

Задача А. Расстояние

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

На плоскости заданы точка и прямая. Требуется найти расстояние между ними.
Прямая задана уравнением $Ax + By + C = 0$. Точка задана координатами $(x_0; y_0)$.

Формат входного файла

Входной файл содержит несколько (до 1000) тестов. Каждый тест записан в отдельной строке и содержит 5 чисел A, B, C, x_0, y_0 разделённых пробелами. Все числа целые и не превосходят по модулю 10000. Окончание тестов — "0 0 0" на отдельной строке.

Формат выходного файла

Для каждого теста в выходной файл одно число — расстояние с точностью до 3-х знаков после точки.

Примеры

dist.in	dist.out
0 1 2 2 3	5.000
4 0 0 6 4	6.000
2 1 3 7 6	10.286
0 0 0	

Задача В. Диаметр множества

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

На плоскости задано N точек, выбрать две максимально удалённые друг от друга точки и вывести расстояние между ними.

Формат входного файла

Первая строка входного файла содержит N — число точек ($2 \leq N \leq 10000$). Далее следуют N строк - координаты точек. $i + 1$ -ая строка содержит x_i y_i - координаты i -той точки, по абсолютному значению не превосходящие 10^4 .

Формат выходного файла

Выведите в выходной файл одно число D_{max} - максимальное расстояние между точками, с точностью до 2-х знаков.

Примеры

diam.in	diam.out
3 0 0 3 0 0 4	5.00

Задача С. Две окружности

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

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

Окружность можно задать уравнением:

$$(x - x_0)^2 + (y - y_0)^2 = R^2$$

Тогда система уравнений для поиска точек пересечения 2-х окружностей:

$$\begin{cases} (x - x_a)^2 + (y - y_a)^2 = R_a^2, \\ (x - x_b)^2 + (y - y_b)^2 = R_b^2, \end{cases}$$

Для упрощения вычислений перенесем начало координат в центр второй окружности (вычтем из всех x -координат x_b , а y - y_b). Потом, перед выводом ответа, добавим обратно x_b и y_b . Тогда уравнения упростятся:

$$\begin{cases} (x - a)^2 + (y - b)^2 = R_a^2, \\ x^2 + y^2 = R_b^2, \end{cases}$$

$a = x_b - x_a$, а $b = y_b - y_a$, координаты центра первой окружности в новой системе координат.

Вычтем из первого уравнения второе, получим: $-2ax + a^2 - 2by + b^2 = R_a^2 - R_b^2$

Выразим теперь $y = \frac{R_a^2 - R_b^2 + 2ax - a^2 - b^2}{-2b}$

Для упрощения введем переменную: $L = R_a^2 - R_b^2 - a^2 - b^2$

И подставим в 2-ое уравнение системы:

$$x^2 + \frac{(L + 2ax)^2}{4b^2} = R_b^2$$

Домножим на $4b^2$ и раскроем скобки:

$$4b^2x^2 + L^2 + 4Lax + 4a^2x^2 = 4R_b^2b^2$$

$$4(a^2 + b^2)x^2 + 4Lax + (L^2 - 4R_b^2b^2) = 0$$

Формат входного файла

Первая строка входного файла содержит число наборов входных данных K ($1 \leq K \leq 10000$). Каждый набор состоит из двух строчек, каждая из которых

описывает окружность. Описание окружности задается в виде трех чисел x, y, r ($-100 \leq x, y \leq 100, 0 < r \leq 100$), разделенным пробелами. Все числа во входном файле целые.

Формат выходного файла

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 двенадцатью точными знаками после запятой.
- “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.

Примеры

circles.in	circles.out
2 0 0 2 4 0 2 0 0 1 100 100 1	There are only 1 of them.... 2.000000000000 0.000000000000 There are no points!!!

Задача D. Ограда сада

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

Великий и мудрый (но очень жадный) правитель Хаттаб решил оградить свой сад от непрошенных гостей. Он поручил разработать проект постройки ограды величайшему архитектору Тузику. Всё было бы хорошо, но в последнюю ночь перед праздником правителю приснился вещий сон, в котором к нему

спустился Аллах и сказал, что ограда, которую он собирается строить должна быть окружностью (если смотреть на сад сверху), только в этом случае он обретёт покой в своём саду. Теперь он хочет именно такую ограду, и при этом желает потратить минимум ресурсов на её возведение.

