

Задача А. Эх, было время...

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

Нас трое: Глеб, Овсеп и Вова — боевое трио из Томского Политеха. Наша команда называлась «*Чапман Рэд*», и это было нашим небольшим бунтарским вызовом самим себе, символом дерзости и уверенности в своих силах. Мы готовились к олимпиаде по программированию Hack IT, которая должна была состояться прямо в Научно-технической библиотеке. Казалось бы, ничего необычного, кроме того, что происходило это целых два года назад. Эх, время-то было — и трава зеленее, и деревья выше... Всю пятницу мы просидели в общаге, не вылезая из-за ноутбуков: тренировка, тренировка и ещё раз тренировка. Легли спать поздно, а проснулись рано в снежную субботу.

Утро было холодным, но удивительно солнечным. Выпавший ночью снег искрился под лучами, и каждый сугроб выглядел как декорация зимней сказки. Мы наспех выпили общажный кофе, собрали тетрадки и, закутавшись потеплее, отправились пешком в библиотеку. Идти было недалеко, но атмосфера вокруг была особенной.

Это было два года назад. Мы шагали по спокойным субботним улицам Томска, а под ногами хрустел свежий снег. Овсеп шутил, пытаясь нас растормошить, Вова был сосредоточен и мысленно прокручивал алгоритмы, а в моей голове еще звенели отголоски бессонной ночи.

Мы миновали старые деревянные дома с резными наличниками, припорошенными снегом. Холодный воздух, полный запаха зимы и свежести, бодрил лучше любого энергетика. Мы шли вперед почти на крыльях — команда «*Чапман Рэд*» была настроена взять призовое место! Эх, время-то было, и трава зеленее, и деревья выше...

Добравшись до Научно-технической библиотеки, мы зарегистрировались и заняли свой стол. В воздухе витало волнение: шум клавиатур, сосредоточенные лица — ожидание старта. Олимпиада началась. Мы работали как единый механизм: я писал основной код, Вова занимался математикой и структурами данных, а Овсеп быстро дебажил и оформлял ввод-вывод.



В середине состязания, когда голова уже начинала трещать, организаторы устроили кофе-брейк и принесли пиццу — настоящее спасение. Перекус, пара шуток — и снова в бой. Несмотря на усталость и недосып, дух команды «*Чапман Рэд*» был на высоте.

Итог превзошел ожидания: мы заняли призовое третье место! Вот это был прикол. Это было два года назад — эх, время-то было, и трава зеленее, и деревья выше...

Вернулись мы в общагу уставшие, но счастливые. Бронза на Наск IT — то, что навсегда останется в памяти.

А вот одна из задач с олимпиады:

На столе лежат n палочек. Палочка номер i имеет длину a_i . Используя эти палочки, вы хотите собрать как можно больше кубов. Чтобы построить один куб, требуется 12 ребер одинаковой длины. Палочки нельзя ломать или соединять; каждая палочка может быть использована не более чем в одном кубе. Определите максимальное количество кубов, которое можно собрать.

Формат входных данных

В первой строке входных данных содержится целое число n ($1 \leq n \leq 2 \cdot 10^5$) — количество палочек. Во второй строке входных данных содержатся n целых чисел a ($1 \leq a_i \leq 10^9$) — длины всех палочек.

Формат выходных данных

Выведите одно целое число — максимальное количество кубов, которое можно собрать из имеющихся палочек.

Примеры

стандартный ввод	стандартный вывод
5 1 2 3 4 5	0
15 3 5 3 3 3 3 3 8 3 3 3 4 3 3	1

Замечание

Это было два года назад — эх, время-то было, и трава зеленее, и деревья выше...

Задача В. Букашки-таракашки

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

Абсолютно в каждом общежитии есть букашки и таракашки. Макар (*настоящее имя изменено по соображениям безопасности*) живет в общежитии ИШИТР ТПУ, и у него есть хобби — разводить тараканов и прочую живность (спасибо тебе, Макар), у него есть террариум, в котором живет n букашек и m таракашек, с которыми он иногда играет или выпускает погулять на волю, чтобы, как он это называет, «нагулять вес».

