

Задача А. Принадлежность точки отрезку

Имя входного файла: `point.in`
Имя выходного файла: `point.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Формат входных данных

Шесть чисел — координаты точки и координаты концов отрезка.

Формат выходных данных

Одна строка YES, если точка принадлежит отрезку, и NO в противном случае.

Примеры

<code>point.in</code>	<code>point.out</code>
3 3 1 2 5 4	YES
4 2 4 2 4 5	YES

Задача В. Расстояние от точки до отрезка

Имя входного файла: `distance.in`
Имя выходного файла: `distance.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Найдите расстояние от заданной точки до заданного отрезка.

Формат входных данных

Шесть целых чисел — координаты точки и координаты концов отрезка.

Формат выходных данных

Одно число — расстояние от точки до отрезка с точностью не менее 10^{-6} .

Примеры

<code>distance.in</code>	<code>distance.out</code>
0 4 2 3 2 5	2.000000000000000000000000
4 0 0 0 4 0	0.000000000000000000000000

Задача С. Пусти козла в огород - 5

Имя входного файла: `goat5.in`
Имя выходного файла: `goat5.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В огород к Ивану Петровичу повадился чужой козел по ночам воровать капусту. Чтобы поймать наглеца, Иван Петрович установил на огороде прожектор, освещающий часть плоскости в форме некоторого угла. И когда очередной ночью Иван Петрович услышал хрумканье в своем огороде, он включил свой прожектор. Определите, увидит ли Иван Петрович чужого козла или нет.

Формат входных данных

Программа получает на вход координаты четырех точек A , O , B , P . Прожектор установлен в точке O , точки A и B лежат на границах освещенной прожектором области (на разных лучах), в точке P находится козел. Все числа во входном файле целые.

Формат выходных данных

Выведите слово «YES», если Иван Петрович увидит козла или слово «NO» в противном случае

Примеры

goat5.in	goat5.out
0 1 0 0 1 0 1 1	YES
1 0 0 0 0 1 -1 -1	NO

Задача D. Расстояние между отрезками

Имя входного файла: `distance.in`
Имя выходного файла: `distance.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Найдите расстояние между двумя отрезками.

Формат входных данных

Восемь целых чисел — координаты четырёх концов двух отрезков.

Формат выходных данных

Одно число — расстояние между отрезками с точностью не менее 10^{-6} .

Примеры

<code>distance.in</code>	<code>distance.out</code>
1 1 2 2 2 1 3 0	0.707106781
1 1 2 2 1 2 2 1	0.000000000

Задача Е. Расстояние от точки до прямой

Имя входного файла: `distance.in`
Имя выходного файла: `distance.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Найдите расстояние от заданной точки до заданной прямой.

Формат входных данных

Пять целых чисел — координаты точки и коэффициенты A , B и C нормального уравнения прямой.

Формат выходных данных

Одно число — расстояние от точки до прямой с точностью не менее 10^{-6} .

Примеры

<code>distance.in</code>	<code>distance.out</code>
1 1 1 1 -1	0.7071067812
1 5 -4 0 8	1.0000000000

Задача F. Пусти козла в огород -7

Имя входного файла: `goat7.in`
Имя выходного файла: `goat7.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Петр Васильевич всегда выгуливает своих козлов у себя в огороде, но ему не нравится, что они вытаптывают его грядки, поэтому он решил усовершенствовать привязь. Теперь ошейник у них прикреплен к веревке, натянутой между двумя деревьями, поэтому козлы могут ходить только по отрезку, их соединяющему.

Но козлам скучно ходить по огороду по одиночке, поэтому они хотят встретиться и пожевать траву вместе. Помогите им встретиться!

Формат входных данных

Программа получает на вход восемь целых чисел, по абсолютной величине не превосходящих 10^4 . Сначала указываются координаты двух деревьев, к которым привязан первый козел, а затем - координаты деревьев, к которым привязан второй. Заметим, что для привязи могут служить одни и те же деревья (в том числе могут совпадать и концы одной привязи).

Формат выходных данных

Если козлам встретиться не суждено, выведите строку «Empty». Если они могут встретиться только в одной точке, то выведите два числа — координаты точки пересечения. Если пересечением является отрезок, то выведите четыре числа — координаты двух концов отрезка в лексикографическом порядке (то есть сначала нужно вывести ту точку, у которой меньше координата x , а если у них равны координаты x то ту, у которой меньше координата y). Все числа следует выводить с точностью не менее 6 знаков после запятой.

Примеры

goat7.in	goat7.out
0 0 9 9 9 5 0 5	5.0000000000 5.0000000000
0 0 9 9 15 15 7 7	7.0000000000 7.0000000000 9.0000000000 9.0000000000
0 0 9 9 10 10 10 10	Empty

Задача G. Вписанная окружность

Имя входного файла: `incircle.in`
Имя выходного файла: `incircle.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Треугольник задан координатами своих вершин. Найдите центр вписанной в него окружности.

