

## Задача А. Делоне

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

Где карта, Билли? Нам нужна карта...

---

Черный пес

Этой ночью на «Адмирал Бенбоу» нападет шайка пиратов, желающая завладеть картой сокровищ. Но юный Джимми Хокинс уже готов их встретить. Его знакомый пират Делоне сконструировал хитрый механизм, представляющий собой секретные ловушки, а сам Джим сидит на втором этаже с сетями.

Ловушки представляют собой набор скрытых рычагов, расположенных на плоскости. Некоторые рычаги объединены между собой в ловушки. Ловушка может быть образована любыми тремя рычагами при условии, что ее площадь не равна нулю. При этом общая площадь любой пары ловушек равна нулю. Также гарантируется, что суммарная площадь, покрытая ловушками, максимальна.

Система работает следующим образом. Как только пират оказывается на территории одной из ловушек, Джим кидает сверху круглую сеть, в которую оказывается пойман пират. При этом край сети, имеющий форму окружности, касается всех трех рычагов, образующих ловушку, в которую попал пират.

Единственный недостаток этой системы заключается в том, что если в момент бросания сети она заденет своей внутренней точкой какой-нибудь рычаг, не имеющий отношения к нужной ловушке, вся система сломается. Джим хочет проверить, является ли система надежной. Более точно, он хочет выяснить, существует ли такая ловушка, что внутри сети, относящейся к этой ловушке (окружности, описанной вокруг треугольника, образованного соответствующими рычагами), содержится еще хотя бы один рычаг.

### Формат входных данных

В первой строке входного файла задано число  $t$  ( $1 \leq t \leq 5$ ) — число тестов. Далее идут описания  $t$  тестов.

В первой строке описания теста задано число  $n$  ( $3 \leq n \leq 50\,000$ ) — число рычагов в системе и число  $m$  — число ловушек.

Далее в  $n$  строках заданы целые числа  $x_i, y_i$  ( $|x_i|, |y_i| \leq 10\,000$ ) — координаты рычагов.

Далее в  $m$  строках заданы по три числа — описания ловушек.

Каждое описание состоит из трёх чисел  $a_j, b_j, c_j$  ( $1 \leq a_j, b_j, c_j \leq n$ ) — номера рычагов, из которых состоит  $i$ -я ловушка.

### Формат выходных данных

Для каждого теста в отдельной строке выведите «YES», если система является надёжной, то есть, в любой описанной вокруг рычагов ловушки окружности больше не содержится рычагов или «NO» в ином случае.

## Пример

стандартный ввод	стандартный вывод
2	YES
4 2	NO
0 1	
0 -1	
-2 0	
2 0	
1 2 3	
1 2 4	
4 2	
0 1	
0 -1	
-2 0	
2 0	
3 4 2	
3 4 1	

## Задача В. Адская мухобойка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
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 дешёвую 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  $10^{-6}$ .

### Пример

стандартный ввод	стандартный вывод
3	1.00000000000000000000 1.00000000000000000000
0 2	1.41421356237309510000
0 0	
2 0	

## Задача С. 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

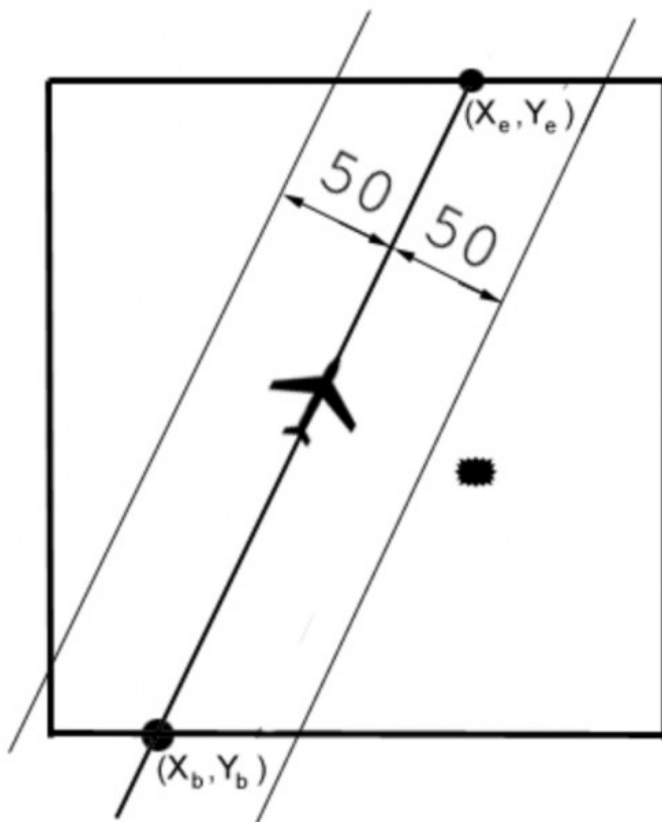
стандартный ввод	стандартный вывод
3 7 0 0 3 0 0 4	No
4 3 0 0 2 0 1 2 1 3	Yes 0 0 1 3 2 0

## Задача D. Лесной пожар

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

В МЧС поступило сообщение о возможном лесном пожаре в заданном квадрате тайги. Для поиска места возгорания было послано  $N$  самолетов. Однако ни один из экипажей пожар не обнаружил.

