

Задача А. Точка в многоугольнике

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

Формат входных данных

В первой строке содержится три числа — N ($3 \leq N \leq 100\,000$) и координаты точки. Последующие N строк содержат координаты углов многоугольника. Координаты — целые, не превосходят 10^6 по модулю.

Формат выходных данных

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

Примеры

<code>point.in</code>	<code>point.out</code>
3 2 3 1 1 10 2 2 8	YES

Задача В. Выпуклая оболочка

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

Вам дано множество точек на плоскости. Найдите их выпуклую оболочку.

Формат входных данных

Первая строка входного файла содержит целое число n — количество точек ($3 \leq n \leq 200\,000$). В следующих n строках описываются точки. i -ая строка состоит из двух целых чисел — координат i -ой точки. Координаты не превосходят 10^9 по модулю. Гарантируется, что все точки не лежат на одной прямой. Точки могут совпадать.

Формат выходных данных

В первую строку выходного файла выведите количество вершин в выпуклой оболочке. Во вторую — номера вершин через пробел, которые ее образуют. Выводите вершины в порядке обхода против часовой стрелки. Никакие два ребра выпуклой оболочки не должны лежать на одной прямой.

В третью строку выведите периметр оболочки, в четвертую - ее площадь.

Периметр должен быть выведен с абсолютной или относительной погрешностью не больше 10^{-9} . Площадь должна быть выведена абсолютно точно.

Примеры

<code>convex.in</code>	<code>convex.out</code>
5	4
0 0	3 5 1 4
1 1	6.47213595499958000000
2 2	2.0
1 0	
0 1	

Задача С. Теодор Рузвельт

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

«Теодор Рузвельт» — флагман военно-морского флота Кукуляндии. Заклятые враги кукуляндцев, флатландцы, решили уничтожить его. Они узнали, что «Теодор Рузвельт» представляет собой выпуклый многоугольник из n вершин и узнали его координаты. Затем они выпустили m баллистических ракет и определили координаты точек, где эти ракеты взорвались. По расчётам штаба флатландцев, «Теодор Рузвельт» будет уничтожен, если в него попадёт хотя бы k ракет. Вычислите, удалось ли флатландцам уничтожить корабль.

Формат входных данных

В первой строке через пробел записаны целые числа n , m , k ($3 \leq n \leq 10^5$, $0 \leq k \leq m \leq 10^5$). В последующих n строках записаны координаты вершин многоугольника в порядке обхода против часовой стрелки. В следующих m строках записаны координаты точек. Гарантируется, что все координаты — целые числа, не превосходящие по модулю 10^9 .

Формат выходных данных

Выведите «YES», если в многоугольнике или на его границе лежит по крайней мере k точек, и «NO» в противном случае.

Примеры

stdin	stdout
5 4 2 1 -1 1 2 0 4 -1 2 -1 -1 -2 -1 1 -1 0 1 2 3	YES

Задача D. Долина лотосов

Имя входного файла: `area.in`
Имя выходного файла: `area.out`
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Дед Андрей выкопал у себя на участке небольшой водоём в форме выпуклого многоугольника. Сначала он планировал разводить и ловить в нём рыбу, и изредка уходить в недельное плавание с друзьями на собственной яхте.

Дед любил мечтать и как-то раз он подумал, что было бы неплохо развести лотосы в водоёме, и в период цветения водить экскурсии.

Дед стал думать, какую часть водоёма огородить для прихотливого цветка, и ему стало интересно сколько лотосов там вырастет. Оказалось, что количество лотосов, которые смогут вырасти на каком-то участке, зависит от площади этого участка, а от этого может зависеть и цена экскурсии!

Для простоты дед начал думать, как огородить часть водоёма, напрямую соединив какие-то два его различных угла мостом. Помогите деду Семёну определить площадь огороженной части, для каждого варианта, который пришёл ему в голову.

Формат входных данных

В первой строке задано натуральное число N ($3 \leq N \leq 50000$) — количество углов в водоёме. Далее в N строках записаны координаты углов водоёма в порядке обхода против часовой стрелки ($-10^9 \leq x_i, y_i \leq 10^9, 1 \leq i \leq N$). Все координаты целые и чётные. В i -ой строке записаны координаты i -ого угла. Затем следует число M ($1 \leq M \leq 50000$) — количество вариантов, пришедших в голову Андрея. Далее в M строках записаны пары номеров углов ($1 \leq l_i, r_i \leq N$), которые хочет соединить Андрей.

Формат выходных данных

Для каждой пары l_i, r_i выведите площадь участка водоёма, который лежит по правую сторону, если бы посетители шли по мосту из угла с номером l_i в угол с номером r_i .

