

Содержание

Задача А. Суффиксное дерево [0.5 sec, 256 mb]	2
Задача В. Рефрен [0.5 sec, 256 mb]	3
Задача С. Свобода выбора [0.5 sec, 256 mb]	4
Задача D. Подстроки-3 [0.5 sec, 256 mb]	5
Задача Е. Набор строк [0.5 sec, 256 mb]	6

Каждую из задач ABCD нужно сдать двумя способами –
и суффиксным деревом (первая копия задачи), и суффиксным автоматом (вторая копия задачи).

Задача А. Суффиксное дерево [0.5 сек, 256 mb]

Дана строка s . Постройте сжатое суффиксное дерево для строки s и выведите его. Найдите такое дерево, которое содержит минимальное количество вершин.

Формат входных данных

В первой строке записана строка s ($1 \leq |s| \leq 10^5$), последний символ строки доллар «\$», остальные символы строки маленькие латинские буквы.

Формат выходных данных

Пронумеруйте вершины дерева от 0 до $n - 1$ в порядке обхода в глубину, обходя поддеревья в порядке лексикографической сортировки исходящих из вершины рёбер. Используйте ASCII-коды символов для определения их порядка.

В первой строке выведите число n – количество вершин дерева. В следующих $n - 1$ строках выведите описание вершин дерева, кроме корня, в порядке увеличения их номеров.

Описание вершины дерева v состоит из трёх целых чисел: p, lf, rf , где p ($0 \leq p \leq n, p \neq v$) – номер родителя текущей вершины. На ребер ведущем из p в v написана подстрока $s[lf..rf)$ ($0 \leq lf < rf \leq |s|$).

Примеры

stdin	stdout
aaa\$	7 0 3 4 0 0 1 2 3 4 2 1 2 4 3 4 4 2 4
b\$	3 0 1 2 0 0 2
ababa\$	10 0 5 6 0 0 1 2 5 6 2 1 3 4 5 6 4 3 6 0 1 3 7 5 6 7 3 6

Задача В. Рефрен [0.5 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

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

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

Задача С. Свобода выбора [0.5 сек, 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

Задача D. ^{10-й — антихелп-тест}Подстроки-3 [0.5 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 Е. Набор строк [0.5 сек, 256 mb]

В Инновационном Отделе НИИ Исследований Данных Строк разработана клавиатура для внутреннего пользования, облегчающая набор строк огромной длины. Кроме обычных клавиш, соответствующих маленьким латинским буквам, на клавиатуре есть еще n функциональных клавиш F_1, \dots, F_n , соответствующих заданным строкам из словаря S_1, \dots, S_n . При нажатии такой клавиши F_i строка S_i загружается во внутреннюю память клавиатуры. В каждый момент времени в памяти может находиться не более одной строки из словаря.

Кроме того, в клавиатуру встроен графический манипулятор «Кыш», с помощью которого легким движением руки можно ввести любую подстроку находящейся в памяти строки.

Вася занимается исследованием эффективности данного нововведения. Для этого ему требуется написать программу, которая будет вычислять минимальное необходимое количество действий (нажатий и использований «Кыш») для ввода данной строки S . В момент начала ввода строки память пуста.

Например, если требуется ввести строку “abacaba”, а в словаре есть строки “baba” и “caca”, то это можно сделать за четыре действия — нажать F_1 , выбрать манипулятором подстроку “aba”, затем нажать ‘с’, и опять выбрать манипулятором подстроку “aba”. Если бы нужно было набрать с таким словарем “bacababa”, то это можно сделать за пять действий: ‘b’, F_2 , “aca”, F_1 , “baba”.

Формат входных данных

В первой строке входного файла задано число n ($1 \leq n \leq 50$). В последующих n строках заданы S_i , составленные из не более чем 500 символов. В последней строке вводится непустая строка S , длина которой не превосходит 100 000. Все символы строк — маленькие латинские буквы.

Формат выходных данных

Выведите минимальное необходимое количество действий.

Пример

stdin	stdout
2 baba caca abacaba	4
2 baba caca bacababa	5