Известно, что с самолета видна полоса тайги, границы которой находятся на расстоянии 50 км справа и слева от той линии на поверхности Земли, над которой пролетает самолет (см. рисунок), причем точки, находящиеся на расстоянии ровно 50 км от этой линии, все еще видны. Донесение с каждого самолета содержало информацию о том, в каких двух различных точках  $(x_b, y_b)$  и  $(x_e, y_e)$  самолет входил в заданный квадрат и покидал его соответственно. Между этими точками самолет двигался строго по прямой.



Требуется написать программу, которая определит, действительно ли весь заданный квадрат тайги был просмотрен с самолетов. Если это не так, то программа должна находить координаты какой-нибудь точки, находящейся внутри или на границе квадрата и не попавшей ни в одну из просмотренных полос.

### Формат входных данных

В первой строке записано натуральное число  $L$  — размер заданного квадрата тайги в километрах ( $0 \leq L \leq 1000$ ).

Во второй строке — натуральное число  $N$  ( $1 \leq N \leq 100$ ) — количество самолетов.

В каждой из последующих  $N$  строк записано донесение с самолета — четыре вещественных координаты  $x_b, y_b, x_e, y_e$ . Координаты заданы в километрах. Стороны квадрата тайги параллельны осям координат, его левый нижний угол находится в точке с координатами  $(0, 0)$ , а правый верхний — в точке  $(L, L)$ .

## Формат выходных данных

Если заданный квадрат был просмотрен полностью, то выведите единственное слово «YES», написанного заглавными латинскими буквами. В противном случае выведите слово «NO», а в следующей строке выведите через пробел координаты  $x$  и  $y$  какой-либо точки, которая не попала ни в одну из просмотренных полос. Координаты нужно выводить в километрах с ошибкой не более одного метра (в одном километре 1000 метров).

## Примеры

стандартный ввод	стандартный вывод
245 1 26.1 0 193.568 245	NO 155.123 100
50 1 0 25 25 25	YES

## Замечание

Вы можете написать тут нормальное решение и умереть. А можете вспомнить про точку в квадрате, непокрытую окружностями.

## Задача Е. Бульдозер

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

Королевство JOI известно добычей золота. Каждый год они используют бульдозер для его добычи.

Земля королевства описывается плоскостью с декартовыми координатами. Есть  $n$  мест,  $i$ -е место имеет координаты  $(x_i, y_i)$ . В каждом месте возможно добыть золото или камни, но нельзя одновременно и то, и другое.

Если в  $i$ -м месте можно добыть золото, то его ценность там равна  $v_i$ . Если же там камни, то, чтобы избавиться от них, необходимо потратить  $c_i$ .

Мы используем бульдозер следующим образом. Сначала мы выбираем две параллельные прямые, затем добываем всё золото и камни, находящиеся между этими прямыми, включая их границы.

Выгода равна сумме ценностей золота за вычетом трат на избавление от камней. Необходимо максимизировать выгоду.

### Формат входных данных

Первая строка содержит целое число  $n$  ( $1 \leq n \leq 2000$ ) — количество мест.

Далее следуют  $n$  строк,  $i$ -я из которых содержит три целых числа  $x_i, y_i, w_i$  ( $|x_i|, |y_i|, |w_i| \leq 10^9$ ,  $w_i \neq 0$ ) — координаты места  $(x_i, y_i)$  и специальное число.

- Если  $w_i \geq 1$ , то  $i$ -е место содержит золото, и  $v_i = w_i$ .
- Если  $w_i \leq -1$ , то  $i$ -е место содержит камни, и  $c_i = -w_i$ .

Гарантируется, что все точки различны.

### Формат выходных данных

Выведите одно число — ответ.

