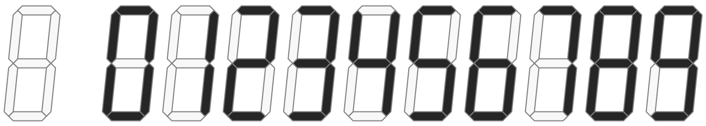
Задача A. Цифровой дисплей

Имя входного файла:	`digits.in`
Имя выходного файла:	`digits.out`
Ограничение по времени:	5 секунд
Ограничение по памяти:	64 мегабайта

Умный мальчик Вася — начинающий математик. Сегодня у него день рождения: собралось много гостей, все дарят ему подарки — все как обычно. Но его лучший друг Вовочка сделал ему необычный подарок: он подарил ему устройство, на котором был цифровой дисплей. Он отображает N-значное число X, с помощью N индикаторов из семи полосок.



Это устройство очень напоминает электронные часы, только оно показывает время не в обычном формате, а в секундах от какого-то очень важного момента. Это было так давно, что Вовочка и не помнит, какое важное событие произошло, когда он впервые включил свое изобретение.

Он рассказал Васе, что по этим часам можно узнать, когда в следующий раз произойдет какое-нибудь важное событие. Для этого надо взять число, которое сейчас на дисплее и переставив в нем не более чем K палочек получить минимальное число, больше данного. При этом разрешается перемещать палочки из одной цифры в другую. Вовочка уже давно планировал подарить это замечательное изобретение своему лучшему другу, поэтому он расчитал тот момент, когда он будет рассказывать это Васе.

Так как Вася не так хорош в математике как Вовочка, он просит вас написать программу, которая даст ответ на его вопрос.

Формат входного файла

Первая строка входного файла содержит целое число K ($1 \le K \le 100$). Вторая строка содержит число X ($0 \le X < 10^{100000}$).

Формат выходного файла

В выходной файл выведите одно число: ответ на задачу, либо «NO SOLUTION», если ответа не существует.

Примеры

digits.in	digits.out
2 4598	4600
3 888	NO SOLUTION

Задача B. Игра «Делимость»

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

Как-то раз, разбирая вещи на чердаке, Петя нашел странную игру. В начале игры на поле выкладываются карточки с числами. Затем двое игроков по очереди берут себе по одной карте за ход. Игра заканчивается, когда на поле не остается карточек. Выигрывает тот, у кого сумма чисел на карточках делится на три. В случае если сумма чисел делится на три у обоих игроков, объявляется ничья. Также ничья объявляется, если ни у одного игрока сумма не делится на три.

Петя выяснил, что эта игра была довольно популярна когда-то, и что у некоторых его друзей она тоже есть. Единственное отличие состоит в том, что набор карт у всех разный.

Теперь Пете интересно, каков будет исход партии при оптимальной игре. Помогите ему выяснить это.

Формат входного файла

Первая строка входного файла содержит натуральное число n — количество карточек ($1 \le n \le 50$). Следующая строка содержит n целых чисел, разделенных пробелами — значения, написанные на карточках. Все числа во входном файле не превосходят 1000 по модулю.

Формат выходного файла

В первой строке выходного файла выведите «FIRST», если при оптимальной игре выигрывает первый игрок, «SECOND», если второй. В случае, если при оптимальной игре случается ничья, выведите «DRAW».

Примеры

divisigame.in	divisigame.out
2 1 3	FIRST
3 3 6 9	DRAW

Задача C. Лепреконское золото

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

Коренной житель Ирландии лепрекон Патрик однажды крупно поссорился со своей женой Клариссой и решил в срочном порядке убежать на остров Бора-Бора. Для этого у мудрого Патрика есть n спрятанных на одной прямой горшочков с лепреконским золотом. Ссора произошла спонтанно, поэтому Патрик не смог запастись достаточным количеством магических амулетов и талисманов, а это значит, что его магических сил хватит лишь на одну телепортацию, но зато в любое место, например — к любому из горшочков с золотом. Эту телепортацию необходимо использовать, до начала сбора горшочков с золотом.

Так как лепреконы по природе своей не очень хорошие бегуны, без помощи телепортаций Патрик может перемещаться со скоростью 1 метр в минуту. Но на один из горшочков Кларисса наложила заклинание исчезновения, и он пропадет через t минут. Помогите Патрику за минимальное время собрать все горшочки с золотом! Если он не заберет хотя бы один из них, ему не хватит золота на путешествие.

