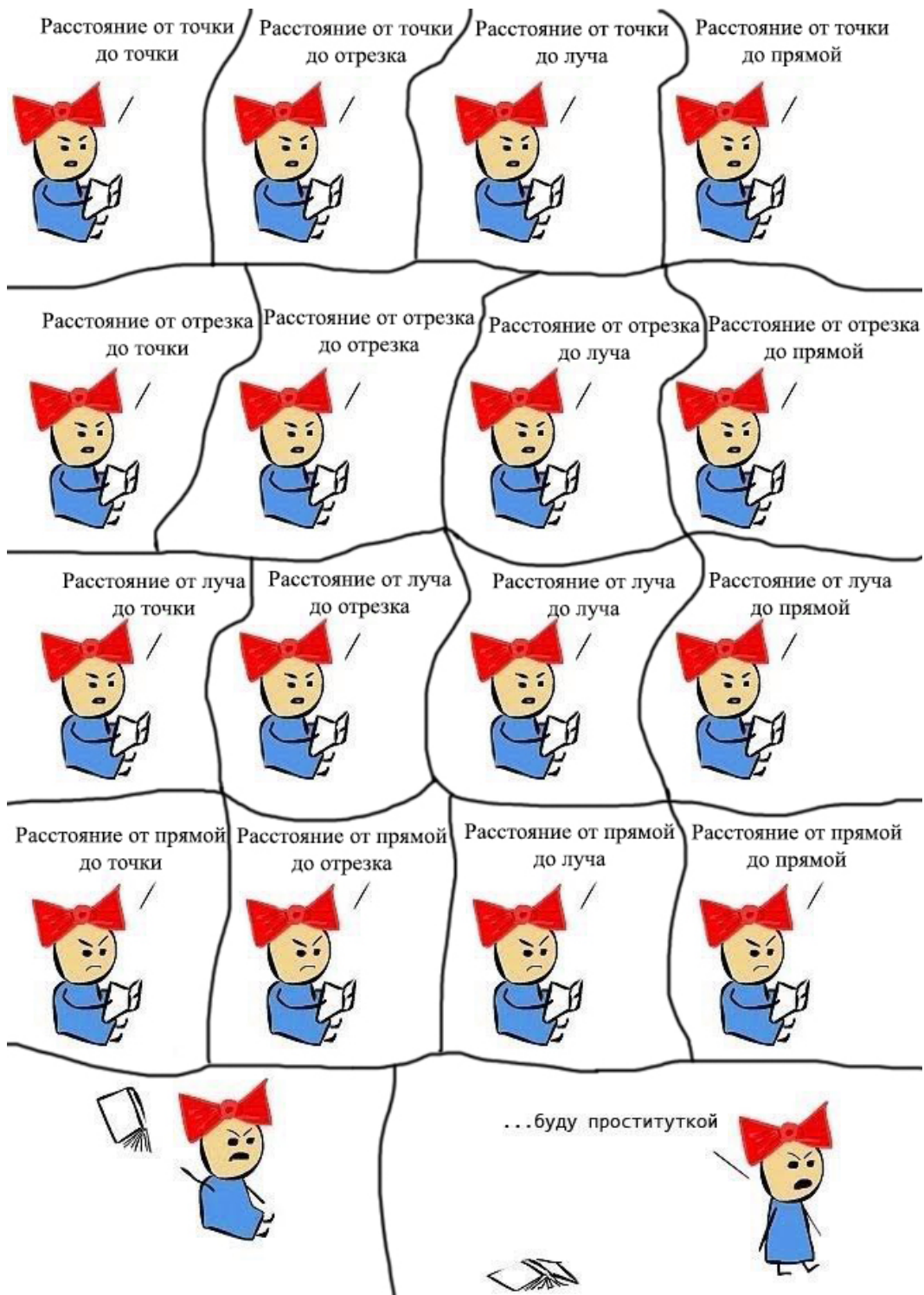


Задача А. 16

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Даны 4 точки A, B, C, D.
Посчитайте:



Даны 4 точки A, B, C, D.