## Примеры

стандартный ввод	стандартный вывод
5 -5 5 -2 2 5 10 1 4 -2 4 -5 4 -2 2 7	19
6 0 0 6 1 0 -2 2 0 8 0 1 -2 1 1 5 2 1 -2	15
5 0 0 2 4 0 2 3 2 -1 1 2 2 1 1 -1	5
2 0 0 -1 1 0 -1	0
15 10 3 30 5 10 -17 4 -5 14 0 -3 -9 -2 3 17 6 9 -19 -9 -6 -14 -2 -3 10 -3 -3 30 8 1 -28 9 -9 -5 7 -5 -24 -8 -10 5 -7 2 20 10 -3 -13	107



## Задача F. Астрономия

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	512 мегабайт

18 год нашей эры. Известный астроном Филон Берляндский публикует трактат «Об устройстве небесного свода», в котором рассказывает о невероятном явлении, замеченном им во время наблюдений за звёздами. Однажды ночью Филон увидел на безоблачном небе  $2n$  звёзд и Луну. Удивительно, что звёзды можно было мысленно разбить на пары так, что любая прямая, проходящая через центры двух звёзд из пары проходила также и через центр Луны, при этом все такие прямые были различны. Филон тщательно запечатлел данное явление на карте звёздного неба, в которой была введена система координат, и обнаружил, что центры всех звёзд и центр Луны оказались в точках с целочисленными координатами. Поскольку Филон считал, что Земля и Луна плоские, система координат на карте была двумерная. Система координат была выбрана астрономом так, чтобы координаты всех объектов, включая Луну, по модулю не превосходили  $10^6$ . Помимо этого, никакие два объекта (две звезды или звезда и Луна) не были в одной точке.

Помимо карты звёздного неба Филон Берляндский написал в своём трактате предсказание, что спустя 2000 лет звёзды на небе вернутся в то же положение, а на месте Луны появится огромная комета, которая уничтожит Землю.

2018 год нашей эры. Вам в руки попадает трактат Филона Берляндского, и вы с ужасом обнаруживаете, что звёзды на небе находятся в том же положении, что и 2000 лет назад! К сожалению, время не пощадило карту астронома, поэтому на ней остались отмеченными только точки, соответствующие центрам звёзд, и не осталось никаких упоминаний, каким образом следовало разбить точки на пары, чтобы все построенные по этим парам прямые прошли через центр Луны. Что ещё хуже, на карте стёрлась точка, соответствующая центру Луны. Чтобы узнать, откуда прилетит комета, и спасти человечество от неминуемой гибели, необходимо срочно восстановить какую-нибудь подходящую позицию центра Луны!

### Формат входных данных

В первой строке задано целое число  $n$  ( $2 \leq n \leq 2600$ ) — количество пар звёзд, которые астроном увидел на небе.

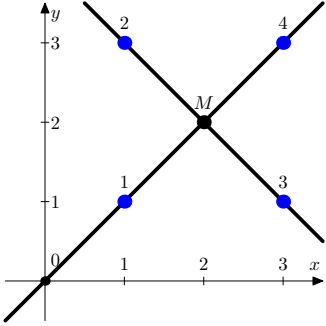
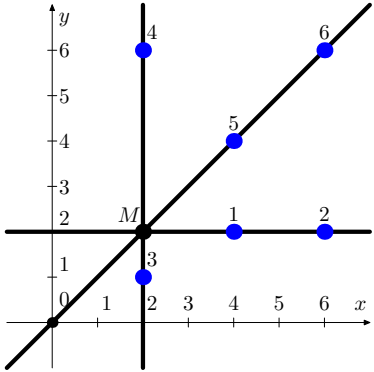
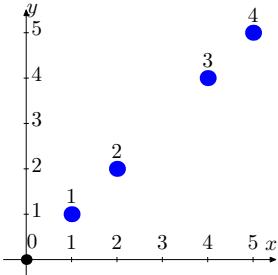
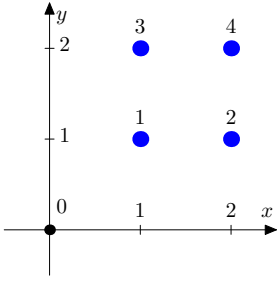
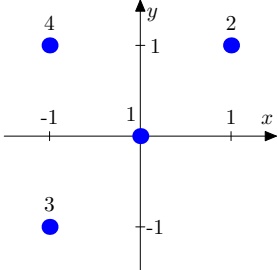
В следующих  $2n$  строках содержатся пары целых чисел  $x_i, y_i$  ( $-10^6 \leq x_i, y_i \leq 10^6$ ) — координаты центров звёзд на карте. Обратите внимание, что звёзды указаны в произвольном порядке, никак не связанном с тем, как Филон Берляндский разделял их на пары. Центры никаких двух звёзд не находятся в одной точке.

