

Задача А. Автобусы

Имя входного файла: a.in
Имя выходного файла: a.out
Максимальное время работы на одном тесте: 1 секунда
Максимальный объем используемой памяти: 64 мегабайта

На кольцевом маршруте суммарной длиной L километров на равном расстоянии друг от друга расположены N остановок (пронумерованных от 1 до N). По этому маршруту движутся M автобусов с одинаковой скоростью v километров в час так, что интервал между двумя идущими друг за другом автобусами один и тот же для всех автобусов (интервалом между автобусами будем называть время, которое проходит между приездом на одну и ту же остановку двух идущих друг за другом автобусов). Автобусы пронумерованы числами от 1 до M . Движение происходит в направлении увеличения номеров остановок.

В некоторый момент времени на всем участке между остановками номер X и Y (не обязательно соседними) начался ремонт дороги, из-за чего скорость движения на этом участке стала w километров в час. Скорость на ремонтируемом участке может оказаться как меньше обычной, так и больше за счет регулировщиков на этом участке дороги. При этом автобусы продолжили движение по маршруту с максимально возможной скоростью (w на ремонтируемом участке и v на остальном). Однако из-за этого интервалы движения между автобусами перестали быть равными.

Если какой-нибудь автобус оказался между остановками X и Y в момент начала ремонта, то он мгновенно меняет свою скорость с v на w , и едет с этой скоростью на протяжении всего ремонтируемого участка. Миновав его, он опять начинает ехать с нормальной скоростью v .

Известно, что в тот момент, когда начался ремонт, автобус номер один находился на остановке номер 1. В тот момент, когда этот автобус в следующий раз оказался на остановке номер 1, на эту остановку пришел диспетчер и стал измерять интервалы между автобусами. Он записывал интервалы между двумя автобусами до тех пор, пока автобус номер один опять не оказался на остановке номер 1.

Напишите программу, которая по информации о параметрах маршрута и ремонтируемом участке определит максимальный из интервалов времени, записанных диспетчером.

Формат входных данных

В единственной строке входного файла через пробел записаны целые числа L, N, M, X, Y, v, w .
 $1 \leq L, M, v, w \leq 10^9, 2 \leq N \leq 10^9, 1 \leq X < Y \leq N$.

Формат выходных данных

Выведите одно число с точностью до пяти знаков после десятичной точки — время в часах, равное максимальному интервалу между двумя автобусами, записанному диспетчером.

Примеры

a.in	a.out
9 4 3 2 4 5 5	0.60000
16 4 2 1 2 5 4	1.80000
15 4 3 2 3 5 9	1.00000

Частичные ограничения

Первая группа состоит из тестов, в которых $v = w$.

Вторая группа состоит из тестов, в которых $M = 2$ (при этом v не обязательно равно w).

Задача В. Праздничные открытки

Имя входного файла: `b.in`
 Имя выходного файла: `b.out`
 Максимальное время работы на одном тесте: 1 секунда
 Максимальный объем используемой памяти: 64 мегабайта

Петя с Васей решили поздравить всех своих одноклассниц с Международным Женским Днем. Важной частью любого праздника являются открытки. Купив их достаточно, друзья сели писать пожелания. Подписанные открытки они складывали на специальный стол, расчерченный в квадратную клетку параллельно краям стола так, что длина и ширина его составляли N и M клеток соответственно. По удивительному совпадению каждая открытка была размером точь-в-точь с две клетки стола. Петя настоял на том, чтобы класть подписанные поздравления строго по линиям сетки — горизонтально или вертикально, накрывая одной открыткой ровно две клетки.

По окончании работы оказалось, что каждая клетка стола накрыта ровно двумя открытками — крайне неудобное расположение для того, чтобы их дарить. К счастью, рядом был еще один такой же стол, поэтому они решили переложить на него половину открыток так, чтобы остальные, оставаясь на своем месте, образовывали ровно один слой — не накладывались друг на друга и полностью покрывали бы стол. Чтобы не нарушать порядка, открытки надо доставать по одной, извлекать очередную разрешается только в случае, если хотя бы одна из ее половинок лежит сверху (то есть эту половинку не накрывает другая открытка).