Сейчас Макару стало скучно, и он решил поделить свою живность на две равные команды и провести между ними товарищеский футбольный матч шариком из хлебного мякиша по правилам ФИФА: с арбитрами, системой видеофиксации и бассейном после матча для победившей команды.

Он опросил каждую из n букашек и каждого из m таракашек и узнал от каждой животинки, с кем она хочет быть в команде. Каждая из n букашек показала лапкой или на другую букашку, с которой хочет быть в команде, или на саму себя, если ей без разницы, с кем быть в команде. Каждый из m таракашек направил усиком или на другого таракашку, с которым хочет быть в команде, или на самого себя, если ему без разницы, с кем быть в команде.

В одной команде могут быть как и букашки, так и таракашки — главное для Макара, чтобы не было ни одной букашки или таракашки, которая хотела быть в одной команде с кем-то, кто оказался в другой команде: если такое произойдет, то обидевшаяся живучка непременно на зло всем съест футбольный мячик из хлебного мякиша, и все останутся без футбола.

Определите, можно ли разделить всю живность Макара на две равные команды.

Формат входных данных

В первой строке входных данных содержатся два целых числа n, m ($1 \leq n, m \leq 1000$) — количество букашек и количество таракашек в террариуме Макара.

Во второй строке содержатся n целых чисел a ($1 \leq a_i \leq n$) — число a_i обозначает номер букашки, с которой букашка с номером i хочет быть в одной команде.

В третьей строке содержатся m целых чисел b ($1 \leq b_j \leq m$) — число b_j обозначает номер таракашки, с которым таракашка с номером j хочет быть в одной команде.

Формат выходных данных

Если существует хотя бы одно разделение на две равные команды согласно описанным условиям, выведите «Yes», иначе выведите «No».

Примеры

стандартный ввод	стандартный вывод
3 5 3 3 1 3 3 1 4 2	Yes
2 2 1 2 2 1	Yes
4 2 3 4 2 4 1 1	No

Задача С. Ноль, Целковый, Полушка, Подушка...

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

Я изобрел новую счетную систему H . Да, использовал обычные знакомые цифры, но это пока.

Суть в том, что все начинается с нуля — это так называемая «база» (или «основа»). Далее идет 1, потом идет 2, затем 3, потом 22, потом 5, затем по порядку 6, 7, 8, 9, затем 10, 11, 12, 13, 122, затем 15 и так далее. Принцип вы уловили — все интуитивно понятно.

Да, в моей системе нет так называемой цифры «четыре». Потому что «4» — это два раза по «2». То есть цифра «4» — это бессмыслица, бред, в нормальной системе ее не должно существовать. Такой некий «аппендикс» незаслуженно популярной десятичной системы счисления.

Пока есть один недостаток (а так, в целом, очень удобная система). Но это, опять же, недостаток только (для) тех, кто любит использовать такой бред, как «четыре». Так вот: не всегда понятно, какое число в моей системе счисления является каким по счету в десятичной системе счисления. Например, в моей системе H число «222» может быть по счету как «24₁₀», так и «42₁₀» или «222₁₀».

По заданной записи S_H некоторого числа в системе H определите и выведите сумму всех десятичных чисел, которые могут ей соответствовать.

Формат входных данных

В единственной строке входных данных содержится строка S ($1 \leq |S| \leq 17$) — запись некоторого числа в системе H . Гарантируется, что запись корректна и не содержит ведущих нулей.

Формат выходных данных

Выведите единственное целое число — сумму всех десятичных чисел, которые могут соответствовать записи S_H .

Примеры

стандартный ввод	стандартный вывод
52	52
22	26
222	288

Замечание

Цифра «четыре» неприкольная.

Задача D. Не баг, а фича!

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

Студент Константин участвует в олимпиаде по программированию *Hack IT* в ТПУ. Он пытается решить самую сложную задачу конкурса, но его решение упорно не проходит, потому что содержит множество мелких багов. Чтобы добиться успеха, Константин дебажит код и отправляет новые послыки, однако после каждой послыки в его решении обнаруживаются новые ошибки.

