

Дан набор слов и текст, требуется определить для каждого слова, присутствует ли оно в тексте как подстрока.

Формат входных данных

В первой строке дан текст (не более 10^6 строчных латинских букв).

Далее дано число M — количество слов в словаре.

В следующих M строках записаны слова (не более 30 строчных латинских букв). Слова различны и отсортированы в лексикографическом порядке.

Суммарная длина слов в словаре не более 10^5 .

Формат выходных данных

M строк вида Yes, если слово присутствует, и No иначе.

Пример

stdin	stdout
trololo	No
3	Yes
abacabadabacaba	Yes
olo	
trol	

Замечание

Эта задача решается просто Бором. Без алгоритма Ахо-Корасик.

Задача 4С. Поиск набора образцов [0.1 sec, 256 mb]

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 входных данных

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 выходных данных

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 .

Пример

stdin	stdout
3	sudislavl
gr	group a'
sud	
abc	
lksh	
sudislavl	
kostroma	
summer	
group a'	

Замечание

Если у вас WA 36, вы неправильно читаете входные данные.
Строки могут состоять только из пробелов.

Бонусные задачи

Задача 4D. Подстроки-3 [0.1 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 mycabarchive acabistrue	cab

Задача 4Е. Подстроки [8 сек, 256 mb]

Дана строка s . Вам требуется подсчитать количество её различных подстрок. Пустую строку учитывать не следует.

Формат входных данных

В единственной строке входного файла содержится данная строка s , состоящая из строчных латинских букв. Длина строки не превосходит 20 000 символов.

Формат выходных данных

В единственной строке выходного файла выведите единственное число — количество различных подстрок s .

Примеры

stdin	stdout
aaaa	4
abacaba	21

Замечание

Можно попробовать сдать хеш-таблицей.

Можно попробовать сдать бором.

Задача 4F. Общая подпоследовательность [6 сек, 256 mb]

У Никиты есть массив A из N элементов. Он хочет найти максимизировать величину

$$A[l_1] \wedge A[l_1 + 1] \wedge \dots \wedge A[r_1]) + (A[l_2] \wedge A[l_2 + 1] \wedge \dots \wedge A[r_2])$$

Где $1 \leq l_1 \leq r_1 < l_2 \leq r_2 \leq N$. Никита – обычный художник, помогите ему.

Формат входных данных

На первой строке число N ($2 \leq N \leq 10^6$). На второй строке N целых чисел от 0 до 10^9 .

Формат выходных данных

Одно число – максимум.

Примеры

stdin	stdout
5 1 2 3 1 2	6