

Задача А. Не курить!

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

Вася — хороший парень. Но у него есть плохая привычка — он курит. Все то время, сколько Петя дружит с Васей, он пытается отучить его от этого. Но ему это так и не удалось, потому что Вася не хочет бросать курить.

Недавно Петя придумал способ, как отучить своего друга от курения. Вася — неряха, поэтому его сигареты не лежат в пачке, а разбросаны по огромному столу. Петя хочет брать несколько сигарет в день незаметно для Васи. Вася не заметит пропажи сигарет, если в день будет пропадать не более одной сигареты. Кроме того, Петя должен брать только ту сигарету, которая пересекается с какой-нибудь другой сигаретой на столе. Помогите Пете узнать, сможет ли он начать реализацию своего плана.

Формат входного файла

Сигарета представляется как отрезок прямой. В первой строке входного файла записано число N ($1 \leq N \leq 125\,000$) — количество сигарет на Васином столе. Следующие N строк содержат описания сигарет: $(i + 1)$ -я строка содержит координаты концов i -й сигареты — целые числа x_1, y_1, x_2, y_2 ($-10\,000 \leq x_1, y_1, x_2, y_2 \leq 10\,000$).

Формат выходного файла

В первой строке выходного файла выведите слово “YES”, если Пете удастся начать реализацию своего плана. Вторая строка должна содержать числа i и j : i — номер сигареты, которую должен взять Петя, j — номер сигареты, с которой она пересекается.

Если Петя не сможет взять ни одной сигареты, выведите в единственной строке выходного файла “NO”.

Примеры

smoking.in	smoking.out
2 0 0 2 2 0 2 2 0	YES 2 1
1 5 5 10 97	NO
4 0 0 3 1 2 0 3 -2 4 -1 7 2 -1 2 6 2	NO

Задача В. Футбол

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

На футбольном поле размером $x \times y$ находятся n футболистов. Они уже очень устали и стоят на месте, но ждут, куда упадет мяч, чтобы побежать к нему. Футболист бежит к мячу в том случае, если мяч упал к этому футболисту ближе, чем к любому другому футболисту. Требуется определить для каждого футболиста границы зоны, при попадании в которую он побежит к мячу, если известно, что она представляет собой многоугольник.

Формат входного файла

В первой строке входного файла заданы три целых числа x, y и n ($2 \leq x, y \leq 10^5$, $1 \leq n \leq 1000$). Следующие n строк содержат целые координаты футболистов x_i, y_i ($0 < x_i < x, 0 < y_i < y$). Никакие два футболиста не стоят в одной точке.

Формат выходного файла

В выходной файл выведите n строк. В каждой из строк первое число — количество вершин зоны k_i , далее k_i чисел — координаты вершин x_{ij}, y_{ij} в порядке обхода против часовой стрелки, начиная с самой нижней из самых левых вершин зоны. Вещественные числа выводите с максимальной точностью.

Примеры

football.in	football.out
4 4 4	4 0 0 2 0 2 2 0 2
1 1	4 0 2 2 2 2 4 0 4
1 3	4 2 0 4 0 4 2 2 2
3 1	4 2 2 4 2 4 4 2 4
3 3	

Задача С. Футбол 2

Имя входного файла:	football2.in
Имя выходного файла:	football2.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

На футбольном поле размером $x \times y$ находятся n футболистов. Они уже очень устали и стоят на месте, но ждут, куда упадет мяч, чтобы побежать к нему. Футболист бежит к мячу в том случае, если мяч упал к этому футболисту ближе, чем к любому другому футболисту. Требуется определить для каждого футболиста границы зоны, при попадании в которую он побежит к мячу, если известно, что она представляет собой многоугольник.

Формат входного файла

В первой строке входного файла заданы три целых числа x, y и n ($2 \leq x, y \leq 10^5$,

$1 \leq n \leq 30\,000$). Следующие n строк содержат целые координаты футболистов $x_i y_i$ ($0 < x_i < x$, $0 < y_i < y$). Никакие два футболиста не стоят в одной точке.

Формат выходного файла

В выходной файл выведите n строк. В каждой из строк первое число — количество вершин зоны k_i , далее k_i чисел — координаты вершин $x_{ij} y_{ij}$ в порядке обхода против часовой стрелки, начиная с самой нижней из самых левых вершин зоны. Вещественные числа выводите с максимальной точностью.

