

Содержание

Разделяй и властвуй	2
Задача А. Деление многочленов [1 sec, 256 mb]	2
Суффиксный автомат	3
Задача В. Суффиксный автомат [1 sec, 256 mb]	3
Задача С. Помогите, спасите! [1 sec, 256 mb]	4
Задача D. Подстроки-4 [1 sec, 256 mb]	5
Задача Е. LZSS encoding [2 sec, 256 mb]	6
Паросочетания	7
Задача F. Рабочее расписание [0.25 sec, 256 mb]	7

Вы не умеете читать/выводить данные, открывать файлы? Воспользуйтесь **примерами**.

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 компилятор GNU C++11 5.1.0 (TDM-GCC-64) inc, который позволяет пользоваться **дополнительной библиотекой**. Под ним можно сдать **вот это**.

Разделяй и властвуй

Задача А. Деление многочленов [1 сек, 256 mb]

Даны два многочлена с коэффициентами из $\mathbb{Z}/7\mathbb{Z}$. Старший коэффициент обоих не равен нулю. Нужно поделить их с остатком.

Формат входных данных

Каждая из двух строк задаёт описание многочлена. Многочлен $a_k x^k + \dots + a_2 x^2 + a_1 x + a_0$ описывается числом k ($0 \leq k \leq 50\,000$) и $k + 1$ числами от 0 до 6: $a_k, \dots, a_2, a_1, a_0$.

Формат выходных данных

На первой строке многочлен-частное. На второй строке многочлен-остаток. Выводите многочлены в том же формате. Если многочлен – тождественный ноль, для него $k = 0$.

Примеры

divpoly.in	divpoly.out
3 1 1 1 1 1 1 1	2 1 0 1 0 0
3 1 1 3 1 2 1 1 1	1 1 0 1 2 1
8 2 1 2 1 2 1 2 1 2 4 1 2 3 4 5	4 2 4 2 5 2 3 3 1 3 6

Суффиксный автомат

Задача В. Суффиксный автомат [1 сек, 256 mb]

Дана строка. Постройте её суффиксный автомат.

Формат входных данных

Строка длины от 1 до 100 000, состоящая из маленьких латинских букв.

Формат выходных данных

На первой строке число состояний автомата и число рёбер. Следующие строки содержат рёбра в формате “откуда” “куда” “символ на ребре”. Далее число терминальных состояний и строка, содержащая все терминальные состояния в произвольном порядке. Начальным состоянием автомата должно быть состояние номер один.

Примеры

automaton.in	automaton.out
ababb	7 9 1 2 a 1 7 b 2 3 b 3 4 a 3 6 b 4 5 b 5 6 b 7 4 a 7 6 b 3 6 7 1

Задача С. Помогите, спасите! [1 sec, 256 mb]

Дана строка. Найдите для каждого её префикса количество различных подстрок в нём.

Формат входных данных

В единственной строке входных данных содержится непустая строка S , состоящая из N ($1 \leq N \leq 2 \cdot 10^5$) маленьких букв английского алфавита.

Формат выходных данных

Выведите N строк, в i -й строке должно содержаться количество различных подстрок в i -м префиксе строки S .

Примеры

keepcounted.in	keepcounted.out
aabab	1 2 5 8 11
atari	1 3 5 9 14

Задача D. Подстроки-4 [1 сек, 256 mb]

Даны K строк из маленьких латинских букв. Найдите их наибольшую общую подстроку.

Формат входных данных

В первой строке число K ($1 \leq K \leq 10$). Далее K строк длины от 1 до 200 000.

Формат выходных данных

Наибольшая общая подстрока.

Примеры

substr4.in	substr4.out
3 abacaba mycabarchive acabistrue	cab

Задача E. LZSS encoding [2 sec, 256 mb]

Алиса хочет отправить сообщение Бобу. Она хочет зашифровать сообщение, используя оригинальный метод шифрования. Сообщение – строка S , состоящая из N строчных английских букв.

$S[a \dots b]$ означает подстроку S от $S[a]$ до $S[b]$ ($0 \leq a \leq b < N$). Если первые i букв уже зашифрованы, Алиса найдёт такие (j, k) : $s[j..j+k] = s[i..i+k]$, $k \geq 0$, $0 \leq j < i$, $k = \max$. Если несколько j дают максимальное k , Алиса выберет минимальное j . Если $k > 0$ Алиса добавит пару $\langle j, k \rangle$ в шифр и увеличит i на k , иначе Алиса добавит -1 и ASCII код буквы $S[i]$ в шифр и увеличит i на 1.

Очевидно шифр начнёт с -1, далее будет ASCII код символа $S[0]$. Помогите Алисе реализовать её метод шифрования.

Формат входных данных

Первая строка ввода содержит количество тестов T ($1 \leq T \leq 50$). Следующие T строк содержат сообщения для шифровки, каждое длины от 1 до 10^5 , состоящие из строчных английских букв. Гарантируется, что суммарная длина всех сообщений не превосходит $2 \cdot 10^6$.

Формат выходных данных

Для каждого теста на отдельной строке выведите “Case #X:”, где X – номер теста, нумерация с 1. Далее выведите шифр, в каждой строке по два целых числа через пробел.

Примеры

lzss.in	lzss.out
2 aaaaaa aaaaabbbbbaaabbc	Case #1: -1 97 5 0 Case #2: -1 97 4 0 -1 98 4 5 5 2 -1 99

Паросочетания

Задача F. Рабочее расписание [0.25 сек, 256 mb]

Дан неориентированный граф из N вершин и нескольких рёбер.

Найти максимальное паросочетание.

Здесь можно прочесть оригинальную легенду.

Формат входных данных

На первой строке число вершин N ($1 \leq N \leq 222$).

Далее строки, содержащие рёбра.

Формат выходных данных

На первой строке число вершин, покрытых максимальным паросочетанием.

На следующих строках рёбра паросочетания.

Если есть несколько максимальных паросочетаний, выведите любое.

Примеры

schedule.in	schedule.out
3	2
1 2	1 2
2 3	
1 3	

Подсказка по решению

Алгоритм Эдмондса сжатия соцветий. Реализация Габова.