

Задача А. Планировщик для робота

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

В каждую из n минут роботу назначается очередное задание. На выполнение любого задания у него уходит две минуты. Одновременно выполнять несколько заданий нельзя, и после того, как робот начал выполнять задание, прервать его выполнение уже нельзя.

Задания бывают разной срочности, и, чем выше срочность задания, тем раньше необходимо начать его выполнение. Поэтому как только робот завершает выполнение очередного задания, он берется за задание с максимальной срочностью.

Определите для каждого из n заданий номер минуты, в которую робот закончит его выполнение.

Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 50000$) — количество заданий, данных роботу. Во второй строке через пробел указана срочность выполнения задания a_i ($1 \leq a_i \leq 10^9$), данного роботу в минуту номер i — чем она выше, тем раньше необходимо выполнить задание. Гарантируется, что все a_i -е различны.

Формат выходных данных

Выведите через пробел n натуральных чисел b_i — номера минут, в которые i -е задание будет выполнено.

Пример

стандартный ввод	стандартный вывод
4 7 3 9 2	3 7 5 9

Замечание

События в примере будут развиваться следующим образом:

В первую минуту дается первое задание и робот сразу же начинает его выполнять.

Во вторую минуту дается второе задание. Продолжается выполнение первого задания.

В третью минуту дается третье задание. Завершается выполнение первого задания. Из двух заданий, ожидающих выполнения, более высокий приоритет у третьего, и робот начинает его выполнять.

В четвертую минуту поступает последнее задание. Продолжается выполнение задания номер 3.

В пятую минуту его выполнение завершается. Из оставшихся двух заданий более высокий приоритет у того, которое поступило во вторую минуту — робот начинает его выполнять.

В шестую минуту продолжается выполнение второго задания.

В седьмую минуту завершается выполнение второго задания. Робот начинает выполнять последнее задание, которое будет выполнено в минуту номер 9.

Задача В. Бумажный дом

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

Даша сделала бумажный домик, состоящий из двух компонент: — основания в форме квадрата с длиной диагонали a и крыши, являющейся равнобедренным прямоугольным треугольником с длиной катета b .

Во время сильного ветра домик сдуло, после чего он приземлился на землю так, как показано на рисунке ниже. Вычислите координату самой высокой точки перевернутого бумажного домика. Известно, что дом симметричен относительно прямой $y = x$.

Формат входных данных

В первой строке дано натуральное число a , во второй — натуральное число b ($1 \leq a, b \leq 10^9$).