Формат входного файла

В первой строке число n — количество горшочков с золотом — и число t — время исчезновения (в минутах) одного из них ($2 \leq n, t \leq 100$). В следующей строке n чисел — координаты горшочков в метрах. Все числа различны и по абсолютной величине не превосходят 100. Координаты горшочков даны в порядке возрастания. В следующей строке записан номер горшочка, который исчезнет через t минут.

Формат выходного файла

В первой строке выходного файла выведите минимальное время, которое потребуется Патрику для сбора всего золота. В следующей строке выведите n чисел — порядок, в котором следует собирать горшочки.

Примеры

gold.in	gold.out
5 5 1 4 9 16 25 2	24 1 2 3 4 5
6 4 1 2 3 6 8 25 5	31 5 4 3 2 1 6

Задача D. Награды

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

На одном Очень Важном Предприятии решили наградить некоторых k его работников. Конечно же, решили сделать это в соответствии со следующей Очень Важной Процедурой.

Всех n работников выстроили в один ряд. Причем, получилось так, что каждый работник видит только своих непосредственных соседей в этом ряду. Для повышения уровня производства на Очень Важном Предприятии начальство решило сделать так, чтобы каждый награжденный считал, что наградили именно его и только его. Для этого необходимо, чтобы в ряду не было двух рядом стоящих награжденных работников.

Вам необходимо написать программу, которая будет считать количество способов раздать таким образом k наград среди n стоящих в ряд работников. Так как это число может быть весьма большим, необходимо найти его остаток от деления на простое число m.

Формат входного файла

В первой и единственной строке входного файла заданы три целых неотрицательных числа n, k и m — количество работников на Очень Важном Предприятии, количество наград и простой модуль ($1 \le k \le n \le 100000, 1 \le m \le 10^9$).

Формат выходного файла

В выходной файл выведите единственное целое число — ответ на задачу, взятый по модулю простого числа m.

Примеры

grants.in	grants.out
3 2 569	1
5 2 673	6

Задача E. Большие скачки

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

В одном очень большом городе устраивают необычные скачки. От обычных скачек они отличаются тем, что проходят не на ипподроме, а на специально заготовленной трассе. Она представляет из себя бесконечную прямую на плоскости.

Трасса очень длинная, поэтому соревнования могут затягиваться не на один день и проходить не только днем, но и ночью. Организаторы глубоко задумались о том, как они будут освещать трассу, ведь освещать бесконечно длинную трассу не так уж и просто. Для этого они закупили N прожекторов, которые будут установлены в некоторых точках города. Известно что прожекторы освещают землю, образуя круги.

Так получилось, что компания, которая устанавливала оборудование, перепутала места установки, поэтому некоторые прожекторы могут вообще не освещать трассу. Теперь соревнование может потерпеть неудачу, организаторы очень обеспокоены тем, что зрители не увидят самые интересные моменты соревнований из-за ошибки мастеров. Помогите организаторам выяснить, какова длина освещенной части трассы.

Формат входного файла

Первая строка входного файла содержит четыре числа x_1, y_1, x_2, y_2 — координаты двух точек на прямой. Во второй — строке число N ($1 \le N \le 100000$) — количество прожекторов. В каждой их следующих N строк заданы 3 числа x, y и R, координаты и радиус кругов, образованных прожекторами.

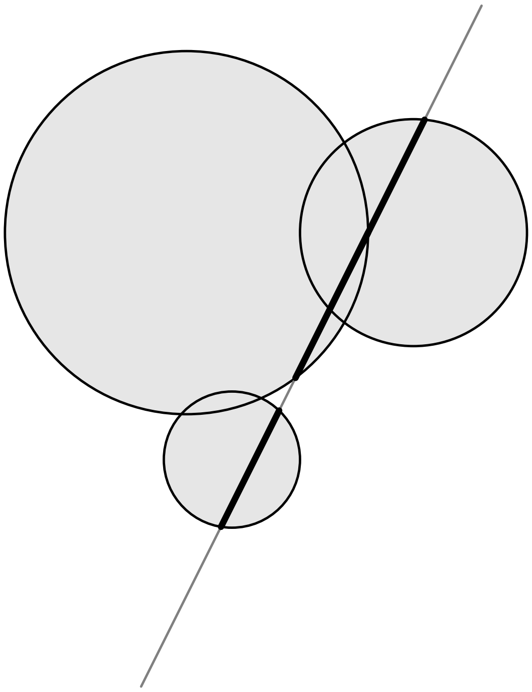
Все координаты и радиусы — целые числа, не превышающие по модулю 10^5.

