Задача A. Коттеджный поселок

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

Поблизости от столицы Флатландии одна компания решила построить коттеджный поселок. Строительная компания, которая занимается возведением коттеджей, решила раскрасить некоторые коттеджи в розовый цвет, а остальные — в голубой. Но они не могут решить, какой коттедж раскрасить в какой цвет.

Директор компании утверждает, что раскраска *симпатичная*, если есть хотя бы один розовый коттедж, хотя бы один голубой коттедж, и можно провести такую прямую, что все розовые коттеджи окажутся с одной стороны от нее, а все голубые — с другой стороны (при этом на самой прямой коттеджей быть не должно). На это главный дизайнер возразил, что есть несколько способов сделать симпатичную раскраску.

Помогите им определить, сколько существует различных симпатичных раскрасок.

Формат входного файла

Первая строка входного файла содержит число n — количество коттеджей ($1 \le n \le 300$). Следующие n строк содержат координаты коттеджей, каждая строк содержит два целых числа x_i и y_i ($-10^4 \le x_i, y_i \le 10^4$).

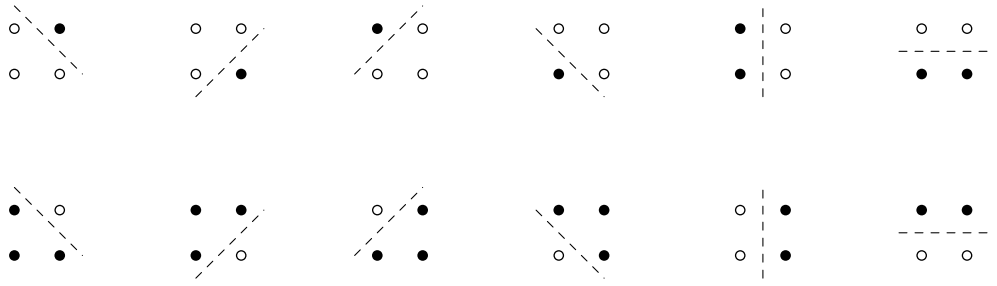
Формат выходного файла

Выведите одно число — количество симпатичных раскрасок коттеджного поселка.

Пример

cottages.in	cottages.out
4 0 0 1 0 1 1 0 1	12

Возможные раскраски приведены на следующем рисунке.



Задача B. Взлом хеш-функции

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

В некоторых задачах защиты информации используются так называемые *хеш-функции*. Одним из важнейших классов хеш-функций являются так называемые *полиномиальные хеш-функции*.

Пусть дана строка $S = s_1 s_2 \dots s_l$, состоящая из цифр от 0 до 9. Тогда значение *полиномиальной хеш-функции* $p(S, x, m)$ вычисляется следующим образом

$$p(S, x, m) = (\sum_{i=1}^{l} s_i x^{i-1}) \bmod m$$

($a \bmod b$ обозначает остаток от деления числа a на число b). Например, пусть $S = 0123$, тогда $p(S, 2, 5) = (0 \cdot 1 + 1 \cdot 2 + 2 \cdot 4 + 3 \cdot 8) \bmod 5 = 4$.

Одним из способов применения хеш-функций является хранение паролей. Часто бывает так, что пароли приходится хранить в незащищенной таблице базы данных, поэтому вместо них хранят хеш-функции от них. При проверке пароля вычисляется хеш-функция от введенной строки и сравнивается со значением, хранящимся в таблице.

Ваша задача состоит в том, чтобы по заданным числам x, m, L и v найти строку S из цифр от 0 до 9 длины L, значение полиномиальной хеш-функции $p(S, x, m)$ равно v.

Формат входного файла

Входной файл содержит четыре целых числа: x (x — простое число, $5 \le x \le 100$), m (m является степенью двойки, $1 \le m \le 256$), L ($10 \le L \le 100$) и v ($0 \le v \le m - 1$).

Формат выходного файла

В выходной файл выведите искомую строку или `NO SOLUTION`, если такой строки не существует.

Примеры

crack.in	crack.out
5 16 10 9	0422207956

Задача C. Обмен валюты

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

Петя работает в обменном пункте во Флатландии. Недавно Петя получил от начальства набор цифр для отображения обменного курса. К сожалению, набор содержит всего по две копии каждой цифры. Теперь Петя хочет узнать, сколько различных обменных курсов он сможет отобразить.

Петя обменивает флатландские доллары на крайландские тугрики. Петя уверен, что курс обмена будет целым числом, которое находится в диапазоне от l до r, включительно.

Формат входного файла

Входной файл содержит два целых числа l и r ($1 \le l \le r \le 10^{18}$).

Формат выходного файла

Выведите одно целое число — количество обменных курсов, которые Петя может отобразить с использованием полученного набора.

Пример

exchange.in	exchange.out
1 1000	990

Задача D. Плейлист

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

Антон — большой любитель музыки. Но в разные моменты ему нравится разная музыка. Однажды он решил составить рейтинг своих песен, причем считать его так — если песня дослушана до конца, то ее рейтинг увеличивается на единицу, если же он так и не дослушал ее, переключив на следующую, то уменьшается на единицу. Отметим, что вследствие этого рейтинг может стать отрицательным.

Он попросил Вас написать ему программу, которая подсчитывала бы такой рейтинг. Для этого он записал в каком порядке прослушивались песни и в какие моменты времени он переключал песню на следующую. Изначально рейтинги всех песен равны нулю.