Формат входных данных

Шесть чисел — координаты вершин треугольника.

Формат выходных данных

Координаты центра вписанной в данный треугольник окружности.

Примеры

<code>incircle.in</code>	<code>incircle.out</code>
0 0 0 15 20 0	5 5
1 0 0 1 0 0	0.29289321881345248277 0.29289321881345248277

Задача Н. Площадь пересечения двух окружностей

Имя входного файла: `area.in`
Имя выходного файла: `area.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Требуется найти площадь пересечения двух окружностей с точностью **не менее** 10^{-6} .

Формат входных данных

В единственной строке заданы 6 целых чисел $x_1, y_1, r_1, x_2, y_2, r_2$ — координаты центров и радиусы окружностей соответственно.

Координаты неотрицательны и по абсолютному значению не превышают 1 000.

Формат выходных данных

Требуется вывести единственное неотрицательное число — площадь пересечения двух окружностей.

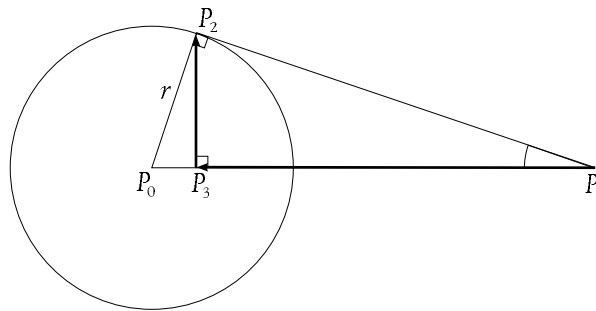
Примеры

<code>area.in</code>	<code>area.out</code>
20.0 30.0 15.0 40.0 30.0 30.0	608.365932190890130000000
100 100 50 200 100 10	0.00000000000000000000000

Задача I. Касательные к окружности

Имя входного файла: `tangent.in`
Имя выходного файла: `tangent.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам задана окружность и точка. Точка может лежать вне окружности, на ее границе, а также внутри окружности. Необходимо провести касательные к окружности (если это возможно) и найти точки касания.



Формат входных данных

В первой строке входного файла находятся два целых числа — координаты центра окружности P_0 . Во второй строке записано целое число r — радиус окружности. В третьей строке находятся два целых числа — координаты точки P_1 .

Формат выходных данных

В первой строке выходного файла выведите количество точек касания (0, 1 или 2). В случае одной точки касания выведите во второй строке координаты этой точки. В случае двух точек касания выведите во второй строке координаты точки P_3 , в третьей длины векторов $\overrightarrow{P_1P_3}$ и $\overrightarrow{P_3P_2}$, в следующих двух строках должны находиться координаты точек касания.

Абсолютная или относительная погрешность чисел в выходных данных не должна превышать 10^{-7} .

Пример

tangent.in	tangent.out
2 2	2
2	2.0000000000 3.3333333333
2 5	1.6666666667 1.4907119850
	0.5092880150 3.3333333333
	3.4907119850 3.3333333333

Задача J. Точка в многоугольнике

Имя входного файла: `point.in`
Имя выходного файла: `point.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Формат входных данных

В первой строке три числа — N ($3 \leq N \leq 100\,000$) и координаты точки. Далее в N строках по паре чисел — координаты очередной вершины простого многоугольника в порядке обхода по или против часовой стрелки.

Формат выходных данных

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

Пример

<code>point.in</code>	<code>point.out</code>
3 0 0 1 0 0 1 1 1	NO

Задача К. Площадь многоугольника

Имя входного файла: `area.in`
Имя выходного файла: `area.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Формат входных данных

В первой строке одно число N ($3 \leq N \leq 100\,000$). Далее в N строках по паре чисел — координаты очередной вершины простого многоугольника в порядке обхода по или против часовой стрелки.

Все координаты — целые числа, по модулю не превосходящие 10^4 .

Формат выходных данных

Одно число — величина площади приведённого многоугольника.

Пример

area.in	area.out
3 1 0 0 1 1 1	0.5

Задача L. Выпуклый многоугольник

Имя входного файла: `polygon.in`
Имя выходного файла: `polygon.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Формат входных данных

В первой строке одно число N ($3 \leq N \leq 100\,000$). Далее в N строках по паре целых чисел — координаты очередной вершины простого многоугольника в порядке обхода по или против часовой стрелки.

Формат выходных данных

Одна строка «YES», если приведённый многоугольник является выпуклым, и «NO» в противном случае.

Пример

<code>polygon.in</code>	<code>polygon.out</code>
3 0 0 0 1 1 0	YES