Задача A. Циклический сдвиг

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

Задана строка s, состоящая из строчных букв латинского алфавита. Необходимо выяснить, есть ли среди ее циклических сдвигов два различных.

Формат входного файла

Входной файл содержит непустую строку s длиной не более $2 \cdot 10^5$ символов.

Формат выходного файла

В выходной файл выведите слово `YES`, если у строки s есть два несовпадающих циклических сдвига, и `NO` — иначе.

Примеры

cyclic.in	cyclic.out
abc	YES
aaa	NO

Задача B. Уравнение

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

Необходимо написать программу, решающую уравнения вида

$$\frac{a \cdot x + b}{c \cdot x + d} = v.$$

Числа a, b, c, d, v заданы, а x — неизвестно.

Формат входного файла

Входной файл содержит пять целых чисел: a, b, c, d, v. Все они не превосходят 1000 по абсолютной величине. Числа во входном файле разделены пробелами и/или переводами строк.

Формат выходного файла

Если у указанного уравнения нет решений, выведите в выходной файл слово `NONE`. Если у уравнения ровно решение, то выведите в выходной файл строку вида `X = p/q`, где p — целое число, q — натуральное, p и q взаимно-просты, а дробь $\frac{p}{q}$ является решением уравнения. Если у уравнения более одного решения, выведите в выходной файл слово `MULTIPLE`.

Примеры

equation.in	equation.out
`1 2 3 4 5`	`X = -9/7`
`1 1 1 1 1`	`MULTIPLE`

Задача C. Высота треугольника

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

Задан треугольник. Найдите длину самой короткой из его высот.

Формат входного файла

Первая строка входного файла содержит три целых числа: l_1, l_2, l_3 — длины сторон треугольника. Они положительны и не превосходят 1000. Гарантируется, что треугольник с такими длинами сторон существует и имеет ненулевую площадь.

Формат выходного файла

В выходной файл выведите ответ на задачу с точностью не хуже 10^{-5}.

Примеры

height.in	height.out
1 1 1	0.866025403784
3 4 5	2.4

Задача D. Марсианский язык

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

Недавно ученые лаборатории радиоастрономии научно-исследовательского института формального исследования галактического антропогенеза (НИИ ФИГА) записали радиосигнал, передаваемый с Марса. После расшифровки выяснилось, что этим радиосигналом передавалась последовательность букв латинского алфавита. Этот факт не вызвал у ученых большого удивления, больше их поразило то, что на Марсе есть радиопередатчики.

Для того, чтобы прочитать переданную информацию, необходимо последовательность букв разбить на слова. Известно, что слова в марсианском языке состоят не менее, чем из c и не более, чем из C букв.

Назовем *качеством* слова разность между максимальным и минимальным номерами в алфавите букв, входящих в слово. Например, качество слова `ab` равно $2-1=1$, а слова `abcz` равно $26-1=25$.

Пусть была передана строка $c_1, c_2, \ldots, c_n$. Для того, чтобы разбить ее на слова, поставим между некоторыми соседними буквами пробелы. Тогда последовательности букв, ограниченные пробелами, началом и концом строки будут являться словами. Качеством разбиения строки на слова назовем сумма качеств всех входящих в него слов.

Ваша задача — найти для строки, полученной учеными лаборатории радиоастрономии, разбиение на слова марсианского языка, обладающее максимально возможным качеством.

Формат входного файла

Первая строка входного файла содержит три целых числа: n, c, C ($1 \le c \le C \le n \le 5000$). Вторая строка входного файла содержит n строчных букв латинского алфавита — переданную последовательность.

Формат выходного файла

В первой строке выходного файла выведите качество построенного разбиения. Во второй строке выведите число k — количество слов в разбиении. Далее выведите все слова разбиения — по одному в каждой из последующих k строк. Конкатенация всех выведенных слов в том порядке, в котором они выведены должна быть равна исходной строке. Если возможных вариантов ответа несколько, выведите любой из них.

