Задача A. Куда повернуть?

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Робот Пётр стоит на плоскости в начале координат и смотрит в направлении на точку $u = (x_u, y_u)$.

Задача робота — стоя в начале координат, повернуться так, чтобы смотреть в направлении на точку $v = (x_v, y_v)$.

В какую сторону ближе поворачивать: по часовой стрелке или против часовой стрелки?

Например, если Пётр смотрит на точку $u = (2, 0)$ и должен посмотреть на точку $v = (0, 2)$ (как в верхнем примере), ему придётся повернуть либо на 90 градусов (или $\pi/2$ радиан) против часовой стрелки, либо на 270 градусов (или $3\pi/2$ радиан) по часовой стрелке. Поворачивать против часовой стрелки ближе.

Формат входных данных

В первой строке задано целое число t — количество наборов входных данных ($1 \leqslant t \leqslant 10^4$). Далее для каждого набора:

В единственной строке заданы четыре целых числа в формате "x_u y_u x_v y_v" — координаты точек ($-10^9 \leqslant x_u, y_u, x_v, y_v \leqslant 10^9$). Оба вектора u и v ненулевые.

Формат выходных данных

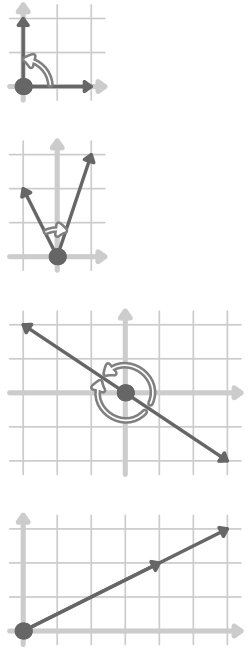
Для каждого набора входных данных:

В отдельной строке выведите ответ:

- «`clockwise`», если поворачивать по часовой стрелке ближе, чем против часовой стрелки;
- «`counterclockwise`», если поворачивать против часовой стрелки ближе, чем по часовой стрелке;
- «`any`», если эти два варианта одинаковы;
- «`none`», если поворот не требуется.

Каждую букву можно выводить в любом регистре.

Пример

стандартный ввод	*стандартный вывод*	*пояснение*
4 2 0 0 2 -1 2 1 3 3 -2 -3 2 4 2 6 3	counterclockwise clockwise any none	

Задача B. Угол между векторами

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Необходимо найти угол между двумя векторами на плоскости в радианах.

Формат входных данных

В первой строке задано четыре целых числа — координаты двух невырожденных векторов, не превосходящие $10\,000$ по абсолютной величине.

Формат выходных данных

Выведите одно число — величину неориентированного угла между ними с точностью до пятого знака после десятичной точки из отрезка $[0, \pi]$.

Пример

стандартный ввод	*стандартный вывод*
2 1 3 5	0.56673

Задача C. Точка пересечения прямых

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Даны две прямые на плоскости. Каждая прямая задаётся двумя точками. Найдите точку пересечения данных прямых.

Формат входных данных

Ввод состоит из двух строк. Каждая из них задаёт одну прямую и содержит четыре целых числа x_1, y_1, x_2, y_2 — координаты двух различных точек, через которые проходит прямая ($|x_1|, |y_1|, |x_2|, |y_2| \leqslant 10\,000$).

Формат выходных данных

Если точка пересечения прямых существует и единственна, выведите в первой строке два вещественных числа — координаты точки пересечения. Абсолютная погрешность не должна превышать 10^{-6}.

Если точки пересечения не существует, выведите в первой строке единственное число: 0.

Если прямые совпадают, выведите -1.

Примеры

стандартный ввод	*стандартный вывод*
0 0 1 1 0 1 1 0	0.5 0.5
1 5 5 5 2 5 1 5	-1

Задача D. Пересечение отрезков

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Два отрезка $[A, B]$ и $[C, D]$ на плоскости заданы координатами своих концов — точек A, B, C, D, имеющих координаты (X_a, Y_a), (X_b, Y_b), (X_c, Y_c) и (X_d, Y_d).