Формат выходного файла

В выходной файл выведите ответ на задачу, с точностью до 10^{-4}.

Примеры

horseraces.in	horseraces.out
0 0 1 1 1 5 5 1	2.0000
1 1 2 3 3 5 5 5 -5 5 8 -3 -5 3	18.4460



Этот рисунок соответствует второму примеру. Отмеченная жирной линией часть дороги, является освещенной.

Задача F. Острова

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

Островное государство Исола состоит из n островов. Для удобства передвижения между некоторыми островами были построены мосты, но чтобы никакой остров не был перегружен транспортом, к каждому острову ведет не более двух мостов. По мосту можно проезжать в обоих направлениях. Для получения средств на поддержание мостов и дорог правительство Исолы установила плату за проезд по мосту в размере одной условной единицы.

До недавнего времени в Исоле не было автобусного сообщения. В срочном порядке была основана первая автобусная компания «Коррейра», и решено проложить по автобусному маршруту между каждой парой островов. Поскольку между некоторыми островами не существует пути по мостам, между такими островами решено маршрут не создавать.

Было решено, что каждому маршруту будет совершаться два рейса в сутки: сначала в одном направлении, а затем в обратном. Естественно автобусы всегда движутся по самому дешевому маршруту. В «Коррейре» очень интересуются, сколько условных единиц в день будет уходить на оплату проездов автобусов по мостам. Поскольку программистов в небольшом государстве Исолы нет, компания просит Вас решить эту задачу.

Формат входного файла

В первой строке два целых числа n и m ($1 \le n \le 100000$; $0 \le m \le n$) — количество островов и мостов Исолы. Далее следует m строк, описывающих мосты Исолы. В каждой строке содержится два целых числа x и y ($1 \le x, y \le n$; $x \ne y$) — номера островов, соединенных мостом. Гарантируется что к каждому острову ведет не более двух мостов.

Формат выходного файла

В выходной файл выведите единственное целое число — количество условных единиц, необходимых для работы автобусного сообщения.

Примеры

islands.in	islands.out
5 4 1 2 1 3 2 3 5 4	8
5 4 5 4 4 3 3 2 2 1	40

В первом примере не все острова соединены между собой. От первого острова до второго можно добраться по одному мосту, от первого до третьего — один мост, от второго до третьего — один. До четвертого или пятого от первого, второго или третьего островов добраться по мостам нельзя. От четвертого до пятого — один мост. Итого $2(1 + 1 + 1 + 1) = 8$ условных единиц.

Задача G. Ответный матч

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

Две популярные футбольные команды А и Б решили выявить сильнейшего между собой. Для этого они решили провести двухматчевое противостояние — сначала сыграть матч на поле А, затем на поле Б.

Противостояние выигрывает та команда, которая в сумме забьет больше голов, чем соперник. Если же команды забили одинаковое число мячей в ворота соперника, то победителем признается команда, забившая большее число голов в гостях (на стадионе соперника). Если же и эти величины совпадают, то команды исполняют серию пенальти.

Второй матч противостояния на поле Б на радио «Слушаем внимательно» комментировал ведущий Вася. Во время трансляции его слушатели регулярно интересовались не только текущим результатом матча, который Вася зачитывал с табло, но и возможностью такого исхода встречи, что будет исполнена серия пенальти (как самое захватывающее футбольное зрелище).

Математические способности Васи оставляли желать лучшего, и он попросил Вас написать программу, которая, зная результат первого матча и текущий счет на табло во время второго матча, поможет определить, смогут ли зрители наблюдать серию пенальти.

Формат входного файла

Первая строка входного файла содержит результат матча А — Б (на поле команды А) в формате «a:b», где a — число мячей, забитых хозяевами поля, а b, соответственно, гостями ($0 \le a, b < 10$). Вторая строка в том же формате содержит текущий счет на табло в матче Б — А.

Формат выходного файла

В выходной файл выведите слово «`YES`», если команды могут закончить встречу со счетом, после которого результат противостояния определиться в серии пенальти, иначе — выведите слово «`NO`».

Примеры

match.in	match.out
2:1 0:0	YES
2:1 0:2	NO

Задача H. Добыча нефти

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

Компания *RectanGas* занимается добычей нефти уже с незапямятных времен и является бесспорным лидером в своей области. Маркетологи считают, что она стала такой успешной благодаря бессменному исполнительному директору. На нем стоит остановиться поподробнее.