Процесс работы Константина устроен следующим образом:

- Изначально в решении содержится B багов;
- Он исправляет A багов, затем отправляет послыку в систему (если в программе менее A багов, то Константин исправляет их все, затем отправляет послыку);
- Если после исправления баги все еще остаются, то сразу же появляются еще C новых багов.

Константин продолжает отправлять послыки до тех пор, пока не будет отправлена послыка, не содержащая багов. Определите, сможет ли Константин исправить все баги и, если да, то сколько послылок ему потребуется.

Формат входных данных

В единственной строке входных данных содержатся три целых числа A, B, C ($0 \leq A, B, C \leq 10^{18}$) — количество багов, которые Константин исправляет за одну послыку, количество багов в решении изначально и количество новых багов, возникающих после каждой послыки, соответственно.

Формат выходных данных

Выведите одно целое число — минимальное количество послылок, необходимых Константину для полного исправления всех багов. Если избавиться от всех багов невозможно, выведите «-1».

Примеры

стандартный ввод	стандартный вывод
5 12 3	5
3 10 5	-1
4 12 0	3

Задача Е. Проклятие Кассандры

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

Кассандра получила от Аполлона дар пророчества, но была проклята — никто не верил ее предсказаниям. Ее пророчества записывались как строки, и лишь некоторые фрагменты этих записей содержали истину.

Истинным считается такой фрагмент пророчества, где каждая буква встречается не более k раз. Найдите длину самого длинного истинного фрагмента в пророчестве Кассандры.

Формат входных данных

В первой строке входных данных содержится строка s ($1 \leq |s| \leq 2 \times 10^5$), содержащая только строчные латинские буквы. Во второй строке содержится целое число k ($1 \leq k \leq 2 \times 10^5$).

Формат выходных данных

Одно целое число — длина самой длинной подходящей подстроки.

Примеры

стандартный ввод	стандартный вывод
apollo 1	4
cassandra 2	8
prophecy 3	8

Замечание

Кассандра, называемая также Александра, — в древнегреческой мифологии троянская царевна, наделённая Аполлоном даром пророчества и предвидевшая гибель Трои. За отказ во взаимности Аполлону тот сделал так, что предсказаниям Кассандры никто не верил.

(с) Википедия

Задача F. Да где эта 105-я аудитория!?

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

Убереги вас всевышний от попадания в 4-й корпус. Аминь.

Вам поставили пару в 4-м корпусе Магического Политехнического — поздравляю! Это прекрасный день, чтобы остаться дома! Но вы уже пропустили одну пару на прошлой неделе (стыдно конечно), поэтому сегодня решили навестить преподавателя, надеясь, что он сам-то пришел.

Четвертый корпус — это, на самом деле, страшное с точки зрения пространственного восприятия реальности место, особенно для неподготовленного человека. Поэтому на входе вы взяли брошюрку с картой корпуса. Эта карта представляет из себя матрицу $n \times m$ ячеек. Каждая ячейка — это *направляющая движения* или по вертикали, или по горизонтали, то есть из каждой ячейки можно попасть на соседнюю с ней по стороне или по горизонтали, или по вертикали.

Вы поняли, что находитесь в ячейке в левом верхнем углу, а попасть нужно в правый нижний угол. Определите, за какое минимальное число перемещений можно попасть в аудиторию, находящуюся в правом нижнем углу. Одним перемещением считается переход на соседнюю ячейку.

Формат входных данных

В первой строке входных данных содержатся целые числа n, m ($1 \leq n, m \leq 10$) — число строк и столбцов матрицы. Далее следуют n строк, каждая длиной m и состоит только из символов «|» и «=». Где «|» обозначает, что из соответствующей ячейки можно попасть на другую ячейку в том же столбце, но строкой ниже или выше, «=» же означает, что из ячейки можно попасть на ячейку левее или правее.

Формат выходных данных

Выведите единственное целое число — минимальное число перемещений для попадания в правый нижний угол матрицы. Если в правый нижний угол нельзя попасть ни при каких условиях, выведите «-1».