Требуется найти пересечение этих отрезков и вывести:

- слово `Empty`, если эти отрезки не пересекаются;
- координаты точки пересечения, если пересечение состоит из единственной точки;
- координаты точек — начала и конца отрезка пересечения в лексикографическом порядке, если пересечение заданных отрезков — отрезок. То есть первой следует выводить точку с минимальным x, а если координаты x равны, то с минимальным y.

Формат входных данных

Во входных данных содержатся восемь целых чисел, не превосходящих 1000 по абсолютной величине — координаты (X_a, Y_a), (X_b, Y_b), (X_c, Y_c) и (X_d, Y_d). Отрезки могут быть вырожденными.

Формат выходных данных

Числовые значения в ответе должны отличаться от правильных не более чем на 10^{-6}.

Примеры

стандартный ввод	*стандартный вывод*
0 0 9 9 9 5 0 5	5.000000 5.000000
0 0 9 9 15 15 7 7	7.000000 7.000000 9.000000 9.000000
0 0 9 9 10 10 10 10	Empty

Задача E. Две окружности

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Заданы две окружности на плоскости. Задача заключается в нахождении всех точек их пересечения.

Формат входных данных

Первая строка содержит число наборов входных данных K ($1 \leqslant K \leqslant 10\,000$). Каждый набор состоит из двух строчек, каждая из которых описывает окружность. Описание окружности задаётся в виде трёх чисел x, y и r, разделённых пробелами ($-100 \leqslant x, y \leqslant 100$, $0 < r \leqslant 100$). Все числа во вводе целые.

Формат выходных данных

Для каждого из наборов необходимо вывести одно из нижеследующих сообщений:

- «`There are no points!!!`» — если точки пересечения отсутствуют.
- «`There are only` i `of them....`» — если окружности имеют в точности i точек пересечения. В этом случае последующие i строчек должны содержать координаты точек x'_j и y'_j. Точки требуется выводить в порядке возрастания (сначала с меньшими x, если значения x равны, то с меньшими y). Числа необходимо выводить не менее чем с четырьмя точными знаками после десятичной точки.
- «`I can't count them - too many points :(`» — если точек пересечения бесконечно много.

Все сообщения необходимо выводить без кавычек.
Разделяйте вывод для различных наборов пустой строкой.

Пример

стандартный ввод
2 0 0 2 4 0 2 0 0 1 100 100 1
стандартный вывод
There are only 1 of them.... 2.00000000000000 0.00000000000000 There are no points!!!

Задача F. Принадлежность точки отрезку

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Требуется проверить, принадлежит ли заданная точка заданному отрезку (концы включаются в отрезок) на плоскости.

Формат входных данных

В первой строке задано шесть целых чисел — координаты точки и координаты концов отрезка, не превосходящие $10\,000$ по абсолютной величине.

Формат выходных данных

Выведите «YES», если точка принадлежит отрезку, и «NO» в противном случае.

Пример

стандартный ввод	*стандартный вывод*
3 3 1 2 5 4	YES

Задача G. Точка пересечения медиан

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Дан невырожденный треугольник. Найдите точку пересечения его медиан.

Формат входных данных

В первой строке находятся шесть целых чисел — координаты трёх вершин треугольника.

Гарантируется, что три вершины не лежат на одной прямой и удовлетворяют ограничениям: $|x_i|, |y_i| \leqslant 10\,000$.

Формат выходных данных

В первой строке выведите два вещественных числа — координаты точки пересечения медиан треугольника. Абсолютная погрешность не должна превышать 10^{-6}.

Пример

стандартный ввод	*стандартный вывод*
0 0 0 2 3 1	1 1

Задача H. Площадь треугольника

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Даны три точки на плоскости. Какова площадь треугольника с вершинами в этих точках?

Формат входных данных

В первых трёх строках входных данных заданы координаты точек. Каждая из этих строк содержит два целых числа x_i и y_i, разделённых пробелом ($|x_i|, |y_i| \leqslant 100$). Учтите, что вершины треугольника могут лежать на одной прямой и даже совпадать.

Формат выходных данных

В первой строке выведите одно число — площадь треугольника с вершинами в данных точках. Выводите ответ с точностью не менее шести знаков после десятичной точки.