Посчитайте:

- Расстояние от точки A до точки C.
- Расстояние от точки A до отрезка CD.
- Расстояние от точки A до луча CD.
- Расстояние от точки A до прямой CD.
- Расстояние от отрезка AB до точки C.
- Расстояние от отрезка AB до отрезка CD.
- Расстояние от отрезка AB до луча CD.
- Расстояние от отрезка AB до прямой CD.
- Расстояние от луча AB до точки C.
- Расстояние от луча AB до отрезка CD.
- Расстояние от луча AB до луча CD.
- Расстояние от луча AB до прямой CD.
- Расстояние от прямой AB до точки C.
- Расстояние от прямой AB до отрезка CD.
- Расстояние от прямой AB до луча CD.
- Расстояние от прямой AB до прямой CD.

Формат входных данных

Даны координаты четырех точек, по одной точке в строке: $X_a, Y_a, X_b, Y_b, X_c, Y_c, X_d, Y_d$. Все числа целые, по модулю не превосходят 10000.

Формат выходных данных

Выведите 16 чисел по одному в строке. Числа в ответе должны быть выданы с точностью не менее 6 знаков после десятичной точки.

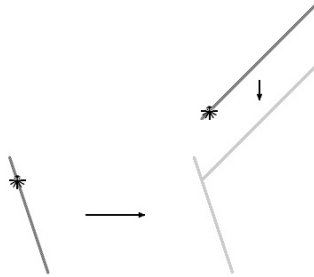
Пример

стандартный ввод	стандартный вывод
1 2	5.6568542495
7 1	5.6000000000
5 6	5.6000000000
8 2	5.6000000000
	4.6031716446
	1.4142135624
	1.4000000000
	1.4000000000
	4.6031716446
	1.1507929111
	0.0000000000
	0.0000000000
	4.6031716446
	1.1507929111
	0.0000000000
	0.0000000000

Задача В. Про любовь...

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Паук и паучиха плывут по озеру на двух веточках. Плавать они не умеют, поэтому смогут встретиться только тогда, когда веточки соприкоснутся.



Считая, что веточки имеют форму отрезков, и что они плывут с постоянными скоростями, определите, сколько осталось ждать встречи несчастным членистоногим.

Формат входных данных

Входной файл содержит 12 чисел: $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4, v_{1x}, v_{1y}, v_{2x}, v_{2y}$. Координаты вершин первого отрезка: (x_1, y_1) и (x_2, y_2) , координаты вершин второго отрезка: (x_3, y_3) и (x_4, y_4) , скорость первого отрезка (v_{1x}, v_{1y}) , скорость второго отрезка (v_{2x}, v_{2y}) . Все числа целые и не превосходят по модулю 10^4 . В начальный момент времени веточки не соприкасаются.

Гарантируется, что веточки имеют ненулевую длину.

Формат выходных данных

Выведите в выходной файл время до ближайшего момента, когда веточки соприкоснутся, с ошибкой не более 10^{-4} . Если веточки не соприкоснутся никогда, выведите число -1 .

Примеры

стандартный ввод	стандартный вывод
0 0 -1 3 4 4 7 7 3 0 0 -1	1.6000000000
0 0 -1 3 4 4 7 7 1 0 0 -3	-1

Задача С. Теодор Рузвельт

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

«Теодор Рузвельт» — флагман военно-морского флота Кукуляндии. Заклятые враги кукуляндцев, флатландцы, решили уничтожить его. Они узнали, что «Теодор Рузвельт» представляет собой выпуклый многоугольник из n вершин и узнали его координаты. Затем они выпустили m баллистических ракет и определили координаты точек, где эти ракеты взорвались. По расчётам штаба флатландцев, «Теодор Рузвельт» будет уничтожен, если в него попадёт хотя бы k ракет. Вычислите, удалось ли флатландцам уничтожить корабль.

Формат входных данных

В первой строке через пробел записаны целые числа n, m, k ($3 \leq n \leq 10^5, 0 \leq k \leq m \leq 10^5$). В последующих n строках записаны координаты вершин многоугольника в порядке обхода против часовой стрелки. В следующих m строках записаны координаты точек. Гарантируется, что все координаты — целые числа, не превосходящие по модулю 10^9 .

Формат выходных данных

Выведите «YES», если в многоугольнике или на его границе лежит по крайней мере k точек, и «NO» в противном случае.

Пример

стандартный ввод	стандартный вывод
5 4 2 1 -1 1 2 0 4 -1 2 -1 -1 -2 -1 1 -1 0 1 2 3	YES

Задача D. Тигры HARD

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

В заповеднике живут q тигров. Чтобы следить за положением тигров на территории заповедника, используются ошейники с радиомаяком. Ошейник у каждого тигра имеет радиомаяк с уникальным сигналом. Система обнаружения настраивается на приём сигнала радиомаяка от i -го тигра последовательно для i от 1 до q .