Примеры

стандартный ввод	стандартный вывод
3 3 == = ==	6
3 5 == = == = =	8
3 3 = = = =	-1

Замечание

Fun fact: чтобы попасть в 105 аудиторию, сначала нужно подняться на второй этаж, пройти в кафедру, которая закрыта на дверь с кодовым замком, а потом опуститься на два этажа ниже, а сама аудитория находится на уровне первого этажа.

Задача G. Простой и ровный парень

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

Ваня — обычный *простой и ровный* парень, увлекающийся математикой. Ему очень нравится одноклассница Алена, но он никак не может найти к ней подход. Чтобы произвести впечатление, Ваня придумал необычный тест на *простоту и ровность*.

Он утверждает следующее: любое ровное (четное) число можно разложить в сумму двух простых чисел. «Смотри, — говорит он Алене, — возьмем число 10. Его можно собрать из простых чисел 3 и 7 либо из 5 и 5. Все сходится!»

Алена решила проверить Ваню — она дала ему список из m целых четных чисел. Помогите Ване: для каждого числа из списка Алены найдите хотя бы одно разложение этого числа в сумму двух простых чисел, если какое-то число нельзя разложить в сумму двух простых, то об этом тоже нужно сообщить.

Формат входных данных

В первой строке входных данных содержится одно целое число m ($1 \leq m \leq 10^5$) — количество чисел в списке Алены. В каждой из следующих m строк содержится по одному целому четному числу n ($2 \leq n \leq 20251206$).

Формат выходных данных

Необходимо дать ответ для каждого из m чисел. Если существует разложение $n = p + q$, где p и q — простые числа, выведите два таких числа p и q через пробел. Если возможных разложений несколько, выведите любое. Если не существует ни одного разложения, выведите «-1».

Примеры

стандартный ввод	стандартный вывод
1 10	3 7
3 12 14 16	5 7 3 11 3 13

Задача Н. Тэк, ну тут единички посчитать, да?

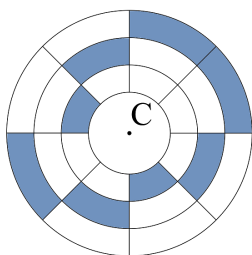
Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Отдыхая на балконе, вы нашли странную открытую шкатулку с кольцевым замком на дверце. Этот замок устроен следующим образом (вам будет проще понять, если вы сначала изучите иллюстрацию ниже):

- Есть три кольца с центром в одной общей точке C : внутреннее, среднее и внешнее. Каждое кольцо поделено на 8 равных сегментов — границы сегментов на соседних кольцах совпадают. Каждый из восьми сегментов на каждом из колец может быть или *открытым*, или *закрытым*.

Вы решили положить в шкатулку 55 рублей и закрыть ее до лучших времен. Однако, как оказалось, замок может быть ненадежным, и закрыть шкатулку не получится.

Замок шкатулки закрыт только тогда, когда центральную точку C внутри замка не видно снаружи замка: то есть нет ни одного луча с началом в центральной точке такого, что он проходит через все кольца, не попадая ни на границу *закрытого* сегмента, ни на сам *закрытый* сегмент.



Справа находится иллюстрация внешнего вида кольцевого замка и одновременно одного из вариантов решения в первом тестовом примере — нельзя увидеть точку, являющуюся центрами трех колец, если смотреть «снаружи» замка.

Вам дана конфигурация замка и разрешено поворачивать любые кольца на 45 градусов в любом направлении любое количество раз. Определите, возможно ли закрыть такой замок?

Формат входных данных

Входные данные в трех строках содержат три строки a, b, c длиной 8 — описания сегментов внутреннего, среднего и внешнего колец. Каждый символ строки может быть или «0», что означает *открытый* сегмент, или «1», что означает *закрытый* сегмент. Помните, что строки описывают кольца: например, сегмент, описываемый a_1 , является соседним с a_2 и a_8 .

Формат выходных данных

Выведите единственную строку «Yes», если можно прокрутить некоторые кольца замка так, чтобы закрыть его. Иначе выведите «No».