Поскольку одноклассниц у Пети и Васи довольно много, они обратились за помощью к Вам. Напишите программу, которая подскажет, какие открытки извлекать и в какой последовательности, либо определит, что это невозможно.

Формат входных данных

В первой строке входного файла записаны два целых числа N и M ($1 \leq N, M \leq 700$) — длина и ширина стола. Гарантируется, что хотя бы одно из N, M четное. Будем считать, что все открытки занумерованы числами от 1 до NM . Следующие $2N$ строк содержат по M чисел: первые N строк описывают нижний слой, следующие N строк — верхний слой. Число k в i -й строке j -м столбце нижнего или верхнего слоя означает наличие на этой позиции одной из половинок открытки номер k .

Гарантируется, что входные данные корректны, то есть что каждое число 1 до NM встречается ровно два раза, и эти вхождения находятся на соседних позициях, при этом они могут находиться как в одном слое, так и в разных. Кроме того, если две открытки покрывают одни и те же клетки, то одна из них находится обеими половинками снизу, а другая — сверху.

Формат выходных данных

В выходной файл запишите единственное слово NO, если не существует способа извлечь половину открыток нужным образом. В противном случае в первую строку выведите YES, во вторую — последовательность из $NM/2$ номеров открыток, которые надо достать, в правильном порядке. У каждой из них на момент извлечения хотя бы одна из половинок должна находиться сверху. Если искомым последовательностей несколько, выведите любую из них.

Примеры

<code>b.in</code>	<code>b.out</code>
2 2 1 1 3 2 4 2 4 3	YES 4 2
2 3 1 1 4 2 3 4 2 6 5 3 6 5	YES 2 6 5

Частичные ограничения

Первая группа состоит из тестов, в которых произведение $NM \leq 24$.

Вторая группа состоит из тестов, в которых $N, M \leq 100$.

Задача С. Предпраздничная суета

Имя входного файла: c.in
Имя выходного файла: c.out
Максимальное время работы на одном тесте: 2 секунды
Максимальный объем используемой памяти: 64 мегабайта

Помимо открыток Петя и Вася решили устроить одноклассникам чаепитие и заразили своей идеей еще $K-2$ своих друзей. Они собрались вместе и выбрали в одном довольно известном супермаркете P тортиков. Настал черед рассчитываться за них.

В магазине есть N работающих касс, занумерованных числами от 1 до N . Про i -ю кассу известно, что кассиру требуется A_i единиц времени на обработку одного товара и B_i единиц времени для того, чтобы рассчитаться с покупателем. Обойдя все кассы, школьники посчитали, что на обслуживание покупателей, уже стоящих в i -ю кассу, уйдет T_i единиц времени.

Теперь Петя и Вася задались вопросом, в какие кассы надо встать им и их друзьям (в каждую из выбранных касс должен стоять хотя бы один из них, и каждый из них может стоять не более, чем в одну кассу, поэтому суммарно они могут стоять не более чем в K касс) и сколько тортиков каждый должен взять, чтобы последний из них вышел из магазина как можно раньше. Некоторые из ребят могут в кассу не стоять, а, отдав все тортики другим, выйти через специальный выход для тех, кто ничего не купил.

Напишите программу, которая определит это минимальное время.

Формат входных данных

В первой строке записано одно число N — количество касс в супермаркете ($1 \leq N \leq 100000$). В следующих N строках записано по три числа A_i, B_i, T_i ($0 \leq A_i, B_i, T_i \leq 100000$). В последней строке записаны два числа — K и P — число школьников и покупок у них соответственно ($0 \leq P \leq 100000, 2 \leq K \leq 100000$).

Все числа во входном файле целые.

Формат выходных данных

Выведите минимальное время выхода последнего школьника из магазина.

Примеры

c.in	c.out	Комментарии
2 100 10 40 10 100 50 2 2	160	Здесь лучше всего встать в обе кассы и купить там по одному тортику.
3 1 2 0 5 2 1 2 10 1 3 5	7	Выгоднее всего одному из школьников встать со всеми тортиками в первую кассу, а остальным выйти без покупок.