Для приёма сигнала на территории заповедника установлено n приёмников в точках с координатами $(x_1, y_1), \dots, (x_n, y_n)$. Система обнаружения позволяет сотруднику заповедника за один запрос выбрать любые m ($3 \leq m \leq n$) приёмников. Выбранные приёмники должны являться вершинами выпуклого многоугольника. Система определяет, находится ли радиомаяк i -го тигра внутри этого многоугольника.

Сотрудник заповедника должен локализовать положение каждого тигра. Положение i -го тигра считается локализованным, если удалось определить такое множество приёмников, являющихся вершинами выпуклого многоугольника, что внутри этого многоугольника находится тигр, но нет других приёмников.

Для того, чтобы локализовать положение каждого из тигров, сотруднику разрешается сделать не более k запросов.

После того как положение i -го тигра локализовано, система автоматически переходит к приёму сигналов от следующего тигра, пока положение всех q тигров не будет локализовано.

Гарантируется, что никакие три приёмника не лежат на одной прямой, и ни один тигр не находится на прямой, проходящей через два приёмника. Гарантируется, что существует хотя бы один выпуклый многоугольник с вершинами в приёмниках, внутри которого находится тигр.

Требуется написать программу, которая взаимодействует с программой жюри и локализует положение каждого тигра.

Формат входных данных

Это интерактивная задача.

Сначала на вход подаётся информация об установленных в заповеднике приёмниках и количестве тигров.

Первая строка входных данных содержит целое число n — количество приёмников ($3 \leq n \leq 5000$). Последующие n строк описывают координаты приёмников, j -я из этих строк содержит два целых числа x_j и y_j — координаты j -го приёмника ($-10^9 \leq x_j, y_j \leq 10^9$). Следующая строка содержит число целое число q — количество тигров ($1 \leq q \leq 2000$).

Для локализации положения тигров необходимо выполнять запросы к системе обнаружения, роль которой выполняет программа жюри.

Для каждого теста зафиксировано число k — максимальное количество запросов к системе обнаружения для локализации положения одного тигра. Гарантируется, что k запросов достаточно, чтобы решить задачу для соответствующих данных. Это число не сообщается программе-решению. Если программа-решение делает более k запросов для определения местоположения одного из тигров, на этом тесте она получает в качестве результата тестирования «Неверный ответ». При $n > 2500$ $k = 10000$. При $n \leq 2500$ $k = 60$.

Запрос к системе обнаружения начинается с символа «?», за которым следует целое число m — количество выбранных в запросе приёмников ($3 \leq m \leq n$), и m различных целых чисел p_i — номера приёмников, перечисленные в порядке обхода многоугольника по или против часовой стрелки ($1 \leq p_i \leq n$).

В ответ программа получает строку «Yes», если тигр находится внутри многоугольника, образованного приёмниками с номерами p_1, \dots, p_m , и строку «No» в противном случае.

После того, как положение тигра локализовано, программа-решение должна вывести строку, начинающуюся с символа «!», за которым следует целое число m — количество выбранных приёмников ($3 \leq m \leq n$), и m различных целых чисел p_i — номера приёмников, перечисленные в порядке

обхода многоугольника по или против часовой стрелки ($1 \leq p_i \leq n$). Эта строка означает, что внутри выпуклого многоугольника, образованного приёмниками с номерами p_1, \dots, p_m , находится тигр и нет других приёмников.

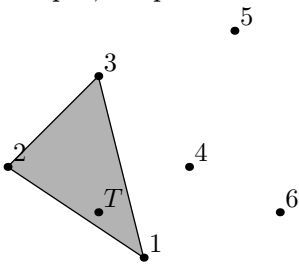
Ответное сообщение от программы жюри отсутствует, и программа-решение должна немедленно приступить к поиску следующего тигра. Локализовав положение тигра с номером q , программа-решение должна завершить работу.

Тигры не перемещаются во время работы системы обнаружения. Координаты тигров в каждом тесте фиксированы и не меняются в процессе тестирования.