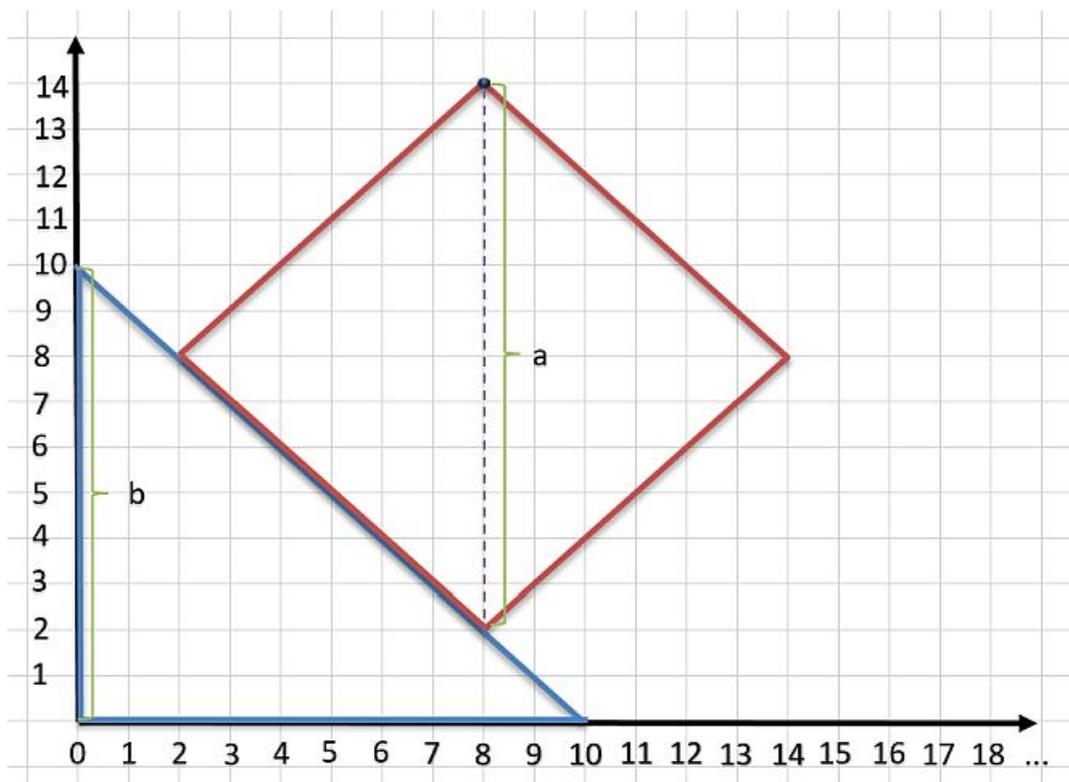
Формат выходных данных

Выведите через пробел координаты искомой точки. Гарантируется, что такая точка одна и ее координаты целочисленные.

Пример

стандартный ввод	стандартный вывод
12	8 14
10	

Замечание



Задача С. Необычная игра

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

Петр с друзьями играют в необычную модификацию игры «Шляпа».

Друзья разбиваются на две команды. Суть игры заключается в том, что капитаны по очереди пытаются объяснить своей команде загаданное слово, не используя в речи загаданное слово. Слово можно объяснять разными способами, поэтому за один такой ход команда может увеличить свой счет на 1, 2 или 3 очка или не увеличить вовсе (если команда не смогла отгадать слово).

Друзья серьезно относятся к игре, поэтому внимательно поддерживают счет, так же как, например, в футболе. До начала игры счет был $0 : 0$.

Будем называть «историей игры» последовательность счетов между теми ходами, которые *както изменяли счет*. Например, история игры может быть такой: $0 : 0, 1 : 0, 1 : 2, 4 : 2, 5 : 2$, но не может быть $0 : 0, 0 : 2, 0 : 2, 1 : 2$ или $0 : 0, 0 : 1, 2 : 2$.

Витя пока не опытный игрок, поэтому в игре не участвовал, однако он запомнил, что в какой-то момент счет был $a : b$. Также известно, что игра закончилась со счетом $c : d$.

После игры Виктор задумался, сколько существует «историй игры», в которых:

1. Начальный счет — $0 : 0$;
2. Промежуточный счет — $a : b$;
3. Окончательный счет — $c : d$.

Помогите мальчику ответить на этот вопрос!

Формат входных данных

В первой строке входного файла через пробел записаны два неотрицательных целых числа: a и b ($0 \leq a, b \leq 20$) — промежуточный счет, который запомнил Виктор. Во второй строке входного файла через пробел записаны два неотрицательных целых числа: c и d ($0 \leq c, d \leq 20$) — окончательный счет игры. Также гарантируется, что $a \leq c$ и $b \leq d$.

Формат выходных данных

Выведите одно натуральное число — количество различных историй игры. Гарантируется, что ответ на задачу при любых входных данных не превосходит 10^{18} .

Примеры

стандартный ввод	стандартный вывод
0 0 0 0	1
0 0 3 1	12
2 0 3 1	4

Замечание

Иллюстрация ко второму примеру:

Задача D. Никита и очки

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

С приближением лета Никита заинтересовался тем, как делают солнцезащитные очки. Чтобы лучше разобраться в этом вопросе, мальчик решил сделать оправы для таких очков из бумаги (стеклами и дужками для очков он хочет заняться позже).

Никита решил вырезать оправы из бумаги необычным способом. Пусть есть прямоугольная заготовка размером $a \times b$. Оправа представляет собой рамку шириной 1 с двумя одинаковыми прорезями наибольшей площади, расстояние между которыми равно 1 при нечетной ширине заготовки или 2 при четной ширине.

Когда Никита вырезает из бумажной заготовки два прямоугольника, у него в руках оказываются еще две заготовки, из которых (если размеры позволяют) можно сделать новые оправы меньшего размера! В конце концов у мальчика останутся маленькие заготовки, которые уже ни на что не годны. Никита, прежде чем придумать, как их ещё можно использовать, хочет узнать их минимально возможную общую площадь.