### Формат выходных данных

Если астроном ошибся, и не существует способа разбить все точки на пары таким образом, чтобы все построенные по этим прямые были различны и пересекались в одной точке с целыми координатами, отличной от центров всех звёзд, то выведите «No» (без кавычек) в единственной строке вывода.

В противном случае в первой строке выведите «Yes» (без кавычек). Во второй строке выведите пару целых чисел  $x, y$  ( $|x|, |y| \leq 10^6$ ) — координаты точки, в которой находился центр Луны в вашем решении. Если подходящих точек несколько, то выведите любую из них. Обратите внимание, что выведенная точка не может совпадать с центром ни одной из звёзд.

Примеры

стандартный ввод	стандартный вывод	Пояснение
2 1 1 1 3 3 1 3 3	Yes 2 2	
3 4 2 6 2 2 1 2 6 4 4 6 6	Yes 2 2	
2 1 1 2 2 4 4 5 5	No	
2 1 1 2 1 1 2 2 2	No	
2 0 0 1 1 -1 -1 -1 1	No	

## Замечание

В четвёртом тесте из условия центр Луны мог располагаться только в точке  $(1.5, 1.5)$ , но эта точка обладает нецелыми координатами, поэтому ответа нет.

В пятом тесте из условия невозможно найти подходящую точку, которая бы не являлась при этом центром звезды.

## Задача G. Идол Могоху-Ри

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

Давным давно где-то в глубине Америки существовало могущественное племя с великим вождем Пиннивухом. Однажды это племя захватило три города Майя. Пиннивух задумался: необходимо как-то контролировать покоренные территории. Для этого он обратился к жрецам верховного бога Могоху-Ри за помощью.

Жрецы донесли до него волю бога: для контроля этих трех городов необходимо поставить идол Могоху-Ри, создающий религиозное поле над городами. Однако этот идол настолько могущественный, что будучи неуравновешенным тремя жертвенными алтарями, расположенными по одному в каждом городе, он может запросто свести всех людей в округе с ума. Для уравновешивания идола алтари нужно поставить таким образом, чтобы центр масс системы из этих трех точек совпадал с идолом. При подсчете центра масс считается что все алтари имеют одинаковую массу.

Теперь Пиннивух размышляет: куда же поставить идол. У него есть список возвышенностей, пригодных для установки идола. Помогите ему определить, на какие из них можно поставить идол без риска выжечь мозги населению городов религиозным полем.

Каждый город имеет форму выпуклого многоугольника, никакие три вершины которого не лежат на одной прямой. Города могут пересекаться. Каждый алтарь должен прикрепляться к своему городу особым обрядом, причем он обязан находиться на его территории (возможно на границе). Таким образом, на территории города может быть несколько алтарей, но к нему будет относиться ровно один из них. Алтари, идол и возвышенности являются точками на плоскости, некоторые из которых могут совпадать.

Возвышенности рассматриваются независимо друг от друга, расположение алтарей для разных возвышенностей может быть разным.

### Формат входных данных

Сначала идут разделенные пустыми строками описания трех городов в следующем формате:

В первой строке идет целое число  $n$  ( $3 \leq n \leq 5 \cdot 10^4$ ) — количество вершин многоугольника. Следующие  $n$  строк содержат по два целых числа  $x_i, y_i$  — координаты  $i$ -й вершины многоугольника в порядке обхода против часовой стрелки.

После описания городов идет целое число  $m$  ( $1 \leq m \leq 10^5$ ) — количество возвышенностей. Следующие  $m$  строк содержат по два целых числа  $x_j, y_j$  — координаты  $j$ -го холма.

Все координаты во входных данных не превосходят  $5 \cdot 10^8$  по модулю.

### Формат выходных данных

Выведите для каждой возвышенности на отдельной строке «YES» (без кавычек) или «NO» (без кавычек), в зависимости от того, можно ли поставить три жертвенных алтаря для уравновешивания идола или нет.

## Пример

стандартный ввод	стандартный вывод
3	NO
0 0	YES
1 0	NO
1 1	YES
	NO
4	
8 8	
5 5	
6 4	
8 4	
3	
-1 -1	
-3 -1	
-2 -2	
5	
0 0	
2 1	
7 1	
1 1	
5 3	

## Задача Н. Дартс

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

Костя и Ваня решили сыграть в дартс. Ваня сделал  $n$  бросков дротика. Поскольку Ваня не умеет играть в дартс, дротик всегда прилетал в случайную точку с целочисленными координатами в квадрате  $[1, 10^9] \times [1, 10^9]$  (все целочисленные точки этого квадрата равновероятны). Таким образом, получились точки  $A_1, A_2, \dots, A_n$ .

