

Задача А. Ближайшие точки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Дано n точек в \mathbb{R}^3 с целочисленными координатами. Найдите наименьшее расстояние по всем парам точек.

Формат входных данных

В первой строке находится единственное число n — количество точек. $1 \leq n \leq 150000$.

В следующих n строках находятся сами точки, каждая из которых задаётся тройкой целых чисел (x_i, y_i, z_i) . $1 \leq x_i, y_i, z_i \leq 10^5$.

Формат выходных данных

Выведите единственное целое число — квадрат наименьшего расстояния.

Пример

стандартный ввод	стандартный вывод
2 1 1 1 2 2 2	3

Задача В. Адская мухобойка

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

У Жени дома летает очень много ос. Они постоянно летают под потолком в одних и тех же местах. Теперь Евгений отправился в магазин для покупки новой мухобойки. Все мухобойки имеют форму круга с различными радиусами. Женя — очень экономный студент, поэтому он решил купить самую дешевую мухобойку — с минимально возможным радиусом, но Женя так же очень прагматичен, поэтому он купит только такую мухобойку, что с её помощью можно будет одним ударом убить всех ос. Помогите ему! Для простоты можете считать, что на потолке введена стандартная декартова система координат, и координаты ос постоянны. Помните, что ос у Жени действительно много.

Формат входных данных

В первой строке входного файла содержится число N — количество ос ($1 \leq N \leq 100\,000$). Далее содержатся координаты ос — пара целых чисел, не превосходящих по модулю 10^6 .

Формат выходных данных

В первой строке выходных данных выведите координаты точки, в которой Евгений должен нанести свой сокрушительный удар (это та точка, в которой будет расположен центр мухобойки). На следующей строке выведите одно число — минимальный радиус мухобойки, которого будет достаточно, чтобы уничтожить всех омерзительных ос. Ваш ответ будет считаться правильным, если его абсолютная или относительная погрешность не будет превышать 10^{-6} .

Примеры

<code>circlecover.in</code>	<code>circlecover.out</code>
3	1.00 1.00
0 2	1.4142135624
0 0	
2 0	

Задача С. Принцесса

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

Принцесса Евлампия живет в замке, окруженном забором. Жизнь принцессы тяжела, но при этом и очень интересна. Главным ее развлечением является общение с многочисленными поклонниками, постоянно прибывающими из соседних замков, городов и даже королевств.

Замок принцессы окружен забором, представляющим из себя выпуклый многоугольник. Отец принцессы, король, достаточно строг, поэтому всем поклонникам принцессы приходится попадать туда через единственную во всем заборе дырку, вместо того, чтобы войти на территорию замка через парадные ворота. Дырка находится в одной из вершин многоугольника. При этом, если пройти напрямую к дырке поклоннику не удастся, ему придется обходить забор вдоль его периметра. Естественно, каждому поклоннику интересно, сколько ему придется пройти, чтобы попасть из точки своего начального местоположения к дырке, и все спрашивают об этом принцессу, перед тем как прийти к ней в гости.

Принцесса составила список начальных местоположений всех своих поклонников и описание забора вокруг замка. Вам необходимо для каждого поклонника сообщить длину кратчайшего пути от точки его начального положения до точки, в которой находится дырка. При этом, естественно, ни одна точка этого пути не должна лежать внутри многоугольника, представляющего забор, но может лежать на его границе.

Формат входных данных

В первой строке входного файла находятся два целых числа n и k ($3 \leq n \leq 100\,000$, $1 \leq k \leq n$) — количество вершин в многоугольнике, представляющем забор, и номер вершины, в которой находится дырка. В следующих n строках содержатся пары целых чисел x_i и y_i , описывающих координаты вершин многоугольника в порядке обхода против часовой стрелки.

В следующей строке дано одно целое число m ($1 \leq m \leq 100\,000$) — количество поклонников принцессы. В следующих m строках содержатся пары целых чисел x_i и y_i , описывающих координаты начального положения очередного поклонника.

Все координаты не превышают 10^9 по абсолютной величине.

Формат выходных данных

Для каждого поклонника выведите одно число — ответ на задачу. Ответ должен отличаться от правильного не более, чем на 10^{-5} .

Примеры

<code>princess.in</code>	<code>princess.out</code>
4 2	3.23606797749979
0 1	2.0
0 0	
1 0	
1 1	
2	
2 2	
-2 0	

Задача D. Не курить!

Имя входного файла:	stdin
Имя выходного файла:	stdout
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 входных данных

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 выходных данных

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

stdin	stdout
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

Замечание

Напишите.

Задача Е. Большой треугольник

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 8 секунд
Ограничение по памяти: 1024 мегабайта

Есть странная особенность: если соединить между собой города Ростов, Таганрог, Шахты, то получится треугольник

«Невероятно, но факт»

В ЛКШ приезжают ученики из самых разных уголков России и зарубежья. Вы отметили на карте все города, откуда приехали участники ЛКШ.

Затем, вы решили подготовить интересную инфографику на основе этой карты. Первое что вы захотели сделать — это найти три города на этой карте, так чтобы получился треугольник площади ровно S .

Формат входных данных

В первой строке входных данных находится два целых числа n и S ($3 \leq n \leq 3333$, $1 \leq S \leq 2 \cdot 10^{18}$) — количество городов на карте и требуемая площадь треугольника.

В следующую n строках находится описание городов, по одной на строке. Каждый город описывается своими координатами x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$).

Гарантируется, что все города находятся в различных точках. Также гарантируется, что никакие три города не лежат на одной прямой.

Формат выходных данных

Если решения не существует — выведите «No» (без кавычек).

Иначе выведите «Yes» (без кавычек), после чего выведите три пары координат — координаты городов, образующих треугольник площади S .