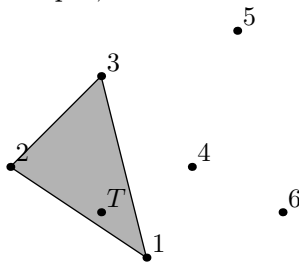
Если существует несколько правильных многоугольников, локализующих положение тигра, можно вывести любой из них.

На рисунке продемонстрирована процедура локализации положения каждого из тигров из приведенного ниже примера.

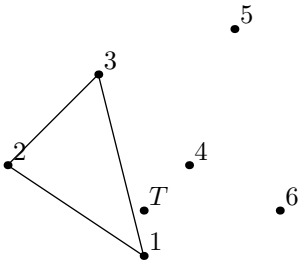
Тигр 1, запрос 1:



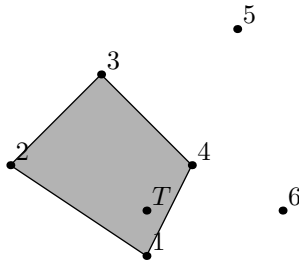
Тигр 1, ответ:



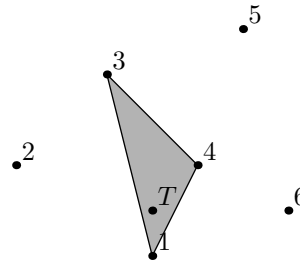
Тигр 2, запрос 1:



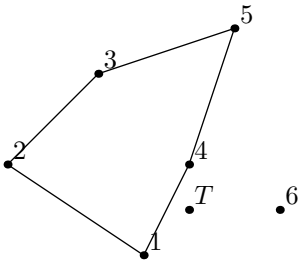
Тигр 2, запрос 2:



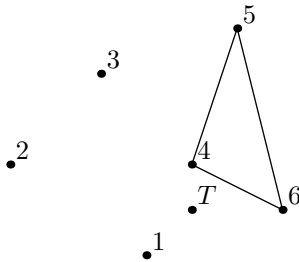
Тигр 2, ответ:



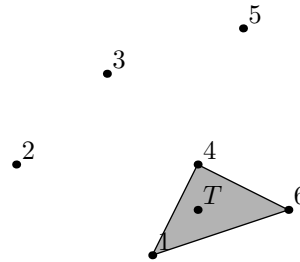
Тигр 3, запрос 1:



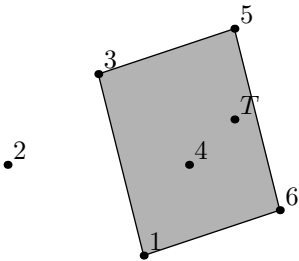
Тигр 3, запрос 2:



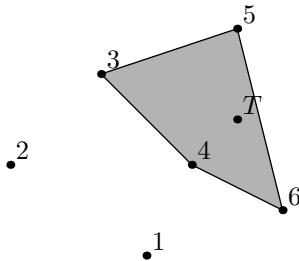
Тигр 3, ответ:



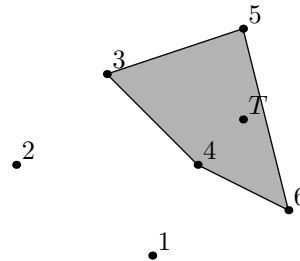
Тигр 4, запрос 1:



Тигр 4, запрос 2:



Тигр 4, ответ:



Пример

стандартный ввод	стандартный вывод
6	
3 0	
0 2	
2 4	
4 2	
5 5	
6 1	
4	
Yes	? 3 1 2 3
	! 3 1 2 3
	? 3 1 2 3
No	
	? 4 1 2 3 4
Yes	
	! 3 4 3 1
	? 5 2 3 5 4 1
No	
	? 3 4 5 6
No	
	! 3 1 4 6
	? 4 1 3 5 6
Yes	
	? 4 4 3 5 6
Yes	
	! 4 4 3 5 6

Замечание

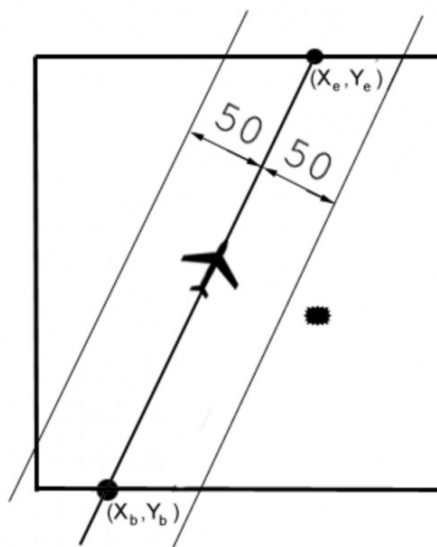
Приведённые примеры иллюстрируют взаимодействие программы-решения с программой жюри «по шагам», для чего в них добавлены дополнительные пустые строки. При реальном тестировании лишние пустые строки вводиться не будут, выводить пустые строки также не требуется.