После этого Костя решил проанализировать результаты игры. Для этого он выбрал  $q$  отрезков бросков  $[l, r]$  ( $1 \leq l \leq r \leq n$ ) и хочет узнать разброс попаданий Вани с  $l$ -о по  $r$ -й бросок, то есть наибольшее расстояние  $|A_i A_j|$  по всем  $l \leq i, j \leq r$ . Найдите квадрат этого числа для каждого запроса.

### Формат входных данных

В первой строке находится единственное целое число  $n$  ( $1 \leq n \leq 50\,000$ ) — количество бросков.

В следующих  $n$  строках находится по два целых числа, в  $i$ -й строке  $x_i, y_i$  ( $1 \leq x_i, y_i \leq 10^9$ ) — координаты точки  $A_i = (x_i, y_i)$ .

В следующей строке находится единственное целое число  $q$  ( $1 \leq q \leq 50\,000$ ) — количество запросов.

В следующих  $q$  строках находится по два целых числа  $l, r$  ( $1 \leq l \leq r \leq n$ ) — параметры очередного запроса.

### Формат выходных данных

Выведите ответы на все  $q$  запросов.

### Пример

стандартный ввод	стандартный вывод
8	67436226429387773
916055892 224397651	727030012313212921
390347132 448521124	331261225664082137
732987350 782360531	67436226429387773
74555561 240606351	727030012313212921
643124580 151216775	366164230808265268
468504801 362584660	0
678784493 207870613	0
623464456 461594715	727030012313212921
10	727030012313212921
7 8	
2 4	
4 6	
7 8	
3 5	
4 7	
1 1	
3 3	
1 4	
3 6	

## Задача I. Место встречи изменить нельзя

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.3 секунд  
Ограничение по памяти: 256 мегабайт

Даны  $N$  точек. Найдите такие две из них, 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

стандартный ввод	стандартный вывод
4	0 0
0 0	0 1
0 1	
1 1	
1 0	

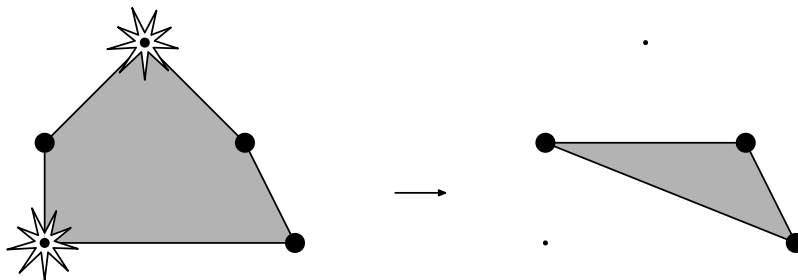
## Задача J. База в джунглях

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

Глубоко в джунглях расположена военная база. Она окружена  $n$  сторожевыми башнями с ультразвуковыми генераторами. В этой задаче сторожевые башни считаются точками на плоскости.

Сторожевые башни генерируют ультразвуковое поле с помощью которого защищают все объекты, расположенные строго внутри выпуклой оболочки сторожевых башен. Не существует сторожевой башни строго внутри выпуклой оболочки и никакие три сторожевые башни не находятся на одной прямой.

Враг может уничтожить несколько башен. Если такое происходит, защищаемая область уменьшается до выпуклой оболочки оставшихся башен.



Командир базы хочет построить штаб внутри защищаемой области. Чтобы максимизировать ее безопасность, он хочет максимизировать минимальное количество башен, который должен уничтожить враг, чтобы оставить штаб без защиты.

### Формат входных данных

Первая строка входного файла содержит одно число  $n$  ( $3 \leq n \leq 50\,000$ ) — количество сторожевых башен. Следующие  $n$  строк содержат декартовы координаты сторожевых башен, по одной паре в строке. Все координаты являются целыми числами и не превосходят по модулю  $10^6$ .

Башни описаны в порядке обхода их выпуклой оболочки по часовой стрелке.

### Формат выходных данных

Выведите в выходной файл количество сторожевых башен, которые должен уничтожить враг, чтобы оставить штаб без защиты, если штаб будет расположен оптимально.

### Примеры

стандартный ввод	стандартный вывод
3 0 0 50 50 60 10	1
5 0 0 0 10 10 20 20 10 25 0	2



## Задача К. Не курить!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

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

Если Петя не сможет взять ни одной сигареты, выведите в единственной строке выходного файла «NO», иначе выведите «YES»

### Примеры

стандартный ввод	стандартный вывод
2 0 0 2 2 0 2 2 0	YES
4 0 0 3 1 2 0 3 -2 4 -1 7 2 -1 2 6 2	NO