Примеры

стандартный ввод	стандартный вывод
3 7 0 0 3 0 0 4	No
4 3 0 0 2 0 1 2 1 3	Yes 0 0 1 3 2 0

Задача F. Выпуклые оболочки

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

Выпуклая оболочка множества точек плоскости — наименьший выпуклый многоугольник, содержащий эти точки.

Вам дано n точек на плоскости. Произвольно одна из них выбирается и удаляется.

Найдите среднее число вершин выпуклой оболочки результирующего множества. В этой задаче считайте, что если выпуклая оболочка — отрезок, то в ней две вершины. Если же она — невырожденный многоугольник, то все углы при вершинах строго меньше π .

Формат входных данных

В первой строке содержится единственное число n ($3 \leq n \leq 200\,000$ — количество точек во множестве). В последующих n строках заданы пары чисел, не превышающих по модулю 10^9 — координаты точек. Никакие две точки не совпадают.

Формат выходных данных

Выведите среднее число вершин в выпуклой оболочке множества без одной точки в виде несократимой дроби p/q .

Примеры

<code>average.in</code>	<code>average.out</code>
5 0 0 0 4 4 0 3 3 4 4	17/5

Задача G. Локализация в триангуляции

Имя входного файла: kirkpatrick.in
Имя выходного файла: kirkpatrick.out
Ограничение по времени: 11 секунд
Ограничение по памяти: 256 мегабайт

Вам дана триангуляция множества точек, имеющих выпуклой оболочкой треугольник и множество точек-запросов. Нужно для каждой точки запроса выяснить, в каком треугольнике триангуляции она лежит. Если точка лежит на границе нескольких треугольников, можно вывести любой из них.

Формат входных данных

В первой строке заданы числа n и m — число вершин и граней триангуляции ($3 \leq n \leq 100\,000$, $m = 2n - 5$). В следующих n строках заданы по паре целых чисел x_i, y_i — координаты вершин ($-10^9 \leq x_i, y_i \leq 10^9$). В следующих m строках задано по три числа a_i, b_i, c_i ($1 \leq a_i, b_i, c_i \leq n$) — номера вершин, составляющих i -й треугольник. Все треугольники заданы в порядке обхода против часовой стрелки. Все треугольники не вырожденные. Охватывающий треугольник (внешняя грань) состоит из вершин 1, 2 и 3 (против часовой стрелки) и не задан во входном файле.

В следующей строке задано число запросов q ($1 \leq q \leq 100\,000$). В следующих q строках заданы запросы $x_{qi} y_{qi}$ ($-10^9 \leq x_{qi}, y_{qi} \leq 10^9$). Гарантируется, что все запросы лежат строго внутри охватывающего треугольника.

Формат выходных данных

Для каждой точки-запроса выведите номер треугольника, в котором она лежит. Треугольники нумеруются в порядке следования во входном файле начиная с 1.

Примеры

kirkpatrick.in	kirkpatrick.out
5 5	3
-2 -3	1
3 0	4
-2 3	1
0 -1	
0 1	
1 5 3	
1 4 5	
4 2 5	
5 2 3	
1 2 4	
4	
1 0	
-2 0	
0 1	
-1 1	

Задача Н. Караваны

Имя входного файла: `caravans.in`
Имя выходного файла: `caravans.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

В этой задаче вам нужно грабить караваны.

В пустыне есть n оазисов (пусть они находятся в точках на плоскости). Иногда караваны отправляются от одного оазиса к другому оазису. Чтобы грабить указатели, нужно уметь предсказывать их пути. Но как это сделать? Ответ знает Номад. Скорость караванов постоянна, и они пытаются минимизировать максимальное время вне оазисов. Поэтому можно считать путь караванов ломаной. Вам известны несколько пар оазисов, и вам нужно найти максимальную длину отрезка вне оазисов оптимального пути каравана, который идет от первого оазиса ко второму. Все оазисы находятся в разных точках и никакие три оазиса не лежат на одной прямой.

Формат входных данных

В первой строке дано одно целое число n — количество оазисов ($3 \leq n \leq 100\,000$).

В следующих n строках даны пары целых чисел x_i, y_i — координаты оазисов ($0 \leq x_i, y_i \leq 10\,000$).

В следующей строке дано одно целое число q — количество караванов ($1 \leq q \leq 100\,000$).

В следующих q строках даны пары целых чисел s_i, t_i — стартовый и конечный оазис на пути каравана ($1 \leq s_i, t_i \leq n$).

Формат выходных данных

Выведите q чисел — длины искомым максимальных отрезков на пути, с относительной или абсолютной погрешностью 10^{-9} .

Примеры

<code>caravans.in</code>	<code>caravans.out</code>
3	50.990195135928
0 0	100.498756211209
50 10	100.498756211209
150 0	
3	
1 2	
1 3	
2 3	

Задача I. 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 входных данных

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 выходных данных

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 J. 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 входных данных

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 выходных данных

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

<code>nearest.in</code>	<code>nearest.out</code>
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 К. 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

В 3D даны n точек и q точек-запросов. Постройте выпуклую оболочку n точек и для каждого из q запросов найдите расстояние до поверхности выпуклой оболочки.

Формат входных данных

На первой строке число n ($4 \leq n \leq 1,000$). Далее n строк содержат точки для выпуклой оболочки. Никакие четыре из них не лежат в одной плоскости. Далее число запросов q ($1 \leq q \leq 10^5$) и q строк, содержащих точки-запросы. Все точки запросы гарантированно внутри выпуклой оболочки. Все координаты по модулю не более 10 000.

Формат выходных данных

Для каждого запроса выведите число с четырьмя знаками после запятой – расстояния до поверхности выпуклой оболочки.

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	