Примеры

стандартный ввод	*стандартный вывод*
0 0 0 1 1 1	0.5
-2 -2 0 0 2 3	1

Задача I. Площадь выпуклого многоугольника

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Многоугольник называется *выпуклым*, если любой отрезок, соединяющий две точки внутри или на сторонах многоугольника, не выходит за границы многоугольника.

Например, любой треугольник является выпуклым многоугольником, а «звезда» не является.

Дан выпуклый многоугольник. Найдите его площадь.

Формат входных данных

В первой строке находится одно целое число N — количество вершин многоугольника. Далее следуют N строк, каждая из которых содержит два целых числа x_i, y_i — координаты вершины многоугольника.

Вершины перечислены последовательно, в порядке обхода по часовой стрелке или против неё. Ограничения: $|x_i|, |y_i| \leqslant 10\,000$, $3 \leqslant N \leqslant 100$.

Формат выходных данных

В первой строке выведите вещественное число — площадь многоугольника. Абсолютная погрешность не должна превышать 10^{-6}.

Примеры

стандартный ввод	*стандартный вывод*
3 0 0 0 1 2 0	1
4 0 0 0 1 1 1 1 0	1

Задача J. Треугольник и точка

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Даны треугольник и точка на плоскости. Определите, лежит ли точка внутри, на границе или вне треугольника.

Формат входных данных

В первых трёх строках заданы координаты вершин треугольника. Каждая из этих строк содержит два целых числа x_i и y_i, разделённых пробелом. В четвёртой строке заданы координаты точки в аналогичном формате. Все координаты не превосходят по модулю 100. Учтите, что вершины треугольника могут лежать на одной прямой и даже совпадать.

Формат выходных данных

Если точка лежит на стороне или в вершине треугольника, выведите «BORDER». Если точка лежит строго внутри треугольника, выведите «INSIDE». Наконец, если точка лежит вне треугольника, выведите «OUTSIDE».

Примеры

стандартный ввод	*стандартный вывод*
0 0 0 2 1 1 0 1	BORDER
0 0 0 2 2 1 1 1	INSIDE
0 0 0 1 1 1 3 2	OUTSIDE

Задача K. Точка в многоугольнике

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

В этой задаче необходимо выяснить, верно ли, что заданная точка находится внутри или на границе заданного многоугольника.

Формат входных данных

В первой строке задано три числа — N ($3 \leqslant N \leqslant 100\,000$) и координаты точки. Далее N строк содержат по два числа каждая — координаты очередной вершины простого многоугольника в порядке обхода по или против часовой стрелки. Все координаты целые и не превосходят $10\,000$ по абсолютной величине.

Формат выходных данных

Выведите «YES», если заданная точка содержится в приведённом многоугольнике или на его границе, и «NO» в противном случае.

Пример

стандартный ввод	*стандартный вывод*
3 0 0 1 0 0 1 1 1	NO

Задача L. Квадрат

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

На уроке геометрии при обсуждении задачи о квадратуре круга семиклассник Иван заявил, что может сделать квадрат из чего угодно. Владимир Алексеевич, учитель геометрии, тут же решил проверить это. Он нарисовал на доске прямоугольную систему координат и отметил две различных точки: A и B. Задача Ивана — отметить ещё две точки C и D так, чтобы четыре отмеченные точки лежали в вершинах какого-то квадрата.

Помогите Ивану выбрать такие точки.

Формат входных данных

Первая строка входных данных содержит два числа x_A и y_A — координаты точки A. Вторая строка содержит два числа x_B и y_B — координаты точки B. Все заданные числа целые и лежат в пределах от 0 до 100. Гарантируется, что заданные точки различны.

Формат выходных данных

Выведите в первой строке два вещественных числа x_C и y_C — координаты точки C. Во второй строке выведите два вещественных числа x_D и y_D — координаты точки D. Числа в строке разделяйте пробелом. Точки A, B, C и D должны лежать в вершинах некоторого квадрата. Если возможных ответов несколько, выведите любой из них.