Примеры

стандартный ввод	стандартный вывод
00001001 00100101 00100110	Yes
00010010 00010000 00000011	No

Задача I. Манипуляция: искусство массового обмана.

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



Пример так называемой *оптической иллюзии* — если внимательно сфокусироваться на изображении слева, то оно может показаться некоторым людям цветным, но на самом деле это изображение черно-белое, как и весь остальной комплект задач.

Это пример достаточно известной иллюзии, которая «ломает» мозг. А какие головоломки любите вы? Впрочем, спор вокруг этого изображения не стихает уже достаточно долгое время, а точные причины такого поведения остаются неизученными.

Сыграем в игру. Есть три колпачка, выставленных в ряд. На протяжении всей игры они нумеруются так: самый левый с номером 1, по середине с номером 2, а справа с номером 3. Под один из них с номером x я кладу шарик.

А далее сообщаю вам n последовательных действий. Каждое действие описывается парой i, j , обозначающей, что я меняю местами колпачок i с колпачком j — после такой операции на месте i -го колпачка окажется j -й и теперь у него будет номер i , и наоборот, если $i = j$, то ничего не происходит.

Что? Определить, где колпачок будет в конце? Ха-ха:) Впрочем, да. Вы правы. Просто скажите номер колпачка, где будет шарик после последовательного выполнения всех действий.

Формат входных данных

В первой строке входных данных содержатся два целых числа: x ($1 \leq x \leq 3$) — номер колпачка, куда я кладу шарик, и n ($1 \leq n \leq 100$) — количество действий.

Далее следует строка s длиной $n + 1$ символ. Каждая k -я операция описывается парой колпачков с номерами $i = s_k$ и $j = s_{k+1}$.

Формат выходных данных

Выведите единственное целое число — номер колпачка, под которым в конце игры окажется шарик.

Примеры

стандартный ввод	стандартный вывод
1 1 12	2
2 3 1323	2
3 5 123213	1

Замечание

Рано или поздно любопытство возьмет верх, и вы перевернете эту страницу, чтобы посмотреть, что на обратной стороне. Но на обратной стороне ничего не будет. Ни мема, ни-че-го.

Или не перевернете?

Не пытайтесь жульничать или выносить страницу на свет. Если на обратной стороне что-то и есть, то написано специальными светопропускающими чернилами. Все продумано...

Задача J. ПОСВЯЯЯТ

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

За окном — начало декабря. Воздух свеж, а землю укрывает пушистый снег. В это время года у первокурсников существует добрая традиция — празднование посвящения в студенты. В этом году праздник, как и всегда, проходил в Международном культурном центре.

Атмосфера была наполнена смехом и радостью, однако для службы охраны это было время повышенной ответственности. Все посетители мероприятия должны были пройти обязательный досмотр. Среди гостей были как типичные студенты с рюкзаками, набитыми конспектами и ноутбуками, так и более празднично настроенные гости с элегантными сумками.

На мероприятие в МКЦ ТПУ пришло p человек с рюкзаками и b человек с сумками. Для обеспечения безопасности все рюкзаки и сумки должны быть проверены охраной. В распоряжении охраны находится **ровно два охранника**, каждый из которых может проверять только одну вещь за раз. На проверку одного рюкзака каждому охраннику требуется t_p секунд, а одной сумки — t_b секунд.

Все охранники начинают работу одновременно в момент времени $t = 0$ и могут работать параллельно, распределяя вещи между собой произвольным образом. Одну вещь может проверять только один охранник. Один охранник, начав проверку вещи, доводит дело до конца, не отвлекаясь и не передавая ее другому охраннику или откладывая проверку.

Необходимо найти минимальное время T (в секундах), за которое все p рюкзаков и b сумок будут проверены, чтобы ни один студент не пропустил начало торжественной части из-за очереди.

Формат входных данных

В первой и единственной строке записаны четыре целых числа: p, b, t_p, t_b ($0 \leq p, b \leq 10^5$), ($1 \leq t_p, t_b \leq 10^9$) — количество участников с рюкзаками, количество участников с сумками, время проверки одного рюкзака и время проверки одной сумки соответственно.