Формат входных данных

Единственная строка входного файла содержит два натуральных числа, записанных через пробел: a и b ($1 \leq a, b \leq 10^4$) — размеры заготовки.

Формат выходных данных

Выведите одно натуральное число — ответ на вопрос задачи.

Примеры

стандартный ввод	стандартный вывод
3 5	2
3 4	12
10 24	8

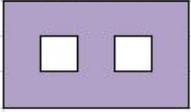
Замечание

В первом примере рассмотрен случай минимально возможной заготовки, из которой можно сделать оправу с двумя единичными по площади отверстиями.

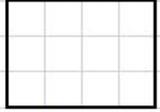
Во втором примере заготовка сразу идет в отходы, её размеры слишком малы.

Во третьем примере Никита может добиться такого выбора направления оправ, чтобы минимизировать общую площадь отходов до 8. При других вариантах его работы ответ будет существенно больше.

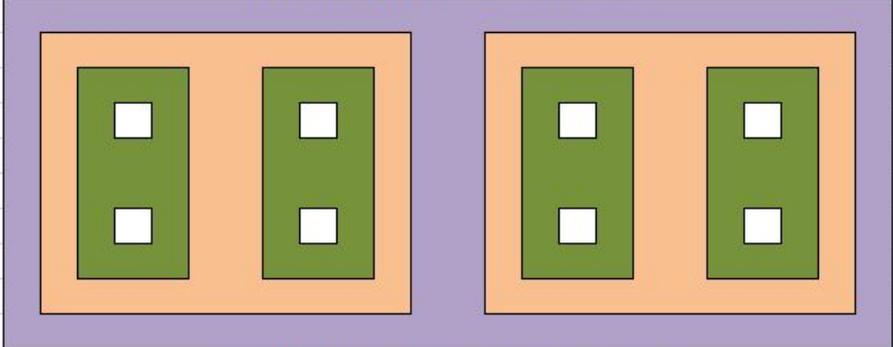
пример 1



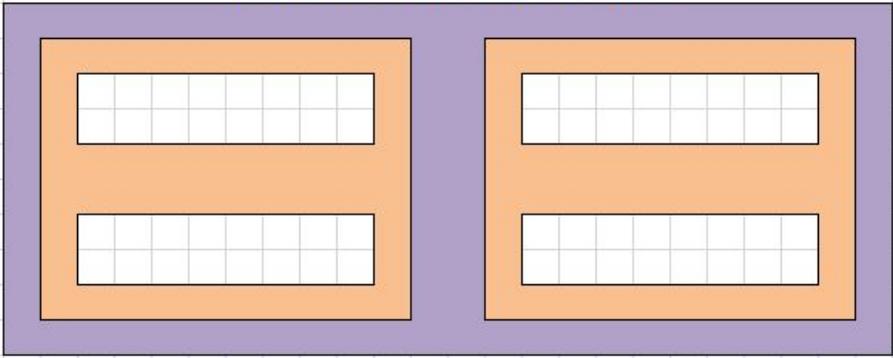
пример 2



пример 3, эффективное построение



пример 3, неэффективное построение



Задача Е. Странные шашки

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

Маленький Петя очень любит играть в шашки, поэтому однажды задумал сделать свою собственную доску для шашек — на такой играть с друзьями куда интереснее!

Изготовив саму доску, Петя принялся за раскраску. Спустя несколько часов, когда краска уже успела засохнуть, он показал творение папе и мгновенно расстроился: как оказалось, раскраска получилась с ошибками. Как всем известно, черные и белые клетки на доске должны чередоваться, то есть любые две черные клетки и любые две белые клетки не должны иметь общей стороны. К сожалению, у Пети это правило выполнялось не всегда.

Отдирать засохшую краску — дело неприятное, поэтому Петя просит вас разработать алгоритм исправления раскраски, при котором потребуется покрасить заново наименьшее число клеток.

Формат входных данных

В первой строке даны два натуральных числа N и M ($1 \leq N, M \leq 100$).