Примеры

area.in	area.out
5	22
4 2	32
6 0	
12 4	
10 10	
4 8	
2	
2 4	
4 2	

Задача Е. Правильный многоугольник

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

Даны n пар вещественных чисел. Есть гипотеза, что это координаты вершин невырожденного правильного n -угольника, записанные в порядке обхода (либо в положительном, либо в отрицательном направлении). Проверьте, так ли это.

Формат входных данных

В первой строке записано целое число n ($3 \leq n \leq 100$). В i -й из следующих n строк через пробел записаны вещественные числа x_i и y_i ($0 \leq x_i, y_i \leq 1$) — координаты i -й точки. Координаты различных точек могут совпадать, но гарантируется, что существует хотя бы одна пара точек на расстоянии не меньше 0.3. Координаты заданы с точностью не менее 10^{-10} .

Формат выходных данных

Если в результате эксперимента не удалось построить вершины правильного n -угольника в порядке обхода, выведите в единственной строке `NO`, в противном случае выведите `YES`. Гарантируется, что в случае отрицательного ответа нельзя изменить координаты точек менее чем на 10^{-5} так, чтобы они стали координатами вершин правильного n -угольника, записанными в порядке обхода.

Примеры

<code>ngon.in</code>	<code>ngon.out</code>
4 0 0 1 0 1 1 0 1	YES
3 0 0 1 0 0.5 1	NO

Задача F. Стена

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

Жил-был жадный король, который однажды приказал главному архитектору окружить королевский замок стеной. Король был настолько жаден, что не желал слушать рассказы архитектора о красивой кирпичной стене с прекрасным силуэтом и изящными высокими башнями. Вместо этого он приказал окружить замок стеной, затратив минимальное количество камня и времени, но потребовал, чтобы стена не подходила к замку ближе, чем на заданное расстояние. Если король узнает, что архитектор потратил не минимально возможное количество ресурсов, то архитектор лишится головы. Более того, король потребовал, чтобы архитектор сразу же предложил проект стены с указанием минимального количества ресурсов, необходимых для постройки.

Вы должны помочь архитектору сохранить голову, написав программу для поиска минимальной длины стены, удовлетворяющей условиям короля.

Задачу упрощает то, что замок короля имеет форму многоугольника и расположен на равнине. Архитектор уже ввел систему координат и точно измерил координаты вершин замка в футах.

Формат входных данных

Первая строка входного файла содержит числа N и L , разделенные пробелом. N ($3 \leq N \leq 1000$) — это количество вершин в королевском замке, а L ($1 \leq L \leq 1000$) — минимальное количество футов, на которое стена может приближаться к замку.

Следующие N строк описывают координаты замка в порядке обхода по часовой стрелке. В каждой строке через пробел записаны целые числа x_i и y_i , разделенные пробелом ($-10000 \leq x_i, y_i \leq 10000$), которые обозначают координаты i -ой вершины. Все вершины различны, и никакие две стороны не пересекаются кроме как по вершинам.

Формат выходных данных

Выведите минимальную длину стены в футах, удовлетворяющей условиям короля с точностью не менее 6 знаков после запятой.

Примеры

<code>wall.in</code>	<code>wall.out</code>
9 100 200 400 300 400 300 300 400 300 400 400 500 400 500 200 350 200 200 200	1628.3185307180

Задача G. Разрезание торта

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

Мама купила Пете на день рождения торт в виде выпуклого многоугольника. Торт большой и вкусный, и Петя хочет разделить его с мамой поровну. Для этого он хочет сделать один прямой разрез, причем ему будет удобнее, если этот разрез будет параллелен оси Oy . Помогите Пете определить, как ему разрезать торт.

Формат входных данных

В первой строке записано целое число N ($3 \leq N \leq 10000$) — количество вершин многоугольника. В последующих N строках записаны координаты вершин многоугольника в порядке обхода. Гарантируется, что все координаты — целые числа, не превосходящие по модулю 10^3 .

Формат выходных данных

Выведите x -координату точки, через которую необходимо провести разрез, с точностью не менее 10^{-6} .

Примеры

<code>cut.in</code>	<code>cut.out</code>
4 0 0 0 2 2 2 2 0	1.000000000

Задача Н. Звёздный праздник

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

Двадцать третьего апреля двухтысячного года в королевстве родилась прекрасная принцесса, волосы которой унаследовали волшебный дар солнца — дар вечной молодости. Хитрая старушка Готель украла юную принцессу, чтобы пользоваться этим даром. Она спрятала Рапунцель в высокой башне и никогда не выпускала её.

Каждый год, в свой день рождения, Рапунцель видела в небе фонарики, и ей очень хотелось узнать, откуда они. Она не знала, что фонарики запускали её родители и жители королевства в надежде, что пропавшая принцесса вернётся.