Пример

стандартный ввод	*стандартный вывод*
1 0 0 1	1 1 0 0

Задача M. Космический зонд

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Космический зонд движется равномерно и прямолинейно по плоской карте. Он начинает движение в точке $(0, 0)$, а через год оказывается в точке $(1, 0)$.

Зонд ищет космические аномалии. Для этого он накапливает энергию, которая равномерно увеличивает радиус обнаружения: через t лет после старта зонд видит все аномалии в круге с радиусом $2t$.

Через сколько лет после старта зонд обнаружит аномалию, которая расположена в точке (x, y)?

Формат входных данных

В первой строке записаны целые числа x и y — координаты аномалии ($|x|, |y| \leqslant 100$). Гарантируется, что координаты не равны одновременно нулю.

Формат выходных данных

В первой строке выведите одно вещественное число t — момент, когда аномалия будет обнаружена. Ответ будет засчитан, если он отличается от ответа жюри не более чем на 10^{-6}.

Примеры

стандартный ввод	*стандартный вывод*
3 0	1.0
-1 4	2.7370341836

Задача N. Космический корабль

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Космический корабль приземлился, и теперь его нужно поставить в ангар. Глядя на космодром сверху, можно упрощённо представить корабль как треугольник на плоскости, а ворота ангара — как отрезок на этой же плоскости. Космический корабль может поворачиваться на любой угол, а также двигаться в любом направлении на любое расстояние.

Выясните, может ли корабль пройти в ангар через предложенные ему ворота. Если корабль в точности помещается в ворота, считается, что он может через них пройти.

Формат входных данных

Каждый тест к этой задаче — это набор вопросов. Каждый вопрос включает в себя положение корабля и концы ворот ангара. На каждый вопрос нужно отвечать независимо от остальных: так, как будто этот вопрос единственный.

В первой строке записано целое число t — количество вопросов ($1 \leqslant t \leqslant 10$). Следующие t строк описывают вопросы. В каждой из этих строк заданы десять целых чисел: сначала координаты x_a, y_a, x_b, y_b, x_c, y_c трёх углов корабля, а затем координаты x_p, y_p, x_q, y_q двух концов ворот. Две координаты каждой точки разделены одним пробелом, а между координатами разных точек по два пробела.

Координаты не превосходят 100 по абсолютной величине. Кроме того, гарантируется, что концы ворот не совпадают, а углы корабля не лежат на одной прямой. Наконец, все x-координаты углов корабля строго отрицательные, а все x-координаты концов ворот строго положительные.

Формат выходных данных

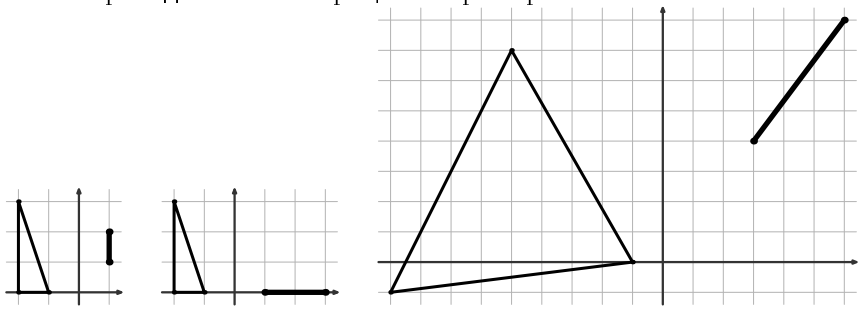
В ответ на каждый вопрос в отдельной строке выведите «YES», если корабль может попасть в ангар, и «NO» в противном случае.

Примеры

стандартный ввод	*стандартный вывод*
1 -1 0 -2 0 -2 3 1 1 1 2	YES
2 -1 0 -2 0 -2 3 1 0 3 0 -1 0 -5 7 -9 -1 3 4 6 8	YES NO

Пояснения к примерам

Ниже приведены иллюстрации к примерам.



Задача O. Два листа бумаги — лайт

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Решите задачу «Два листа бумаги» с ограничениями $1 \leqslant W, H, w, h \leqslant 100$.