Далее в N строках задается текущая раскраска доски, в каждой строке есть M символов. Символ с номером j в строке i означает, в какой цвет Петя покрасил клетку за номером j в строке с номером i : символ B означает черный цвет, W — белый.

Формат выходных данных

В первой строке выведите число t — минимальное число клеток, которые придется перекрасить Пете.

Далее в t строках выведите координаты клеток, разделенные пробелом, цвета которых придется поменять. Координата задается двумя числами i и j , разделенными пробелом ($1 \leq i \leq N, 1 \leq j \leq M$).

Примеры

стандартный ввод	стандартный вывод
4 4 WWWW WWWW WWWW WWWW	8 1 2 1 4 2 1 2 3 3 2 3 4 4 1 4 3
3 3 BWB WBW BWB	0

Задача F. Правильный боулинг

Имя входного файла: `stdin`
 Имя выходного файла: `stdout`
 Ограничение по времени: 1 second
 Ограничение по памяти: 64 megabytes

Вова и Катя придумали новый способ играть в боулинг. Основное отличие заключается в том, что кегли нужно выставлять не в ряд, а так чтобы они образовывали правильный многоугольник, находясь только в его вершинах. Сторона многоугольника при этом может иметь произвольную длину.

Однажды Вова в очередной раз расставил кегли в виде какого-то правильного многоугольника. Однако сбить все кегли за раз у него не получилось: три кегли остались стоять в тех же точках, где стояли.

После этого пришла Катя и задумалась, какую наименьшую площадь мог иметь многоугольник, который образовывали кегли до того, как Вова сбил все, кроме трех оставшихся? Конечно же, Катя смогла ответить на этот вопрос, а сможете ли вы?

Формат входных данных

Входной файл состоит из трех строк, каждая из которых содержит пару чисел — координаты столба. Любая из координат не превосходит по абсолютной величине 1000, и задана с 6 знаками после десятичной точки.

Формат выходных данных

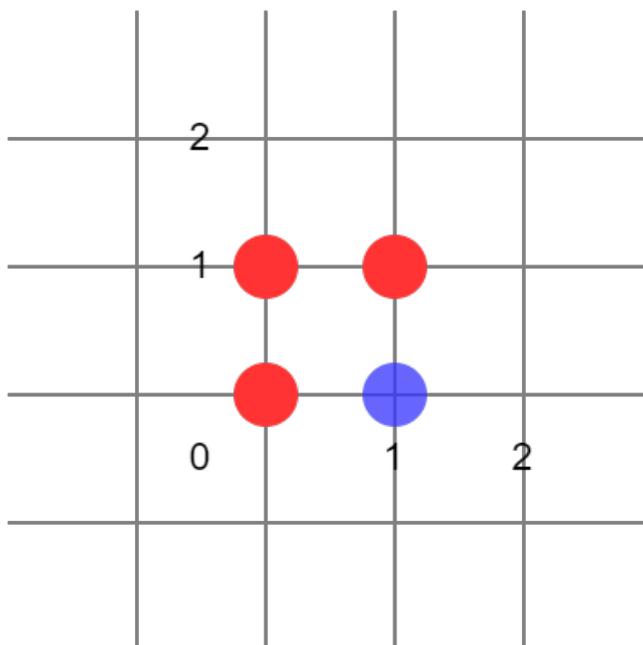
Выведите искомую площадь с точностью не менее 6 знаков после точки. Гарантируется, в оптимальном многоугольнике будет не более 100 вершин.

Примеры

stdin	stdout
0.000000 0.000000	1.00000000
1.000000 1.000000	
0.000000 1.000000	

Замечание

Ниже дана иллюстрация к примеру из условия. Красным обозначены точки, в которых остались кегли. Многоугольник, который образовывали кегли, — квадрат. Голубым цветом показана кегля, которая была сбита.



Задача G. Эрудит

Имя входного файла: `stdin`
 Имя выходного файла: `stdout`
 Ограничение по времени: 1 second
 Ограничение по памяти: 256 megabytes

Дима и Маша играли в игру **Эрудит**. Игра заключается в том, чтобы из карточек с буквами составлять слова.

К сожалению, их игре помешал кот и после того, как друзья выложили свои первые слова, он махнул хвостом и все карточки с буквами оказались на полу.