Если строку нельзя разбить на слова марсианского языка, то выведите в выходной файл строку `NO SOLUTION`.

Примеры

language.in	language.out
7 2 4 abacaba	4 3 ab ac aba
6 4 5 abacab	NO SOLUTION

Задача E. Прямые на плоскости

Имя входного файла:	`lines2d.in`
Имя выходного файла:	`lines2d.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

На плоскости заданы N прямых. Необходимо выбрать максимальное количество прямых так, чтобы среди выбранных прямых не было параллельных (совпадающие прямые в этой задаче будем считать параллельными).

Формат входного файла

Первая строка входного файла содержит размер n ($1 \le n \le 1000$). Последующие n строк содержат каждая описание одной прямой. Каждая прямая описывается тремя целыми числами: A, B, C — коэффициентами в уравнении $Ax+By+C=0$. Для каждой прямой верны следующие условия: $|A|, |B|, |C| \le 10^5$, $A^2 + B^2 \ne 0$.

Формат выходного файла

В выходной файл выведите размер выбранного поднабора попарно непараллельных прямых.

Примеры

lines2d.in	lines2d.out
3 1 1 4 2 2 2 3 1 0	2
4 1 1 3 2 2 3 3 3 3 4 4 4	1

Задача F. Лексикографически минимальное число

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

Найдите число, десятичная запись которого является лексикографически минимальной среди десятичных записей всех целых чисел от L до R.

Формат входного файла

Входной файл содержит два целых числа: L и R ($1 \le L \le R \le 10^9$).

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

minlex.in	minlex.out
4 7	4
4 11	10

Задача G. Палиндром

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

Напомним, что *палиндромом* называется строка, одинаково читающаяся с обеих сторон. Например, строка `abba` является палиндромом, а строка `abc` — нет.

Рассмотрим все натуральные числа, представления которых в десятичной системе счисления являются палиндромами. Необходимо найти n-ое в порядке возрастания такое число.

Формат входного файла

Входной файл содержит целое число n ($1 \leq n \leq 10^9$).

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

palindr.in	palindr.out
1	1
19	101

Задача H. Дороги

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

На недавно составленном генеральном плане развития одного из районов одного из крупных городов страны N-мерики присутствуют n асфальтовых дорог. Кроме этого, на нем присутствуют m дорожных знаков.

Разумеется, перед утверждением главой администрации района план должен быть тщательно проверен. При этом одним из важных условий, которое подлежит проверке, является то, что ни один дорожный знак не находится внутри или на краю какой-нибудь дороги. Действительно, достаточно странно будет выглядеть знак «Уступи дорогу», стоящий посреди проезжей части.

Для простоты будем считать, что каждая дорога изображается на плане полосой, а каждый дорожный знак — точкой.

Пусть задана некоторая прямая l. *Полосой ширины d, симметричной относительно прямой l* называется геометрическая фигура, состоящая из точек, удаленных от прямой l не более, чем на $d/2$.

Ваша задача состоит в том, чтобы написать программу, которая выполнит указанную выше проверку плана.

Формат входного файла

Первая строка входного файла содержит число n ($0 \le n \le 1000$). Каждая из последующих n строк описывает одну дорогу. Описание каждой дороги состоит из пяти чисел: x_1, y_1, x_2, y_2, d. Эти числа обозначают: (x_1, y_1) и (x_2, y_2) — две различные точки прямой, относительно которой дорога симметрична, d — ширина дороги ($1 \le d \le 1000$).

Далее следует строка, содержащая число m ($0 \le m \le 1000$). Последующие m строк содержат каждая по два числа — координаты соответствующего дорожного знака. Все координаты во входном файле целые и не превышают 10^4 по абсолютному значению.

Формат выходного файла

В выходной файл выведите два числа: количество дорог, внутри которых стоит хотя бы один дорожный знак, и количество дорожных знаков, которые стоят хотя бы на одной дороге.

Примеры

roads.in	roads.out
1 0 0 0 1 2 3 0 0 0 10 10 0	1 2