Величайший архитектор поставлен в тупик, помогите ему найти центр и радиус окружности (ограды для сада великого правителя). Для простоты вычислений Тузик принимает следующую модель: деревья в саду = точки на плоскости, заданные своими координатами, искомая ограда - окружность минимального радиуса, ограждает все имеющиеся деревья. Дерево считается ограждённым, если оно лежит внутри или на границе окружности, представляющей ограду. Высота ограды устанавливается правителем, на основе "маркетингового" исследования и в данном случае никакого значения не имеет.

Формат входного файла

Первая строка входного файла содержит натуральное число N — количество деревьев в саду ($2 \leq N \leq 100$), далее следует N строк, i -ая строка содержит пару целых чисел разделённых одним пробелом $x_i y_i$ - координаты i -того дерева. Координаты всех деревьев различны и по абсолютному значению не превышают 10^4 .

Формат выходного файла

В первой строке выведите радиус искомой ограды, во второй координату X центра ограды, в третьей - координату Y центра ограды. Все числа достаточно вывести с точностью до пятого знака.

Примеры

garden.in	garden.out
13	10.00000
0 0	0.00000
-2 3	0.00000
-5 3	
-10 0	
1 1	
10 0	
-1 -1	
7 1	
0 -10	
-5 -5	
0 10	
4 5	
5 5	

Задача Е. Скалярное произведение

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

Скалярное произведение - операция над двумя векторами, результатом которой является скаляр (число), не зависящее от системы координат и характеризующее длины векторов-сомножителей и угол между ними.

Обычно используется одно из следующих обозначений: $\langle \mathbf{a}, \mathbf{b} \rangle$, (\mathbf{a}, \mathbf{b}) , $\mathbf{a} \cdot \mathbf{b}$, или (обозначение Дирака, часто применяемое в квантовой механике для векторов состояния): $\langle a|b \rangle$.

В трёхмерном евклидовом пространстве $\vec{x} = (x_1, x_2, x_3)$ скалярное произведение $\langle \vec{x}, \vec{y} \rangle = x_1 y_1 + x_2 y_2 + x_3 y_3$. Аналогичное утверждение верно для евклидова пространства любой размерности (в сумму тогда входит количество членов, равное размерности пространства).

Нужно посчитать скалярное произведение 2-х векторов в N -мерном пространстве.

Формат входного файла

В первой строке входного файла задано число N ($1 \leq N \leq 100$) — размерность пространства.

Следующие две строки содержат по N чисел (x_1, x_2, \dots, x_N) — вектора. Координаты - вещественные числа заданные с точностью до 10^{-4} .

Формат выходного файла

В первой строке входного файла выведите вещественное число — скалярное произведение с точностью до 3-х знаков после запятой.

Примеры

scalar.in	scalar.out
3	78.000
2 3 9	
3 6 6	
2	0.000
2.1 3.3	
-6.6 4.2	

Задача F. Тангенс

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

Посчитайте тангенс угла заданного в градусах.

Формат входного файла

Первая строка содержит количество наборов исходных данных T ($0 \leq T \leq 100$). Далее T строк — углы в градусах a (целое число) ($0 \leq a \leq 10000$).

Формат выходного файла

Для каждого из углов необходимо вывести значение тангенса с 3-мя знаками после запятой.

Примеры

tang.in	tang.out
6	0.000
0	0.577
30	1.000
45	1.732
60	57.290
89	0.017
361	

Задача G. Параллельность

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

На плоскости заданы две прямых. Определить, параллельны ли прямые?

Прямые заданы коэффициентами уравнения $Ax + By + C = 0$.

Формат входного файла

Входной файл содержит несколько (до 1000) тестов. Каждый тест занимает 2 строки, в каждой по 3 числа: A_1, B_1, C_1 , и A_2, B_2, C_2 , Все числа целые и не превосходят по модулю 10^4 . Окончание тестов — “0 0 0” на отдельной строке.

Формат выходного файла

Для каждого теста выведите “YES” если прямые параллельны или совпадают, и “NO” в противном случае.

Примеры

par.in	par.out
0 1 5	YES
0 2 6	NO
1 2 3	YES
-1 2 6	
1 2 3	
2 4 0	
0 0 0	