Маша и Дима собрали карточки с пола в кучу, однако теперь они боятся, что в куче оказались лишние буквы или, наоборот, какие-то карточки потерялись. Помогите ребятам проверить, можно ли из найденных карточек собрать исходные слова так, чтобы лишних букв **не осталось**.

Формат входных данных

В первой строке входных данных дано слово, которое составил Дима, во второй — слово, которое составила Маша. В третьей строке даны буквы которые нашли ребята под столом. Все строки не пусты и состоят только из заглавных латинских букв. Длина каждой строки не превосходит 100.

Формат выходных данных

Выведите «YES» без кавычек, если из найденных букв можно составить исходные слова и лишних букв не останется, и «NO» без кавычек в противном случае.

Примеры

<code>stdin</code>	<code>stdout</code>
DIMAMASHA MASHADIMA DIMADIMAMASHAMASHA	YES
IGRA ERUDIT IARETIDURB	NO
HIMYNAMEIS DIMA AMIDSIEMANYMIHX	NO

Замечание

В первом примере: из букв, записанных в последней строке можно составить слова, записанные в первых двух, и не останется лишних букв.

Во втором примере: в кучке не хватает буквы «G» и лишняя буква «B».

В третьем примере: лишняя буква «X».

Задача Н. Красная шапочка и волк

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	1 second
Ограничение по памяти:	256 megabytes

Красная шапочка отправилась в гости к бабушке. Идти девочке предстоит через лес, на входе в который в норе живет волк.

Девочка ходит со скоростью v_p километров в час, волк бежит со скоростью v_d километров в час. Волк обнаружит присутствие девочки в лесу через t часов после того, как она зайдет в лес, и немедленно бросится в погоню.

От начала леса до дома бабушки c километров. Если волк догонит красную шапочку, то он ее съест. Однако девочка может спастись, если отдаст волку один пирожок. Тогда волк остановится, возьмет пирожок, вернется к себе в нору (в начало леса), потратит f часов в норе на поедание пирога, после чего вновь отправится в погоню, начиная все с самого начала.

Предполагая, что красная шапочка будет идти по прямой без остановок, определите, сколько пирожков ей нужно взять с собой, чтобы успеть добраться до дома бабушки. Если волк догонит девочку в тот момент, когда она добежит до бабушки, считается, что она успела раньше (дополнительный пирожок не нужен).

Формат входных данных

Входные данные содержат пять целых чисел v_p, v_d, t, f и c ($1 \leq v_p, v_d \leq 100$; $1 \leq t, f \leq 10$; $1 \leq c \leq 1000$) — скорость красной шапочки, скорость волка, время, через которое волк обнаружит девочку, время, которое волк тратит на поедание пирога в норе, расстояние от начала леса до дома бабушки.

Каждое число записано в отдельной строке.

Формат выходных данных

Выведите минимальное количество пирожков, необходимых для того, чтобы успеть добежать до дома бабушки.

Примеры

<code>stdin</code>	<code>stdout</code>
1 2 1 1 10	2
1 2 1 1 8	1

Замечание

В первом примере через час после входа в лес девочка будет находиться на расстоянии 1 от пещеры, а волк обнаружит девочку. Через 2 часа после начала движения девочки волк догонит ее на расстоянии 2 от своей норы, и красной шапочке нужно будет отдать первый пирожок волку. Возвращение в нору и поедание пирога займет у волка еще два часа; за это время девочка уйдет на расстояние 4 от начала леса. Во второй раз волк догонит девочку на расстоянии 8 от норы, и ей понадобится второй пирожок, после чего она спокойно дойдет до бабушки.

Второй пример аналогичен первому, но второй раз волк догонит красную шапочку ровно в тот момент, когда она дойдет до бабушки, и второй пирожок не понадобится.

Задача I. Надежный код

Имя входного файла: `stdin`
 Имя выходного файла: `stdout`
 Ограничение по времени: 1 second
 Ограничение по памяти: 256 megabytes

С приходом весны мальчик Миша достал с балкона велосипед, чтобы кататься на нем все лето. Как известно, оставлять велосипед без присмотра на улице — опасно, поэтому мальчик также достал специальный трос с кодовым замком. Теперь он придумывает новый код для этого замка.