Примеры

football2.in	football2.out
4 4 4	4 0 0 2 0 2 2 0 2
1 1	4 0 2 2 2 2 4 0 4
1 3	4 2 0 4 0 4 2 2 2
3 1	4 2 2 4 2 4 4 2 4
3 3	

Задача D. Кто ходит в гости по утрам

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

В Чудесном Лесу живут N различных персонажей, у каждого из которых есть свой собственный до-мик. Следуя заветам одного из самых известных лесных персонажей, Винни-Пуха, каждый житель считает необходимым проснуться с утра пораньше, умыться, одеться и пойти в гости к кому-либо. Разумеется, что-бы поступить не просто мудро, а очень мудро и не потратить слишком много времени на дорогу, персонаж отправится не к кому-нибудь, а к своему соседу, то есть к тому из жителей, домик которого находится к данному персонажу на наименьшем возможном расстоянии. Нетрудно понять, что хозяина этого домика не окажется дома, поскольку он тоже воспользуется правилом Винни-Пуха. Лишь по этой причине некому будет ни крикнуть “Ура!”, ни обрадоваться гостям. Если вдруг окажется, что несколько домиков расположены на минимальном расстоянии от персонажа, то он выберет для похода в гости домик с наименьшим номером. Ваша задача — определить какие персонажи соберутся у каждого домика.

Формат входного файла

В первой строке задается количество персонажей N ($2 \leq N \leq 100\,000$). В каждой из последующих N строк задаются по два числа — координаты точки на плоскости, в которой расположен домик соответствующего персонажа. Все координаты — целые неотрицательные числа, не превосходящие 10^9 .

Формат выходного файла

Выведите N строк. i -ая строка должна содержать число i , за которым следует двоеточие и далее в порядке возрастания номера персонажей, которые придут в гости в i -ый домик.

Примеры

nearest.in	nearest.out
6	1: 2 3
0 0	2: 1
1 0	3:
0 1	4: 5
3 3	5: 4 6
2 2	6:
3 1	

Задача E. Объединение кругов

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

Найдите площадь объединения кругов.

Формат входного файла

В первой строке записано количество тестов K ($1 \leq K \leq 10$).

Первая строка каждого теста — количество окружностей $1 \leq N \leq 100$. Следующий N строк содержат координаты центра и радиус каждой окружности $x_i y_i r_i$. Все координаты — целые числа, которые по модулю не превосходят 1000. Радиус окружности положительное целое число не больше 1000.

Формат выходного файла

Для каждого теста выведите площадь объединения окружностей с 6 знаками после запятой.

Примеры

circles.in	circles.out
1	40.84070449666731
3	
0 0 3	
1 0 1	
6 0 2	

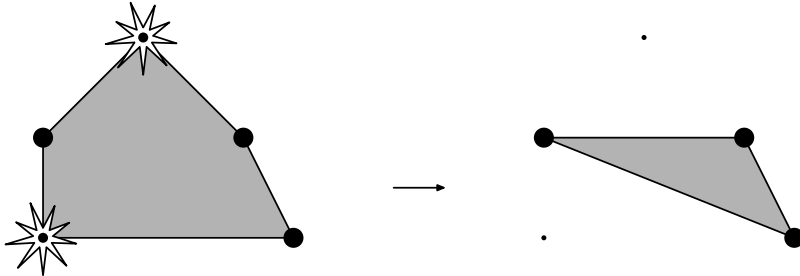
Задача F. Jungle Outpost

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

There is a military base lost deep in the jungle. It is surrounded by n watchtowers with ultrasonic generators. In this problem watchtowers are represented by points on a plane.

Watchtowers generate ultrasonic field and protect all objects that are strictly inside the towers' convex hull. There is no tower strictly inside the convex hull and no three towers are on a straight line.

The enemy can blow up some towers. If this happens, the protected area is reduced to a convex hull of the remaining towers.



The base commander wants to build headquarters inside the protected area. In order to increase its security, he wants to maximize the number of towers that the enemy needs to blow up to make the headquarters unprotected.