Задача Е. Лесной пожар

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

В МЧС поступило сообщение о возможном лесном пожаре в заданном квадрате тайги. Для поиска места возгорания было послано N самолетов. Однако ни один из экипажей пожар не обнаружил.

Известно, что с самолета видна полоса тайги, границы которой находятся на расстоянии 50 км справа и слева от той линии на поверхности Земли, над которой пролетает самолет (см. рисунок), причем точки, находящиеся на расстоянии ровно 50 км от этой линии, все еще видны. Донесение с каждого самолета содержало информацию о том, в каких двух различных точках (x_b, y_b) и (x_e, y_e) самолет входил в заданный квадрат и покидал его соответственно. Между этими точками самолет двигался строго по прямой.



Требуется написать программу, которая определит, действительно ли весь заданный квадрат тайги был просмотрен с самолетов. Если это не так, то программа должна находить координаты какой-нибудь точки, находящейся внутри или на границе квадрата и не попавшей ни в одну из просмотренных полос.

Формат входных данных

В первой строке записано натуральное число L — размер заданного квадрата тайги в километрах ($0 \leq L \leq 1000$).

Во второй строке — натуральное число N ($1 \leq N \leq 100$) — количество самолетов.

В каждой из последующих N строк записано донесение с самолета — четыре вещественных координаты x_b, y_b, x_e, y_e . Координаты заданы в километрах. Стороны квадрата тайги параллельны осям координат, его левый нижний угол находится в точке с координатами $(0, 0)$, а правый верхний — в точке (L, L) .

Формат выходных данных

Если заданный квадрат был просмотрен полностью, то выведите единственное слово «YES», написанного заглавными латинскими буквами. В противном случае выведите слово «NO», а в следующей строке выведите через пробел координаты x и y какой-либо точки, которая не попала ни в одну из просмотренных полос. Координаты нужно выводить в километрах с ошибкой не более одного метра (в одном километре 1000 метров).

Примеры

стандартный ввод	стандартный вывод
245 1 26.1 0 193.568 245	NO 155.123 100
50 1 0 25 25 25	YES

Замечание

Вы можете написать тут нормальное решение и умереть. А можете вспомнить про точку в квадрате, непокрытую окружностями.

Задача F. Большой треугольник

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 8 секунд
Ограничение по памяти: 1024 мегабайта

Есть странная особенность: если соединить между собой города Ростов, Таганрог, Шахты, то получится треугольник

«Невероятно, но факт»

В ЛКШ приезжают ученики из самых разных уголков России и зарубежья. Вы отметили на карте все города, откуда приехали участники ЛКШ.

Затем, вы решили подготовить интересную инфографику на основе этой карты. Первое что вы захотели сделать — это найти три города на этой карте, так чтобы получился треугольник площади ровно S .

Формат входных данных

В первой строке входных данных находится два целых числа n и S ($3 \leq n \leq 3333$, $1 \leq S \leq 2 \cdot 10^{18}$) — количество городов на карте и требуемая площадь треугольника.

В следующую n строках находится описание городов, по одной на строке. Каждый город описывается своими координатами x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$).

Гарантируется, что все города находятся в различных точках. Также гарантируется, что никакие три города не лежат на одной прямой.

Формат выходных данных

Если решения не существует — выведите «No» (без кавычек).

Иначе выведите «Yes» (без кавычек), после чего выведите три пары координат — координаты городов, образующих треугольник площади S .