Сначала, ранним утром, она видела 6 фонариков — это слуги королевства проводили пробный запуск, изучая погодные условия. Вечером все жители выходили на улицу и запускали в небо в точности 2304 фонарика в память о дне рождении Рапунцель. Принцесса любовалась фонариками, представляя, что это — прекрасные звёзды. Тогда Рапунцель стала развлекать себя тем, что объединяла фонарики в созвездия — выпуклые, непересекающиеся (по границе), невырожденные многоугольники. (Выпуклым многоугольником называется многоугольник, все точки которого лежат по одну сторону от любой прямой, проходящей через две его соседние вершины. Многоугольники называются непересекающимися по границе, если не существует ни одной точки, принадлежащей сторонам разных многоугольников. Невырожденный многоугольник — это многоугольник, у которого площадь положительна.)

Развлекитесь вместе с Рапунцель, поздравьте её с днём рождения и объедините фонарики в любое количество созвездий.

Формат входных данных

В первой строке единственное целое число n ($n = 6$ или $n = 2304$).

В каждой из следующих n строк содержится по два целых числа x_i, y_i ($1 \leq x_i, y_i \leq 10^9$) — координаты i -й точки. Гарантируется, что никакие три точки не лежат на одной прямой.

Формат выходных данных

В первой строке выведите количество созвездий k .

В каждой из последующих k строк выведите выпуклое непересекающееся невырожденное созвездие: сначала количество звезд в созвездии, а затем номера точек — звёзд этого созвездия в порядке обхода.

Примеры

<code>rapunzel.in</code>	<code>rapunzel.out</code>
6	2
6 1	3 1 1 1 4 3 7
3 7	3 6 1 6 7 8 4
8 4	
1 4	
1 1	
6 7	
6	2
2 1	3 2 1 1 4 4 3
4 3	3 7 4 7 1 4 6
7 4	
4 6	
7 1	
1 4	

Задача I. Сыр

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Крешо купил вкуснейший сыр с перцем, но Степану перец не нравится, поэтому он хочет отрезать кусок, на котором не было бы перца. Сыр имеет форму выпуклого многоугольника, а каждая перчинка является точкой внутри него. Степан режет сыр только 1 раз. Он выбирает две вершины многоугольника, не являющиеся смежными, и режет по диагонали, соединяющей их. Затем Степан забирает ту из получившихся частей, на которой нет перца (ни внутри, ни на границе).

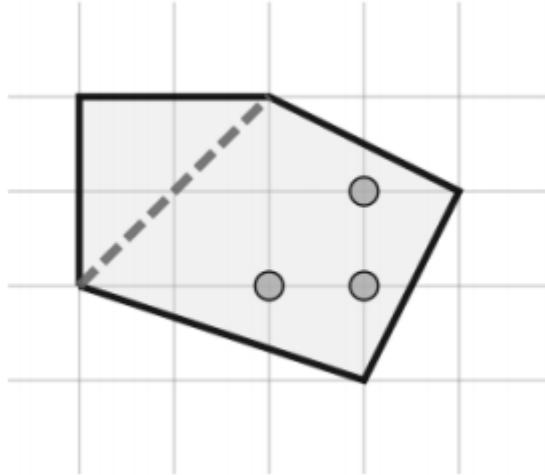


Рисунок соответствует первому тесту. Пунктирной линией показан разрез Степана.

Напишите программу, которая определит, может ли Степан отрезать кусок без перца. Если он может это сделать, выведите максимальную площадь куска, который может отрезать Степан.

Формат входных данных

Первая строка входного файла содержит одно целое число N — количество вершин в многоугольнике ($4 \leq N \leq 300\,000$, $1 \leq M \leq 300\,000$). Каждая из следующих N строк содержит два числа x_i и y_i — координаты i -й вершины. Следующая строка содержит одно число M — количество перчинок. Каждая из следующих M строк содержит два числа x_i и y_i — координаты i -й перчинки. Все координаты — целые числа, по модулю не превосходящие 10^9 .

Вершины многоугольника заданы в порядке обхода против часовой стрелки и образуют выпуклый многоугольник. Никакие две подряд идущие стороны не параллельны.

Все перчинки расположены в различных точках и внутри многоугольника (они не расположены на стороне или снаружи многоугольника).

Формат выходных данных

Выведите единственное число — удвоенную максимальную площадь (это число всегда целое). Если отрезать кусок без перца невозможно, выведите 0.

Примеры

стандартный ввод	стандартный вывод
5 0 1 3 0 4 2 2 3 0 3 3 2 1 3 1 3 2	4
6 -3 3 -3 -4 -2 -5 2 -5 3 -4 3 3 7 1 0 0 -1 0 -3 2 0 0 0 0 2 -1 0	10
6 0 3 -1 2 -1 -2 0 -3 1 -2 1 2 1 0 0	4

Замечание

Пояснение ко второму примеру: Степан делает разрез от вершины 2 к вершине 5.

Пояснение ко третьему примеру: Степан делает разрез от вершины 1 к вершине 3.