

Задача А. Принадлежность точки отрезку

Имя входного файла: point3.in
Имя выходного файла: point3.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Формат входных данных

Шесть целых чисел от -10^4 до 10^4 — координаты точки и координаты концов отрезка.

Формат выходных данных

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

Примеры

point3.in	point3.out
3 3 1 2 5 4	YES

Задача В. Расстояние от точки до прямой

Имя входного файла: distance1.in
Имя выходного файла: distance1.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

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

distance1.in	distance1.out
1 1 1 1 -1	0.7071067812

Задача С. Река

Имя входного файла: `river.in`
Имя выходного файла: `river.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Каждый день в Монастыре Светлой Луны проходит утренний обряд, являющийся важнейшим элементом жизненного пути каждого монаха. Во время этого обряда один из монахов должен выйти из Монастыря, пройти к реке Мацанг (истоку Брахмапутры) и, набрав там воды, принести ее в Храм Солнца Дхармы, расположенный неподалеку от Монастыря. При этом обряд должен быть закончен ровно к рассвету. Для того чтобы достичь такой точности, монахам необходимо знать длину кратчайшего пути от Монастыря до Храма, проходящего через реку Мацанг. Вы призваны помочь Монастырю Светлой Луны рассчитать эту самую длину. Поскольку река близ Монастыря не имеет сколько-нибудь значительных изгибов, вы можете считать ее идеально прямой. Кроме того, можно пренебречь ее шириной. Однако необходимо помнить, что Мацанг — река горная и опасная, а поэтому перейти ее вброд абсолютно невозможно! При этом, естественно, ни Монастырь, ни Храм не находятся в реке.

Формат входных данных

В первой строке записаны координаты Монастыря, во второй — координаты Храма. В третьей строке записаны числа k, b — коэффициенты, задающие уравнение реки ($y = kx + b$). Все числа кроме k — целые и не превосходят по модулю 10000. Коэффициент k — действительное число, не превосходящее по модулю 100.

Формат выходных данных

Выведите одно действительное число, округленное до трех знаков после запятой — длину кратчайшего пути. В том случае, если Монах не может пройти к Храму, необходимо вывести "No solution."

Примеры

<code>river.in</code>	<code>river.out</code>
0 10 20 10 0 0	28.284
0 10 20 -10 0 0	No solution.

Задача D. Пусти козла в огород - 1

Имя входного файла: `goat1.in`
Имя выходного файла: `goat1.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Однажды на огород к Ивану Петровичу забежало целых три козла. Известно, что козлы весьма агрессивные животные, особенно когда речь идет о борьбе за вкусную капусту. Поэтому каждый из трех козлов, заметив других козлов, замер на месте и начал наблюдать за оставшимися козлами: одним глазом за одним козлом, другим — за другим. Естественно, для этого козлу нужно “косить” глазами.

Определите наибольший угол, на который пришлось “раскосить” глазами козлам.

Формат входных данных

Программа получает на вход координаты трех точки, в которых стоят козлы (сначала координаты первого козла, затем второго, затем третьего). Координаты — пара целых чисел, не превосходящих 10^4 по модулю.

Формат выходных данных

Выведите единственное число — величину самого большого угла, на который “косят” глаза козлов с точностью не меньше 6 знаков после запятой.

Примеры

<code>goat1.in</code>	<code>goat1.out</code>
0 0	90.000000
3 0	
0 4	

Задача Е. Пусти козла в огород - 3

Имя входного файла: `goat3.in`
Имя выходного файла: `goat3.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Козла пустили в огород. К несчастью для козла, его крепко-накрепко привязали к точке C крепчайшей альпинистской верёвкой длины L (то есть козёл может кушать травку, удалённую от точки C не более чем на L метров). И только одно скрасило жизнь бедного козла — недалеко от него виднеется грядка так любимых им ананасов, представляющая собой отрезок прямой с концами в точках A и B . Но, при большом старании, козёл может растянуть крепчайшую альпийскую верёвку. Определите, на какую длину козлу придётся растянуть верёвку, чтобы добраться хотя бы до одного ананасика? А чтобы съесть все ананасы?

Формат входных данных

В первой строке входного файла содержатся координаты точек A и B , разделённые пробелами. На второй строке записаны координаты точки C и длина верёвки L , разделённые пробелами. Все числа целые, $L \geq 0$, все координаты не превосходят по модулю 10 000.

Формат выходных данных

В первой строке вывести минимальную длину, на которую козлу придётся растянуть верёвку, чтобы дотянуться до грядки с ананасами. Во второй строке вывести минимальную длину, на которую козлу придётся растянуть верёвку, чтобы добраться до всех ананасов с грядки. Все числа выводить с точностью не менее 6 знаков после запятой.

Примеры

<code>goat3.in</code>	<code>goat3.out</code>
8 -6 8 6	1.0000000000
0 0 7	3.0000000000

Задача F. Полярный угол

Имя входного файла: `angle1.in`
Имя выходного файла: `angle1.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Формат входных данных

Два числа — координаты точки ($|x|, |y| \leq 10^4$), не совпадающей с началом координат.

Формат выходных данных

Одно число — величина её полярного угла в радианах из интервала $[0, 2\pi)$. Все числа следует выводить с точностью не менее 6 знаков после запятой.