Формат входного файла

The first line of the input file contains a single integer n ($3 \leq n \leq 50\,000$) — the number of watchtowers. The next n lines of the input file contain the Cartesian coordinates of watchtowers, one pair of coordinates per line. Coordinates are integer and do not exceed 10^6 by absolute value. Towers are listed in the order of traversal of their convex hull in clockwise direction.

Формат выходного файла

Write to the output file the number of watchtowers the enemy has to blow up to compromise headquarters protection if the headquarters are placed optimally.

Примеры

jungle.in	jungle.out
3 0 0 50 50 60 10	1
5 0 0 0 10 10 20 20 10 25 0	2

Задача G. Триатлон

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

Триатлон — атлетическое соревнование, состоящее из трех последовательных этапов, которые суммарно должны быть пройдены как можно быстрее. Первый этап — плавание, второй — велогонка, а третий — бег.

Известна скорость каждого участника для каждого этапа. Жюри соревнования может выбрать длину каждого этапа любым образом так, чтобы каждый этап не был нулевой длины. В результате, иногда жюри может выбрать длины этапов так, чтобы определенный участник стал победителем соревнования.

Формат входного файла

В первой строке входного файла задано целое число N — количество участников соревнования ($1 \leq N \leq 100$). Следующие N строк содержат по 3 целых числа V, U и W ($1 \leq U, V, W \leq 10000$), определяющие скорость очередного участника на этапах соревнования.

Формат выходного файла

Для каждого участника выведите одну строку, содержащую слово *Yes* в случае если жюри может подобрать длины этапов так, чтобы этот участник стал победителем (т.е. этот участник — единственный, кто первый дошел до финиша), или слово *No* в противном случае.

Примеры

tri.in	tri.out
9	Yes
10 2 6	Yes
10 7 3	Yes
5 6 7	No
3 2 7	No
6 2 6	No
3 5 7	Yes
8 4 6	No
10 4 2	Yes
1 8 7	

Задача Н. Место встречи изменить нельзя

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

Даны N точек. Найдите 2 из них, такие, что расстояние между ними минимально.

Формат входного файла

Первая строка входного файла содержит целое число N ($2 \leq N \leq 100\,000$) — количество точек. Каждая из следующих N строк содержит пару целых чисел X и Y , разделённых пробелом, — координаты ($-1\,000\,000\,000 \leq X, Y \leq 1\,000\,000\,000$). Все точки различны.

Формат выходного файла

Единственная строка выходного файла должна содержать координаты двух выбранных точек.

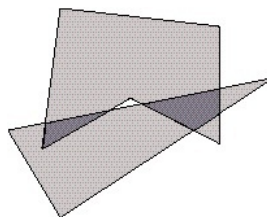
Пример

rendezvous.in	rendezvous.out
4	0 0
0 0	0 1
0 1	
1 1	
1 0	

Задача I. Пересечение невыпуклых многоугольников

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

Most drawing or illustration programs have simple tools for creating polygon objects. The better ones can find the regions that are the intersections of two polygons. The picture below shows two polygons, one is a pentagon and the other is a triangle. Their intersection consists of the two dark regions.



IBM has just hired you as a member of a programming team that will create a very sophisticated drawing/illustration program. Your task is to write the part of the program that deals with polygon intersections. Your boss has told you to delay work on the user interface and focus only on the geometric representations of the intersections.

A polygon in the Cartesian plane can be represented by a sequence of points that are its vertices. The vertices in the sequence appear in the order in which they are visited when traveling clockwise around the polygon's boundary; so any two adjacent vertices in the sequence are the endpoints of a line segment that is one of the polygon's sides. The last and the first vertices in the sequence are also endpoints of a side. Vertices are identified by their x - and y coordinates. Assume the following about each polygon:

- No point will occur as a vertex (on the same polygon) more than once.
- Two sides can intersect only at a common endpoint (vertex).
- The angle between any two sides with a common vertex has a measure that is greater than 0 and less than 360.
- The polygon has at least 3 vertices.

The intersection of two polygons consists of 0 or more connected regions. Your problem is to take two polygons and determine the regions of their intersection that are polygons satisfying the criteria above.

Формат входного файла

The input contains several data sets, each consisting of two polygons. Each polygon appears as a sequence of numbers:

$$n, x_1, y_1, x_2, y_2, \dots, x_n, y_n$$

where the integer n is the number of vertices of the polygon, and the real coordinates (x_1, y_1) through (x_n, y_n) are the boundary vertices. The end of input is indicated by two zeroes for the values of n . These two zeroes merely mark the end of data and should not be treated as an additional data set.