Примеры

стандартный ввод	стандартный вывод
3 7 0 0 3 0 0 4	No
4 3 0 0 2 0 1 2 1 3	Yes 0 0 1 3 2 0

Задача G. Астрономия

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	512 мегабайт

18 год нашей эры. Известный астроном Филон Берляндский публикует трактат «Об устройстве небесного свода», в котором рассказывает о невероятном явлении, замеченном им во время наблюдений за звёздами. Однажды ночью Филон увидел на безоблачном небе $2n$ звёзд и Луну. Удивительно, что звёзды можно было мысленно разбить на пары так, что любая прямая, проходящая через центры двух звёзд из пары проходила также и через центр Луны, при этом все такие прямые были различны. Филон тщательно запечатлел данное явление на карте звёздного неба, в которой была введена система координат, и обнаружил, что центры всех звёзд и центр Луны оказались в точках с целочисленными координатами. Поскольку Филон считал, что Земля и Луна плоские, система координат на карте была двумерная. Система координат была выбрана астрономом так, чтобы координаты всех объектов, включая Луну, по модулю не превосходили 10^6 . Помимо этого, никакие два объекта (две звезды или звезда и Луна) не были в одной точке.

Помимо карты звёздного неба Филон Берляндский написал в своём трактате предсказание, что спустя 2000 лет звёзды на небе вернутся в то же положение, а на месте Луны появится огромная комета, которая уничтожит Землю.

2018 год нашей эры. Вам в руки попадает трактат Филона Берляндского, и вы с ужасом обнаруживаете, что звёзды на небе находятся в том же положении, что и 2000 лет назад! К сожалению, время не пощадило карту астронома, поэтому на ней остались отмеченными только точки, соответствующие центрам звёзд, и не осталось никаких упоминаний, каким образом следовало разбить точки на пары, чтобы все построенные по этим парам прямые прошли через центр Луны. Что ещё хуже, на карте стёрлась точка, соответствующая центру Луны. Чтобы узнать, откуда прилетит комета, и спасти человечество от неминуемой гибели, необходимо срочно восстановить какую-нибудь подходящую позицию центра Луны!

Формат входных данных

В первой строке задано целое число n ($2 \leq n \leq 2600$) — количество пар звёзд, которые астроном увидел на небе.

В следующих $2n$ строках содержатся пары целых чисел x_i, y_i ($-10^6 \leq x_i, y_i \leq 10^6$) — координаты центров звёзд на карте. Обратите внимание, что звёзды указаны в произвольном порядке, никак не связанном с тем, как Филон Берляндский разделял их на пары. Центры никаких двух звёзд не находятся в одной точке.

Формат выходных данных

Если астроном ошибся, и не существует способа разбить все точки на пары таким образом, чтобы все построенные по этим прямые были различны и пересекались в одной точке с целыми координатами, отличной от центров всех звёзд, то выведите «No» (без кавычек) в единственной строке вывода.

В противном случае в первой строке выведите «Yes» (без кавычек). Во второй строке выведите пару целых чисел x, y ($|x|, |y| \leq 10^6$) — координаты точки, в которой находился центр Луны в вашем решении. Если подходящих точек несколько, то выведите любую из них. Обратите внимание, что выведенная точка не может совпадать с центром ни одной из звёзд.

Примеры

стандартный ввод	стандартный вывод	Пояснение
2 1 1 1 3 3 1 3 3	Yes 2 2	
3 4 2 6 2 2 1 2 6 4 4 6 6	Yes 2 2	
2 1 1 2 2 4 4 5 5	No	
2 1 1 2 1 1 2 2 2	No	
2 0 0 1 1 -1 -1 -1 1	No	

Замечание

В четвёртом тесте из условия центр Луны мог располагаться только в точке $(1.5, 1.5)$, но эта точка обладает нецелыми координатами, поэтому ответа нет.

В пятом тесте из условия невозможно найти подходящую точку, которая бы не являлась при этом центром звезды.

Задача Н. Парк

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В столице Байтленда есть огороженный парк, территория которого представляет собой прямоугольник. Деревья и посетители в парке представлены в виде кругов.

В парке четыре входа, по одному в каждом углу (1 — внизу слева, 2 — внизу справа, 3 — вверху справа, 4 — вверху слева). Посетители могут входить и выходить из парка только через входы. Посетители могут входить и выходить из парка, когда они касаются обеих сторон угла соответствующего входа. Посетители могут свободно перемещаться по парку, но они не могут перекрывать деревья и заборы.

Ваша задача — рассчитать для каждого посетителя, учитывая вход, в который они войдут в парк, через какие выходы они могут выйти из парка.