Задача P. Два листа бумаги

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

У Саши есть два прямоугольных листа бумаги: синий размера $W \times H$ сантиметров и зелёный размера $w \times h$ сантиметров. Может ли Саша положить их на бесконечный плоский стол так, чтобы было видно только синий лист?

Формально: после того, как листы уложены на плоскость, каждая точка зелёного листа должна находиться внутри или на границе синего листа.

Формат входных данных

В первой строке задано целое число t — количество наборов входных данных ($1 \leqslant t \leqslant 10^4$). Далее для каждого набора:

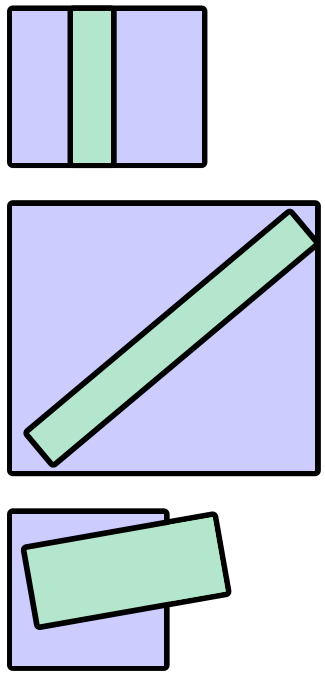
В единственной строке заданы четыре целых числа: W, H, w и h — размеры синего и зелёного листов бумаги ($1 \leqslant W, H, w, h \leqslant 10^9$).

Формат выходных данных

Для каждого набора входных данных:

В отдельной строке выведите «`Yes`», если синим листом можно полностью закрыть зелёный лист, и «`No`» иначе. Каждую букву можно выводить в любом регистре.

Пример

стандартный ввод	*стандартный вывод*	*пояснение*
3 4 5 1 4 7 8 9 1 4 4 2 5	Yes Yes No	

Задача Q. Выпуклая оболочка

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Выпуклая оболочка множества точек на плоскости — это выпуклый многоугольник минимальной площади такой, что все точки множества лежат у него внутри или на границе. Известно, что для любого множества точек выпуклая оболочка единственна с точностью до порядка обхода её вершин.

По данному множеству из n точек постройте его выпуклую оболочку.

Формат входных данных

В первой строке задано число n ($3 \leqslant n \leqslant 100$). Следующие n строк содержат по два числа x_i y_i каждая через пробел — координаты i-й точки ($-1000 \leqslant x_i, y_i \leqslant 1000$). Все заданные числа целые.

Формат выходных данных

В первой строке выведите k — количество вершин выпуклой оболочки. В следующих k строках выведите по два числа x'_k y'_k через пробел — координаты вершин многоугольника, являющегося выпуклой оболочкой, в порядке их обхода по часовой стрелке. Каждый из углов многоугольника должен быть строго больше нуля и строго меньше 180 градусов.

Гарантируется, что заданные точки таковы, что такой многоугольник существует.

Примеры

стандартный ввод	*стандартный вывод*
3 0 0 1 1 0 1	3 0 0 0 1 1 1
6 0 0 0 2 2 0 2 2 2 1 1 1	4 0 2 2 2 2 0 0 0

Задача R. Выпуклая оболочка сверху

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Задана последовательность $y_1, y_2, \ldots, y_n$ из n целых чисел. Каждое её число выбрано случайно, равновероятно из целых чисел от 1 до 10^9 и независимо от других.

Рассмотрим множество точек (i, y_i) на плоскости. Точка принадлежит *верхней выпуклой оболочке* этого множества, если существует прямая, которая проходит через эту точку и при этом строго выше всех остальных точек множества.

Найдите все точки верхней выпуклой оболочки заданного множества.

Формат входных данных

В первой строке записано целое число n — длина последовательности ($1 \leqslant n \leqslant 200\,000$). В следующей строке записаны n целых чисел $y_1, y_2, \ldots, y_n$ — сама последовательность ($1 \leqslant y_i \leqslant 10^9$). Гарантируется, что каждое число выбрано случайно, равновероятно из целых чисел от 1 до 10^9 и независимо от других.