Код для замка всегда состоит из четного количества цифр. Будем говорить, что код замка не надежный, если каждой цифре его первой половины можно сопоставить некоторую цифру второй половины так, что каждая цифра из первой половины **строго меньше** сопоставленной цифры из второй половины, либо каждая цифра из первой половины **строго больше** сопоставленной цифры из второй половины. Каждую цифру в сравнениях нужно использовать в точности один раз. Другими словами, найдется такое взаимно-однозначное соответствие между цифрами первой и второй половины кода, что либо каждая цифра первой половины окажется **строго меньше** соответствующей ей цифры второй половины, либо каждая цифра первой половины окажется **строго больше** соответствующей ей цифры второй половины.

Например, код 2421 удовлетворяет данному критерию ненадежности (искомое соответствие $2 > 1$ и $4 > 2$), код 0135 также удовлетворяет критерию (искомое соответствие $0 < 3$ и $1 < 5$), а код 3754 критерию не удовлетворяет.

Миша придумал очередной код, состоящий из $2n$ цифр. Ваша задача — проверить, является ли он надежным.

Формат входных данных

В первой строке задано целое число n ($1 \leq n \leq 100$). Во второй строке содержится строка, состоящая из $2n$ цифр, задающая код.

Формат выходных данных

В первую строку выведите «YES», если код не является надежным, и «NO» в противном случае (без кавычек).

Примеры

<code>stdin</code>	<code>stdout</code>
2 3468	YES
2 1310	YES
2 1635	NO

Задача J. Разбиение на команды

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

Одноклассники очень любят играть в командные игры. Сегодня $2n$ школьников решили сыграть в футбол между двумя командами, в каждой из которых должно быть n игроков.

Загвоздка в том, что некоторые ребята не согласны играть в одной команде. Оказалось, что существует m пар одноклассников таких, что ребята в них не согласны играть в одной команде.

Школьники просят вас помочь с распределением на команды. Разделите одноклассников на две команды так, чтобы выполнить все m пожеланий, или скажите, что такого распределения не существует.

Формат входных данных

В первой строке входных данных дано два натуральных числа n, m ($1 \leq n \leq 50$; $0 \leq m \leq 100$) — количество ребят в команде и количество пар ребят, которых нельзя ставить в одну команду.

В следующих m строках даны пары чисел x_i, y_i ($1 \leq x_i, y_i \leq 2n$, $x_i \neq y_i$) — номера ребят, которых нельзя ставить в одну команду.

Формат выходных данных

Если распределение, удовлетворяющее всем пожеланиям, существует, выведите 2 строки, в каждой по n чисел через пробел — номера ребят, которых нужно поставить в первую и вторую команду соответственно.

Если решения нет, выведите «IMPOSSIBLE» (без кавычек).

Если задача имеет несколько решений, можно вывести любое.

Примеры

стандартный ввод	стандартный вывод
2 3 1 2 1 3 4 3	2 3 1 4
2 3 1 2 1 3 1 4	IMPOSSIBLE

Задача К. Сигнал из космоса

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

Леонид с детства интересуется космосом. В этот раз он заинтересовался тем, как информация от спутников попадает на Землю.

Мальчик захотел поймать сигналы, идущие от спутников. Для этого он решил построить большую антенну из алюминиевых трубок. Леонид считает, что антенна позволит поймать сигнал из космоса, если:

1. все ее трубки будут образовывать собой выпуклый многоугольник;
2. между любыми двумя соседними трубками будет прямой угол.

У мальчика есть 4 трубки длиной a , b , c и d сантиметров. Помогите Лене понять, можно ли из имеющихся у него трубок собрать антенну, которая позволит поймать сигнал из космоса.

Формат входных данных

В первой и единственной строке входных данных дано четыре натуральных числа a, b, c, d ($1 \leq a, b, c, d \leq 100$) — длины алюминиевых трубок, которые есть у Леонида.

Формат выходных данных

Выведите «YES» (без кавычек), если из трубок мальчика можно построить антенну, принимающую сигналы из космоса, и «NO» в противном случае.

Примеры

стандартный ввод	стандартный вывод
3 3 4 2	NO
3 3 4 4	YES