Дело в том, что исполнительный директор совсем не изучал геометрию в школе и до сих пор считает, что кроме прямоугольников других геометрических фигур не существует. Более того, он никак не может поверить, что стороны прямоугольников могут быть не параллельными осям координат. Поэтому все нефтяные вышки строятся следующим образом: директор на карте отмечает четыре точки так, чтобы они образовывали прямоугольник со сторонами, параллельными осям координат карты, и приказывает строить нефтяные вышки на участках, соответствующим отмеченным точкам на карте. Эти нефтяные вышки за короткий промежуток времени добывают всю нефть, расположенную на соответствующем участке.

Компания *RectanGas* почти выполнила свой план по добыче нефти на май 2010 года. Для этого ей осталось добыть ровно S баррелей нефти. Заметим, что больше добывать нельзя, иначе компанию обвинят в монополии. Меньше добывать тоже нельзя, потому что исполнительный директор, будучи человеком пунктуальным, очень огорчится, что повлечет за собой увольнение сотрудников и всеобщую безработицу. Геологи, подробно изучив карту местности, определили количество залежей нефти на каждом участке и попросили Вас написать программу, определяющую координаты будущих нефтяных вышек с учетом всех требований.

Формат входного файла

Первая строка содержит три числа: размеры карты местности n и m и план по добыче нефти S ($2 \le n, m \le 300$; $0 \le S \le 10^7$). Далее следуют n строк по m чисел a_{ij} — количество залежей нефти на соответсвующем участке ($0 \le a_{ij} \le 10^6$)

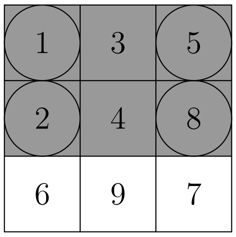
Формат выходного файла

В случае, если увольнения не миновать, выведите «−1» (без кавычек). В противном случае выведите четыре числа — координаты левой верхней и правой нижней вышек соответственно. Если вариантов ответа несколько, то выведите любой из них.

Примеры

oil.in	oil.out
3 3 16 1 3 5 2 4 8 6 9 7	1 1 2 3
2 3 4 12 6 7 4 9 5	-1

Иллюстрация к первому примеру:



Сумма значений в угловых вышках равна $1 + 5 + 2 + 8 = 16$

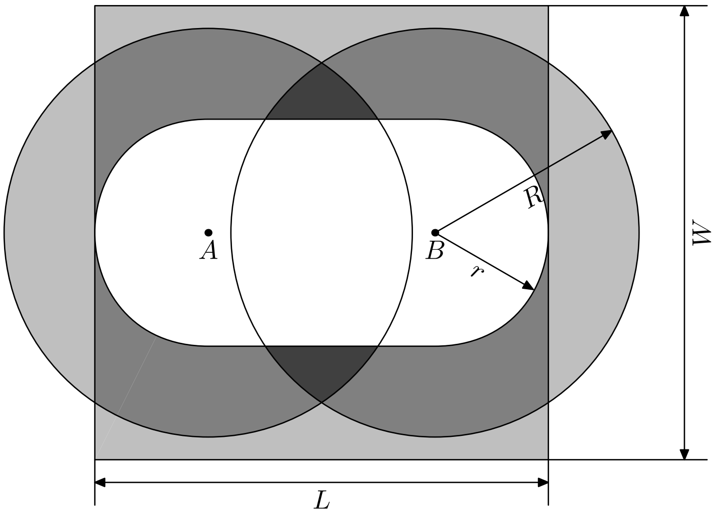
Задача I. Осторожный судья

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

Завтра состоится футбольный матч между двумя знаменитыми командами: Газмясом и Нефтьрыбом. Матч будет проходить на поле длины L и ширины W. Матч будет судить профессиональный футбольный судья в четвертом поколении Вениамин Хлебников.

Быть судьей — ответственное и не всегда безопасное занятие. Поэтому Вениамин решил проработать некоторые игровые эпизоды, которые возникнут в завтрашней игре.

Рассмотрим ситуацию, когда игрок A делает пас игроку B — то есть, передает ему мяч по отрезку, соединяющему точки, в которых находятся игроки. С одной стороны, судья должен хорошо видеть то, что происходит во время паса; с другой стороны, согласно требованиям безопасности, судья не может находиться слишком близко к мячу. Поэтому во время паса судья должен находиться на расстоянии, не меньшем, чем r, и не большем, чем R, от возможного положения мяча. При этом считается, что все то время, в течение которого движется мяч, судья стоит на одном месте. Разумеется, судья должен все время матча находиться на поле.