Формат выходных данных

В первой строке выведите целое число k — количество точек, лежащих на верхней выпуклой оболочке множества точек (i, y_i). В следующих k строках выведите сами эти точки — по одной на строке, в порядке возрастания абсциссы.

Пример

стандартный ввод
6 738019188 156680847 182508705 645867035 421237740 324760886
стандартный вывод
3 1 738019188 4 645867035 6 324760886

Задача S. Заповедник в лесу

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

На бесконечной плоскости, посреди огромного леса, расположен заповедник. Территория заповедника огорожена забором. Забор имеет форму выпуклого многоугольника с n вершинами-столбами.

Лесник Фёдор приехал работать в заповедник, и теперь разбирается, с чем ему предстоит работать. У Фёдора есть карта леса. На ней введена декартова система координат. На карте отмечены n столбов забора в порядке обхода против часовой стрелки. Также на карте отмечены все q деревьев в лесу.

Для каждого дерева выясните, находится ли оно внутри заповедника.

Формат входных данных

В первой строке задано целое число t — количество наборов входных данных ($1 \leqslant t \leqslant 33\,333$). Далее для каждого набора:

В первой строке задано целое число n — количество столбов в заборе ($3 \leqslant n \leqslant 100\,000$).

В каждой из следующих n строк задана позиция столба в заборе. Столбы перечислены в порядке обхода забора против часовой стрелки. Многоугольник, задаваемый забором — простой и строго выпуклый: у него нет самопересечений и самокасаний, а каждый из n углов строго меньше развёрнутого.

В следующей строке задано целое число q — количество деревьев в лесу ($3 \leqslant q \leqslant 100\,000$).

В каждой из следующих q строк задана позиция дерева в лесу.

Позиция каждого столба и каждого дерева задаётся двумя целыми числами — координатами x и y ($0 \leqslant x, y \leqslant 10^8$).

Позиции всех деревьев и всех столбов попарно различны. Ни одно дерево не растёт ровно на границе заповедника.

Сумма n во всех наборах входных данных не превосходит $100\,000$.

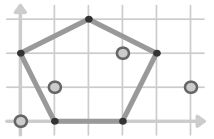
Сумма q во всех наборах входных данных не превосходит $100\,000$.

Формат выходных данных

Для каждого набора входных данных:

Для каждого дерева в отдельной строке выведите «`Yes`», если это дерево находится внутри заповедника, и «`No`» иначе. Каждую букву можно выводить в любом регистре.

Пример

стандартный ввод	*стандартный вывод*	*пояснение*
1 5 1 0 3 0 4 2 2 3 0 2 4 0 0 3 2 1 1 5 1	No Yes Yes No	

Задача T. Ареал

Имя входного файла:	*стандартный ввод*
Имя выходного файла:	*стандартный вывод*
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мебибайт

Специалисты, исследующие дагестанских туров, пришли к выводу, что в течение года каждый отдельно взятый тур мигрирует в пределах некоторой области, которую можно описать как многоугольник Ψ. Летний ареал туров также представляет собой многоугольник Φ. Полный ареал туров теперь можно определить как *сумму Минковского* $\Psi \oplus \Phi$.

Суммой Минковского множеств точек (векторов) Ψ и Φ называется множество точек $c = a + b$, где $a \in \Psi$ и $b \in \Phi$.

Ваша задача — вычислить площадь полного ареала туров.

Формат входных данных

Входные данные состоят из описаний двух выпуклых многоугольников — Ψ и Φ. Описание каждого из них начинается со строки, содержащей число вершин N ($2 < N \leqslant 10^5$). Затем следуют N строк, каждая содержит координаты очередной вершины (x_i, y_i), раздёленные пробелами. Вершины задаются в порядке обхода против часовой стрелки, начиная с любой из них; все координаты целые и не превосходят по модулю 10^8. Последовательные три вершины не могут лежать на одной прямой.

Формат выходных данных

Выведите одно вещественное число с шестью или более верными знаками после десятичной точки — площадь искомого ареала.

Пример

стандартный ввод	*стандартный вывод*
4 0 0 1 0 1 1 0 1 4 0 1 -1 0 0 -1 1 0	7.000000