Примеры

<code>angle1.in</code>	<code>angle1.out</code>
2 3	0.98279372324732906796

Задача G. Векторы

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

Даны два ненулевых вектора. Требуется вычислить:

- Длину первого и второго вектора (два числа)
- Вектор, образованный сложением данных двух векторов
- Скалярное и векторное произведения данных векторов
- Площадь треугольника, построенного из этих векторов

Формат входных данных

В двух строках входного файла заданы по четыре целых числа, не превосходящих по модулю 10 000, — координаты начала и конца первого вектора, затем второго.

Формат выходных данных

В каждой строке выходного файла — ответ на соответствующий пункт задачи с точностью не менее 10^{-6} .

Примеры

<code>vectors.in</code>	<code>vectors.out</code>
5 1 2 6	5.830951895 9.219544457
1 1 7 8	3.000000000 12.000000000
	17.000000000 -51.000000000
	25.500000000

Задача Н. Расстояние между отрезками

Имя входного файла: distance2.in
Имя выходного файла: distance2.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Найдите расстояние между двумя отрезками.

Формат входных данных

В двух строках входного файла даны по четыре целых числа, не превосходящих по модулю 10 000, — координаты концов сначала первого, затем второго отрезков.

Формат выходных данных

В первой строке выходного файла выведите одно вещественное число — расстояние между отрезками с точностью не менее 10^{-6} .

Примеры

distance2.in	distance2.out
1 1 2 2 2 1 3 0	0.707106781

Задача I. Биссектриса

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

Найдите прямую, содержащую биссектрису угла, заданного вершиной X и двумя точками Y и Z на его сторонах.

Формат входных данных

Шесть целых чисел, не превышающих по модулю 10^4 — координаты точек X , Y и Z .

Формат выходных данных

Три числа — коэффициенты нормального уравнения биссектрисы угла $\angle YXZ$ с точностью до шести знаков после запятой.

Примеры

<code>bisector.in</code>	<code>bisector.out</code>
1 1 1 0 0 1	-1.0 1.0 -0.0

Задача J. Пусти козла в огород - 4

Имя входного файла: `goat4.in`
Имя выходного файла: `goat4.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Огород у Петра Васильевича имеет форму треугольника и при этом не огорожен. Петр Васильевич хочет привязать козла к колышку в огороде так, чтобы он мог пастись на участке огорода максимальной площади, но при этом не заходил бы на участки соседей. Определите, в каком месте огорода нужно привязать козла.

Формат входных данных

Программа получает на вход координаты вершин трёх углов огорода. Координаты — пара целых чисел, не превосходящих 10^4 по модулю.

Формат выходных данных

Выведите два числа — координаты колышка, к которому нужно привязать козла, с точностью не менее 6 знаков после запятой.

Примеры

<code>goat4.in</code>	<code>goat4.out</code>
0 0 3 0 0 4	1.00000000 1.00000000

Задача К. Пусти козла в огород - 5

Имя входного файла: `goat5.in`
Имя выходного файла: `goat5.out`
Ограничение по времени: 1 секунда
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 хрумку 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 и не превышают 100 по абсолютному значению. Точки A , O и B не лежат на одной прямой.

Формат выходных данных

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

Примеры

<code>goat5.in</code>	<code>goat5.out</code>
0 1 0 0 1 0 1 1	YES
1 0 0 0 0 1 -1 -1	NO

Задача L. Двойное опрокидывание

Имя входного файла: `intersec2.in`
Имя выходного файла: `intersec2.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

У Кирилла есть две ёлочки и два зловердных котика. Пушок никак не теряет надежды на опрокидывание одной из ёлочек, а Барсик облюбавал вторую ёлочку. Котики приготовились к прыжку. Если траектории прыжков котиков (отрезки с концами в котике и ёлочке, которую каждый из котиков облюбавал) пересекаются, то в прыжке котики могут столкнуться. Котики очень осторожные и не хотят сталкиваться, поэтому не будут прыгать, если траектории пересекаются, и обе ёлки останутся невредимыми. Помогите Пушку и Барсику понять, пересекаются ли траектории их прыжков.

Формат входных данных

В двух строках входного файла заданы по четыре целых числа, не превосходящих по модулю 10 000, — координаты Пушка и его любимой ёлочки, затем координаты Барсика и его ёлочки.

Формат выходных данных

В первой строке выходного файла выведите «YES», если отрезки имеют общие точки, и «NO» в противном случае.

Примеры

<code>intersec2.in</code>	<code>intersec2.out</code>
5 1 2 6 1 1 7 8	YES

Задача М. Пересечение отрезков

Имя входного файла: `segments.in`
Имя выходного файла: `segments.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Даны два отрезка: AB и CD . Определите, какое множество точек является пересечением этих отрезков.

Формат входных данных

Программа получает на вход восемь целых чисел, по абсолютной величине не превосходящих 10^4 — координаты точек A , B , C , D . Точки могут совпадать (в том числе могут совпадать и концы одного отрезка).

Формат выходных данных

Если указанные отрезки не пересекаются, то выведите строку «Empty». Если отрезки пересекаются 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

<code>segments.in</code>	<code>segments.out</code>
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 N. 16

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

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. Точки A и B не совпадают. Точки C и D не совпадают.

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.

Примеры

<code>hard-geometry.in</code>	<code>hard-geometry.out</code>
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