Частичные ограничения

Первая группа состоит из тестов, в которых $N \leq 10$.

Вторая группа состоит из тестов, в которых $N \leq K \leq 100000$.

Задача D. Сообщающиеся сосуды

Имя входного файла: d.in
 Имя выходного файла: d.out
 Максимальное время работы на одном тесте: 2 секунды
 Максимальный объем используемой памяти: 64 мегабайта

Сегодня на уроке физики рассказывали удивительные вещи. Придя домой, Витя решил проверить слова учителя о том, что если взять два одинаковых сосуда, соединенных тонкой трубкой на уровне основания, то уровень жидкости при любом ее количестве также будет одинаковым для обоих сосудов.

Способ убедиться в правильности утверждения Витя избрал довольно оригинальный. Он взял аквариум с основанием длиной N и шириной 1, очень высокими стенками, и поставил $N - 1$ перегородку параллельно узкой боковой стенке аквариума, тем самым, разделив аквариум на N одинаковых отсеков. Каждая перегородка имеет ширину 1 и очень большую высоту. Толщиной перегородки можно пренебречь. В каждой из перегородок есть точечное отверстие на высоте H_i , диаметром которого также можно пренебречь. После всех этих приготовлений Витя медленно наливает в первый отсек (между стенкой и 1ой перегородкой) C литров воды. В часть аквариума размером $1 \times 1 \times 1$ вмещается ровно один литр воды. Так как стенки и перегородки в аквариуме были очень высокими, то через край вода не переливалась. После установления стационарного состояния он замерил уровень жидкости в каждом из N сосудов.

Теперь он хочет убедиться, что его экспериментальные данные не опровергают законы, рассказанные на уроке. Он обратился к вам с просьбой выяснить, какой должна быть высота жидкости в каждом из сосудов с теоретической точки зрения.

Рассмотрим подробно случай $N = 3$. Пусть сначала $H_1 < H_2$. Как только жидкость в первом отсеке достигнет уровня первого отверстия, вода станет поступать во второй отсек до тех пор, пока уровни в обоих отсеках не сравняются (или уровень воды в первом отсеке окажется равным H_1 , тогда во втором отсеке он будет на уровне $C - H_1$). Далее уровень жидкости в первых двух частях будет увеличиваться равномерно (или не будет меняться). Как только вода достигнет второго отверстия, вся она будет поступать в третий отсек, опять же до тех пор, пока уровни жидкости во всех трех частях не сравняются или вода в первых двух отсеках достигнет уровня H_2 . После этого, если воды оказалось достаточно, весь аквариум будет заполняться равномерно.

Пусть теперь $H_1 > H_2$. Как только жидкость в первом отсеке достигнет уровня первого отверстия, вся вода станет поступать во второй отсек. Если после этого уровень во втором отсеке сравняется с уровнем второго отверстия, то вода станет выливаться в третий до тех пор, пока высоты жидкостей во втором и третьем отсеках не станут равными. Далее уровень воды в них будет равномерно увеличиваться, пока не достигнет первого отверстия. После этого весь аквариум будет заполняться равномерно.

Формат входных данных

В первой строке записаны целые N и C ($1 \leq N \leq 100000$, $0 \leq C \leq 2 \cdot 10^9$). В следующих $N - 1$ строках содержится по одному целому числу H_i ($0 \leq H_i \leq 2 \cdot 10^9$), обозначающему высоту отверстия в i -й перегородке.

Формат выходных данных

Выведите N чисел, каждое на новой строке, с точностью до шести знаков после десятичной точки —уровень жидкости в 1, 2, ..., N отсеке соответственно.

Примеры

d.in	d.out
4 4 3 2 1	3 1 0 0
4 10 1 2 3	3 3 3 1

Частичные ограничения

Первая группа состоит из тестов, в которых $N \leq 100$.

Вторая группа состоит из тестов, в которых $N \leq 10000$.