Формат выходного файла

For each data set, your program should output its number (Data set 1, Data set 2, etc.), and the number of regions in the intersection of its two polygons. Label each region in the data set (Region 1, Region 2, etc.) and list its vertices in the order they appear when they are visited going either clockwise or counterclockwise around the boundary of the region. The first vertex printed should be the vertex with the smallest x -coordinate (to break ties, use the smallest y -coordinate). No region may include degenerate parts (consisting of adjacent sides whose angle of intersection is 0). If the three endpoints of two adjacent sides are collinear, the two sides should be merged into a single side. Print each vertex in the standard form (x, y) , where x and y have **two** digits to the right of the decimal. The following sample input contains exactly one data set. (The data set corresponds to the illustration at the beginning of this problem description.)

Примеры

poly.in
3 2 1 0.5 3.5 8 5
5 1.5 3 2 7 6.5 6.5 6.5 3.25 4 4.5
0
0
poly.out
Data Set 1
Number of intersection regions: 2
Region 1: (1.50,3.00) (1.59,3.72) (3.25,4.05)
Region 2: (4.43,4.29) (6.50,4.70) (6.50,4.00) (5.86,3.57)

Задача J. The Worm in the Apple

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

Willy the Worm was living happily in an apple — until some vile human picked the apple, and started to eat it! Now, Willy must escape!

Given a description of the apple (defined as a convex shape in 3D space), and a list of possible positions in the apple for Willy (defined as 3D points), determine the minimum distance Willy must travel to get to the surface of the apple from each point.

Формат входного файла

Input file will begin with a line with a single integer n ($4 \leq n \leq 1,000$), which tells the number of points describing the apple. On the next n lines will be three integers x , y and z ($-10,000 \leq x, y, z \leq 10,000$), where each point (x, y, z) is either on the surface of, or within, the apple. The apple is the convex hull of these points. No four points will be coplanar. Following the description of the apple, there will be a line with a single integer q ($1 \leq q \leq 10^5$), which is the number of queries — that is, the number of points where Willy might be inside the apple. Each of the following q lines will contain three integers x , y and z ($-10,000 \leq x, y, z \leq 10,000$), representing a point (x, y, z) where Willy might be. All of Willy's points are guaranteed to be inside the apple.

Формат выходного файла

For each query, output a single floating point number, indicating the minimum distance Willy must travel to exit the apple. Output this number with 4 decimal places of accuracy. Output each number on its own line, with no spaces, and do not print any blank lines between answers.

Примеры

worm.in	worm.out
6	1.0000000000
0 0 0	2.8867513459
100 0 0	7.0000000000
0 100 0	2.0000000000
0 0 100	
20 20 20	
30 20 10	
4	
1 1 1	
30 30 35	
7 8 9	
90 2 2	
0	

Задача K. Жизнь насекомых

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

В комнате, имеющей форму куба со стороной 200 см, находится N мух. Каждая муха заявила, что ей принадлежит та часть комнаты, которая ближе к ней, чем к какой-либо другой мухе. Мудрый Паук, сидящий в углу, заметил, что первой мухе принадлежит $A_1 \text{ см}^3$, второй - $A_2 \text{ см}^3$, ..., N -ой - $A_N \text{ см}^3$. Какие числа назвал Мудрый Паук?

Формат входного файла

В первой строке входного файла записано натуральное число N ($1 \leq N \leq 6$) - количество мух в комнате. Далее в N строках содержатся описания их положений на момент сделанного заявления. Каждое положение описывается тройкой целых чисел x_i, y_i, z_i ($0 \leq x_i, y_i, z_i \leq 200$) - координатами мухи в пространстве. Никакие две мухи не находятся в одной точке. Система координат введена таким образом, что один угол комнаты имеет координаты $(0, 0, 0)$, а противоположный - $(200, 200, 200)$.

Формат выходного файла

Выведите N действительных чисел A_1, A_2, \dots, A_N .

Примеры

insects.in	insects.out
3	3000000.000000
0 100 100	4000000.000000
150 100 100	1000000.000000
200 100 100	