

Задача А. Выпуклая оболочка

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

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 В. Теодор Рузвельт

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

«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 лежит по крайней мере k точек, и «NO» в противном случае.

Примеры

theodore.in	theodore.out
5 4 2 1 -1 1 2 0 4 -1 2 -1 -1 -2 -1 1 -1 0 1 2 3	YES

Задача С. 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 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

<code>smoking.in</code>	<code>smoking.out</code>
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 D. Место встречи изменить нельзя

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

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

Формат входных данных

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 выходных данных

Единственная строка выходного файла должна содержать координаты двух выбранных точек.

Пример

<code>rendezvous.in</code>	<code>rendezvous.out</code>
4	0 0
0 0	0 1
0 1	
1 1	
1 0	

Задача Е. Платные дороги (задача на 5)

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

Мэр одного большого города решил ввести плату за проезд по шоссе, проходящим в районе города, чтобы снизить объем транзитного транспорта. В районе города проходит n шоссе.

Но руководство области, в которой расположен город, воспротивилось планам мэра. Действительно — дальнбойщики представляют собой неплохой источник доходов для большого количества кафе и гостиниц в небольших городках.

В результате решили, что плата будет введена только на шоссе, которые *проходят через город*.

В городе используется развитая система метрополитена, всего в городе есть m станций метро. Решено было, что шоссе проходит через город, если либо одна из станций метро расположена непосредственно на шоссе, либо есть хотя бы одна станция с каждой стороны от шоссе.

Помогите теперь мэру определить, какие шоссе проходят через город.

Формат входных данных

Первая строка входного файла содержит два целых числа: n и m — количество шоссе и количество станций метро, соответственно ($1 \leq n, m \leq 100\,000$).

Следующие n строк описывают шоссе. Каждое шоссе описывается тремя целыми числами a , b и c и представляет собой прямую на плоскости, задаваемую уравнением $ax + by + c = 0$ ($|a|, |b|, |c| \leq 10^6$).

Следующие m строк входного файла описывают станции метро. Каждая станция описывается двумя целыми числами x и y и представляет собой точку на плоскости с координатами (x, y) ($|x|, |y| \leq 10^6$).

Формат выходных данных

Первая строка выходного файла должна содержать одно целое число — количество шоссе, которые проходят через город. Вторая строка должна содержать номера этих шоссе в возрастающем порядке. Шоссе нумеруются от 1 до n в порядке, в котором они описаны во входном файле.

Примеры

highways.in	highways.out
4 2	3
0 1 0	1 3 4
1 0 1	
1 1 0	
1 1 -1	
0 0	
2 0	