Формат входных данных

Первая строка ввода содержит два целых числа n и m ($1 \leq n \leq 2000$ $1 \leq m \leq 100\,000$) — количество деревьев в парке и количество посетителей.

Вторая строка ввода содержит два целых числа w и h ($1 \leq w, h \leq 10^9$) — ширину и высоту парковой зоны. Нижний левый угол имеет координаты $(0, 0)$, а верхний правый угол имеет координаты (w, h) .

После этого есть n строк, которые описывают деревья. Каждая строка содержит три целых числа x , y и r , что означает, что центр i -го дерева (x, y) и его радиус r . Деревья не перекрывают друг друга или забор.

Наконец, есть m строк, которые описывают посетителей. Каждая строка содержит два целых числа r и e — радиус посетителя и вход, в который они войдут в парк.

Кроме того, ни одно дерево не перекрывает квадратную область $2k \times 2k$ в каждом углу, где k — радиус наибольшего посетителя.

Формат выходных данных

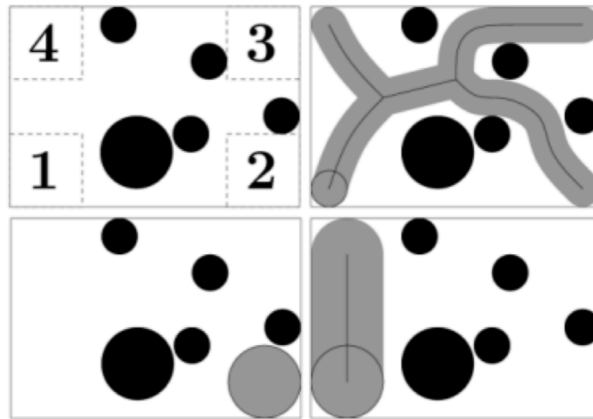
Вы должны вывести для каждого посетителя одну строку, содержащую входы, через которые они могут выйти из парка, в отсортированном порядке без пробелов между ними.

Пример

стандартный ввод	стандартный вывод
5 3	1234
16 11	2
11 8 1	14
6 10 1	
7 3 2	
10 4 1	
15 5 1	
1 1	
2 2	
2 1	

Замечание

На следующем рисунке показаны зоны входа и возможные маршруты для каждого посетителя:



Задача I. Бульдозер

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

Королевство JOI известно добычей золота. Каждый год они используют бульдозер для его добычи.

Земля королевства описывается плоскостью с декартовыми координатами. Есть n мест, i -е место имеет координаты (x_i, y_i) . В каждом месте возможно добыть золото или камни, но нельзя одновременно и то, и другое.

Если в i -м месте можно добыть золото, то его ценность там равна v_i . Если же там камни, то, чтобы избавиться от них, необходимо потратить c_i .

Мы используем бульдозер следующим образом. Сначала мы выбираем две параллельные прямые, затем добываем всё золото и камни, находящиеся между этими прямыми, включая их границы.

Выгода равна сумме ценностей золота за вычетом трат на избавление от камней. Необходимо максимизировать выгоду.

Формат входных данных

Первая строка содержит целое число n ($1 \leq n \leq 2000$) — количество мест.

Далее следуют n строк, i -я из которых содержит три целых числа x_i, y_i, w_i ($|x_i|, |y_i|, |w_i| \leq 10^9$, $w_i \neq 0$) — координаты места (x_i, y_i) и специальное число.

- Если $w_i \geq 1$, то i -е место содержит золото, и $v_i = w_i$.
- Если $w_i \leq -1$, то i -е место содержит камни, и $c_i = -w_i$.

Гарантируется, что все точки различны.

Формат выходных данных

Выведите одно число — ответ.

Примеры

стандартный ввод	стандартный вывод
5 -5 5 -2 2 5 10 1 4 -2 4 -5 4 -2 2 7	19
6 0 0 6 1 0 -2 2 0 8 0 1 -2 1 1 5 2 1 -2	15
5 0 0 2 4 0 2 3 2 -1 1 2 2 1 1 -1	5
2 0 0 -1 1 0 -1	0
15 10 3 30 5 10 -17 4 -5 14 0 -3 -9 -2 3 17 6 9 -19 -9 -6 -14 -2 -3 10 -3 -3 30 8 1 -28 9 -9 -5 7 -5 -24 -8 -10 5 -7 2 20 10 -3 -13	107