Так как эти условия достаточно сложны, то даже опытному судье иногда бывает трудно определить, где он должен находиться в момент паса. По этой причине Вениамин хочет перед матчем потренироваться находить те области, где он может находиться, при различных начальных условиях. Для того чтобы сравнить свой ответ с правильным, ему необходима программа, которая по заданным размерам поля, координатам игроков и числам r и R находит площадь тех областей поля, в которых может находиться судья. Помогите ему!

Формат входного файла

В первой строке входного файла даны два целых положительных числа L и W $(1 \le L, W \le 100)$ — длина и ширина поля.

Во второй строке даны целые числа X_A, Y_A, X_B, Y_B — координаты игроков A и B соответственно. Так как игроки находятся на поле, то $0 \le X_A, X_B \le L$, $0 \le Y_A, Y_B \le W$.

В третьей строке даны целые числа r и R $(0 < r < R < 100)$. Известно, что $R \le D$, где $D = \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2}$ — расстояние между игроками A и B.

Формат выходного файла

В выходной файл выведите ответ на задачу. Ответ принимается, если он отличается от корректного не более, чем на 10^{-6}.

Примеры

referee.in	referee.out
20 20 5 10 15 10 5 9	13.95667511974284

Задача J. Zuma 2.0: Возвращение лягушки-идола

Имя входного файла:	`zuma.in`
Имя выходного файла:	`zuma.out`
Ограничение по времени:	5 секунд
Ограничение по памяти:	64 мегабайта

В одиннадцатом классе Витя познакомился с игрой *Zuma*. Она буквально захватила его разум. Целую неделю он, приходя из школы, до часа ночи щелкал мышкой, уничтожал цепочки разноцветных шариков, громко радовался, когда последний шарик оказывался уничтожен, и выражал отчаяние, когда не успевал спастись.

К счастью, скоро настало время выпускных экзаменов, игра была позабыта, а вскоре потерялась при обновлении системы. Освободившееся время Витя потратил с пользой: сдал экзамены на «хорошо» и «отлично», поступил в университет, а через шесть лет, успешно защитив магистерскую диссертацию, устроился на работу ведущим программистом.

Беда пришла, откуда не ждали. Недавно вышла новая версия игры, *Zuma 2.0*. В этой версии сюжет игры сделался более разнообразным, в частности, на разных уровнях игры действуют разные правила. Игра снова захватила Витю... простите, теперь уже Виктора Сергеевича. Однако, Виктор Сергеевич осознает, что игру нужно пройти как можно быстрее, чтобы вернуться к своим должностным обязанностям. В частности, ему никак не дается предпоследний уровень, поэтому он просит Вас помочь ему сделать это.

На предпоследнем уровне изначально присутствуют N шаров, которые расположены по окружности. На каждом из шаров изображена прописная латинская буква. Игрок может делать выстрелы, которые уничтожают несколько шаров. А именно, если на уровне осталось не менее пяти шаров, то игрок может выбрать любые пять последовательных шаров и выполнить с ними одно из двух возможных действий:

- Уничтожить любые два шара, на которых написаны одинаковые буквы.
- Уничтожить любые три шара, на которых написаны гласные буквы, а именно, «A», «E», «I», «O», «U», «Y» (см. примечание).

После выстрела оставшиеся шары смыкаются, вновь образуя окружность.

Если на уровне осталось менее пяти шаров, то игрок получает очки — чем меньше шаров осталось, тем больше очков он получает. Если же осталось пять или более шаров, а выстрел сделать не получается, то уровень считается непройденным.

Вам дана строка, которая состоит из букв, написанных на шарах. Найдите минимальное число шаров, которые можно оставить на уровне, играя оптимально.

Формат входного файла

Во входном файле дана строка S. Длина строки S составляет от 5 до 20 символов. Строка состоит из прописных латинских букв.

Формат выходного файла

В выходной файл выведите ответ на задачу.

Примеры

zuma.in	zuma.out
RANGER	4
YELLOW	3

Примечание

В настоящее время нет единого мнения о том, считать ли букву «Y» гласной или согласной. В английском языке есть примеры ее применения как в качестве гласной (tycoon, salary, rhythm), так и в качестве согласной (year, coyote, way). Чтобы разрешить противоречия, в рамках данной задачи было принято решение считать букву «Y» гласной.