Формат выходных данных

Требуется вывести одно целое число — минимальное время T , за которое все вещи могут быть проверены.

Примеры

стандартный ввод	стандартный вывод
1 1 20 10	20
1 4 100 26	104

Задача К. Круг подозрений

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

В элитном закрытом клубе произошло ограбление. Сыщик допросил всех членов клуба и выяснил, что каждый из них следил за кем-то другим в ночь преступления. Причем для каждого человека точно известно, за кем он следил.

Сыщик подозревает, что преступник был не один — это были три члена клуба, где каждый «прикрывал» следующего, а именно: A следил за B , B следил за C , C следил за A , где A, B, C — различные члены клуба.

Сейчас нас интересуют циклы именно из **трех** человек. Помогите сыщику и скажите, есть ли вообще хотя бы один цикл длины 3.

Формат входных данных

Первая строка содержит целое число n ($3 \leq n \leq 5000$) — количество подозреваемых. Следующие n строк описывают матрицу слежки a : i -я строка матрицы a содержит строку из n символов, символ j в этой строке равен «1», если подозреваемый i следил за подозреваемым j , иначе равен «0».

На главной диагонали всегда стоят «0» (никто не следил за самим собой).

Формат выходных данных

Если существует хотя бы один цикл длины **ровно 3**, выведите «Yes», иначе выведите «No».

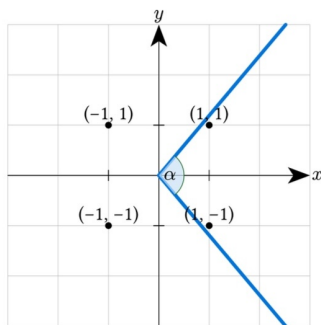
Примеры

стандартный ввод	стандартный вывод
3 010 001 100	YES
5 01000 00100 00010 00001 10000	NO
3 010 101 110	YES

Задача L. Мы следим!

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

Добро пожаловать на олимпиаду *Hack IT!* Чтобы обеспечить честность, организаторы установили наблюдательную камеру в самой большой аудитории проведения олимпиады. Можете не искать ее — она хорошо замаскирована. Предлагаем вам представить себя на месте организаторов и поработать с настройкой камеры.



Перед началом олимпиады участники рассаживаются по местам — их позиции уже фиксированы.

Камера закреплена точно в центре помещения в точке с координатами $(0, 0)$. Камера имеет угол обзора R° , и ее дальность действия считается бесконечной.

Камеру можно повернуть в любую сторону: можно выбирать направление α (в градусах), и тогда она видит всех студентов, находящихся внутри сектора шириной R° , симметричного относительно луча под углом α .

Точнее, участник в точке $(x, y) \neq (0, 0)$ виден, если угол θ между положительным направлением оси Ox и вектором (x, y) удовлетворяет условию:

$$\min(|\theta - \alpha|, 360 - |\theta - \alpha|) \leq \frac{R}{2}.$$

Вам предстоит определить, какое максимальное количество участников можно одновременно зафиксировать в кадре, выбрав оптимальное направление α .

Формат входных данных

В первой строке входных данных содержатся два целых числа n и R ($1 \leq n \leq 10^5$; $0 < R < 360$) — количество участников и угол обзора камеры соответственно. Далее следуют n строк, каждая i -я строка содержит два целых числа x_i и y_i ($-10^4 \leq x_i, y_i \leq 10^4$) — координаты, в которых находится i -й участник. Гарантируется, что ни один участник не находится в точке $(0, 0)$, и все координаты участников попарно различны.

Формат выходных данных

Выведите одно целое число — максимальное количество участников, которых можно одновременно увидеть камерой при оптимальном выборе направления обзора.

Примеры

стандартный ввод	стандартный вывод
4 100 1 1 -1 1 -1 -1 1 -1	2
3 1 1 1 2 2 3 3	3

Замечание

При использовании вещественных чисел в вычислениях рекомендуется использование максимальной доступной точности.