Формат входного файла

Первая строка входного файла содержит одно число n ($1 \le n \le 1000$) — количество песен у Антона. В следующих n строках следуют описания песен — название песни, состоящее не более, чем из 50 маленьких латинских букв, и длина песни в секундах, разделенные пробелом. Каждая песня длится положительное число секунд и не более 30 минут. $(n+2)$-я строка содержит два числа m, k ($0 \le k < m \le 1000$) — количество прослушанных песен и количество переключений на следующую. В следующих m строках следуют названия прослушанных песен в порядке их прослушивания (все эти песни — из списка). И, наконец. в последней строке — k целых неотрицательных чисел — последовательные (в неубывающем порядке) времена переключения на следующую песню (гарантируется, что она всегда существует), отсчитываемые в секундах от начала прослушивания первой. Считается, что если Антон переключил песню в момент окончания какой-нибудь, то он пропускает уже следующую.

Формат выходного файла

В выходной файл выведите n чисел — рейтинги песен в том порядке, в котором они даны во входном файле.

Примеры

music.in	music.out
2 songa 10 songb 20 5 2 songa songb songb songa songa 40 55	1 0

Задача E. $\sqrt{Nim}$

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

$\sqrt{Nim}$ — игра с использованием камней, которая протекает по следующим правилам. Исходно на столе располагается кучка, содержащая n камней. Два игрока по очереди берут камни из кучки. Если перед ходом игрока кучка содержит k камней, игрок может взять из нее от 1 до $\lfloor\sqrt{k}\rfloor$ камней, включительно. Например, если в кучке 10 камней, разрешается взять 1, 2 или 3 камня. Если в кучке не осталось камней, то игрок, который должен сделать ход, проигрывает.

По заданному n определите, сможет ли первый игрок выиграть, если оба игрока будут придерживаться оптимальной стратегии.

Формат входного файла

Входной файл содержит число n ($1 \le n \le 10^{12}$).

Формат выходного файла

Выведите "`WIN`", если первый игрок может выиграть вне зависимости от действий второго игрока, либо "`LOSE`", если второй игрок может заставить первого игрока проиграть.

Пример

nim.in	nim.out
3	WIN
5	LOSE

Задача F. Произведение цифр

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

Найдите количество чисел из отрезка $[l, r]$, которые делятся на произведение своих цифр.

Формат входного файла

Входной файл содержит два целых числа: l и r ($1 \le l \le r \le 10^9$, $|r - l| \le 10^5$).

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

product.in	product.out
1 12	11

Задача G. Точки и отрезки

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

Дано n отрезков на числовой прямой и m точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка x считается принадлежащей отрезку с концами a и b, если выполняется двойное неравенство $min(a, b) \le x \le max(a, b)$.

Формат входного файла

Первая строка содержит два целых числа n ($1 \le n \le 10^5$) — число отрезков и m ($1 \le m \le 10^5$) — число точек. В следующих n строках по два целых числа a_i и b_i — координаты концов соответствующего отрезка. В последней строке m целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю 10^9.

Формат выходного файла

В выходной файл выведите m чисел — для каждой точки количество отрезков, в которых она содержится.

Примеры

segments.in	segments.out
3 2 0 5 -3 2 7 10 1 6	2 0
1 3 -10 10 -100 100 0	0 0 1

Задача H. Дана строка...

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

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

Дана строка s. Необходимо найти непустую подстроку этой строки, обладающую максимальным качеством, а из всех таких — минимальную по длине.

Формат входного файла

Входной файл содержит непустую строку s, состоящую из строчных букв латинского алфавита. Ее длина не превосходит $2 \cdot 10^5$ символов.

Формат выходного файла

В выходной файл выведите искомую подстроку. Если вариантов ответа несколько, выведите любой.

Примеры

string.in	string.out
aba	ab
zzz	z

Задача I. Туристическое агентство

Имя входного файла:	`travel.in`
Имя выходного файла:	`travel.out`
Ограничение по времени:	3 секунды
Ограничение по памяти:	64 мегабайта

Антон работает в межгалактическом туристическом агентстве. Довольно часто ему приходится прокладывать путь с одной планеты на другую с использованием существующих рейсов космических кораблей. К сожалению, количество рейсов невелико, поэтому пассажирам часто приходится пересаживаться на промежуточных планетах.

Антон заметил, что некоторые планеты используются в качестве промежуточных чаще, чем другие. Он решил провести исследование — для каждой планеты A он хотел бы узнать, сколько существует пар различных планет (B, C), таких что любой путь с планеты B на планету C проходит через планету A.

Помогите Антону!

Формат входного файла

Первая строка входного файла содержит два целых числа: n и m — количество планет и количество рейсов космических кораблей, соответственно ($2 \le n \le 20\,000$, $1 \le m \le 200\,000$). Следующие m строк описывают рейсы космических кораблей. Каждый рейс связывает две планеты, и им можно воспользоваться в любом из двух направлений. С любой планеты можно добраться до любой другой.

Формат выходного файла

Выведите n целых чисел — для каждой планеты A выведите количество пар различных планет, таких что любой путь с одной планеты на другую проходит через A.

Пример

travel.in	travel.out
7 9 1 2 1 3 1 4 1 5 1 6 1 7 2 3 4 5 6 7	18 6 6 6 6 6 6