

Задача А. Ярый коллекционер бабочек

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	64 мегабайта

Как известно, Андрей Сергеевич — ярый коллекционер бабочек. Он имеет огромную коллекцию, экспонаты которой собраны со всего мира. Будем считать, что в мире существует 2 000 000 000 видов бабочек.

Чтобы не запутаться, Андрей Сергеевич присвоил каждому виду уникальный номер. Нумерация видов бабочек начинается с единицы.

Теперь он хочет знать, есть ли бабочка с видом K в его коллекции, или же её придётся добывать, затрачивая уйму сил и денег.

Формат входных данных

В первой строке входного файла содержится единственное число N ($1 \leq N \leq 100\,000$) — количество видов бабочек в коллекции Андрея Сергеевича.

В следующей строке через пробел находятся N упорядоченных по возрастанию чисел — номера видов бабочек в коллекции.

Все виды бабочек в коллекции имеют различные номера.

В третьей строке файла записано число M ($1 \leq M \leq 100\,000$) — количество видов бабочек, про которых Андрей Сергеевич хочет узнать, есть ли они у него в коллекции или же нет. В последней строке входного файла содержатся через пробел M чисел — номера видов бабочек, наличие которых необходимо проверить.

Формат выходных данных

Выходной файл должен содержать M строчек. Для каждого запроса выведите “YES”, если бабочка с данным номером содержится в коллекции, и “NO” — в противном случае.

Примеры

стандартный ввод	стандартный вывод
7	YES
23 27 36 45 54 72 98	YES
4	YES
54 36 23 45	YES

Задача В. Мутанты

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	128 мегабайт

Уже долгое время в Институте Искусств, Мутантов и Информационных Технологий разводят милых разноцветных зверюшек. Для удобства каждый цвет обозначен своим номером, всего цветов не более 10^9 . В один из прекрасных дней в питомнике случилось чудо: все зверюшки выстроились в ряд в порядке возрастания цветов. Пользуясь случаем, лаборанты решили посчитать, сколько зверюшек разных цветов живет в питомнике, и, по закону жанра, попросили вас написать программу, которая поможет им в решении этой нелегкой задачи.

Формат входных данных

В первой строке входного файла содержится единственное число N ($0 \leq N \leq 10^5$) — количество зверюшек в Институте. В следующей строке находятся N упорядоченных по неубыванию неотрицательных целых чисел, не превосходящих 10^9 и разделенных пробелами — их цвета. В третьей строке файла записано число M ($1 \leq M \leq 100\,000$) — количество запросов вашей программе, в следующей строке через пробел записаны M целых неотрицательных чисел (не превышающих $10^9 + 1$).

Формат выходных данных

Выходной файл должен содержать M строчек. Для каждого запроса выведите число зверюшек заданного цвета в питомнике.

Примеры

стандартный ввод	стандартный вывод
10	1
1 1 3 3 5 7 9 18 18 57	2
5	1
57 3 9 1 179	2
	0

Замечание

Запрещено использовать встроенные структуры данных.

Задача С. Приближенный двоичный поиск

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 6 секунды
Ограничение по памяти: 64 мегабайта

Формат входных данных

В первой строке входных данных содержатся числа N и K ($1 \leq N, K \leq 100\,000$). Во второй строке задаются N чисел первого массива, отсортированного по возрастанию, а в третьей строке — K чисел второго массива. Каждое число в обоих массивах по модулю не превосходит 10^9 .

Формат выходных данных

Для каждого из K чисел выведите в отдельную строку число из первого массива, наиболее близкое к данному. Если таких несколько, выведите меньшее из них.

Примеры

stdin	stdout
5 5	1
1 3 5 7 9	3
2 4 8 1 6	7
	1
	5

Замечание

В этой задаче нельзя пользоваться встроенной сортировкой.

Задача D. Вербочки

Имя входного файла: `ropes.in`
Имя выходного файла: `ropes.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

С утра шел дождь, и ничего не предвещало беды. Но к обеду выглянуло солнце, и в лагерь заглянула СЭС. Пройдя по всем домикам и корпусам, СЭС вынесла следующий вердикт: бельевые веревки в жилых домиках не удовлетворяют нормам СЭС. Как выяснилось, в каждом домике должно быть ровно по одной бельевой веревке, и все веревки должны иметь одинаковую длину. В лагере имеется N бельевых веревок и K домиков. Чтобы лагерь не закрыли, требуется так нарезать данные веревки, чтобы среди получившихся веревочек было K одинаковой длины. Размер штрафа обратно пропорционален длине бельевых веревок, которые будут развешены в домиках. Поэтому начальство лагеря стремится максимизировать длину этих веревочек.

Формат входных данных

В первой строке заданы два числа — N ($1 \leq N \leq 10001$) и K ($1 \leq K \leq 10001$). Далее в каждой из последующих N строк записано по одному числу — длине очередной бельевой веревки. Длина веревки задана в сантиметрах. Все длины лежат в интервале от 1 сантиметра до 100 километров включительно.

Формат выходных данных

В выходной файл следует вывести одно целое число — максимальную длину веревочек, удовлетворяющую условию, в сантиметрах. В случае, если лагерь закроют, выведите 0.

Примеры

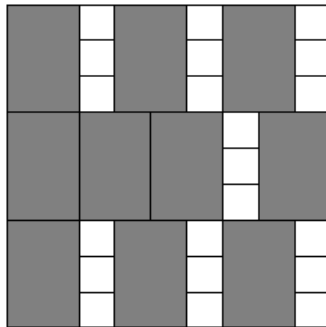
<code>ropes.in</code>	<code>ropes.out</code>
4 11 802 743 457 539	200

Задача Е. Дипломы

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

Когда Петя учился в школе, он часто участвовал в олимпиадах по информатике, математике и физике. Так как он был достаточно способным мальчиком и усердно учился, то на многих из этих олимпиад он получал дипломы. К окончанию школы у него накопилось n дипломов, причём, как оказалось, все они имели одинаковые размеры: w — в ширину и h — в высоту.

Сейчас Петя учится в одном из лучших российских университетов и живёт в общежитии со своими одногруппниками. Он решил украсить свою комнату, повесив на одну из стен свои дипломы за школьные олимпиады. Так как к бетонной стене прикрепить дипломы достаточно трудно, то он решил купить специальную доску из пробкового дерева, чтобы прикрепить её к стене, а к ней — дипломы. Для того чтобы эта конструкция выглядела более красиво, Петя хочет, чтобы доска была квадратной и занимала как можно меньше места на стене. Каждый диплом должен быть размещён строго в прямоугольнике размером w на h . Дипломы запрещается поворачивать на 90 градусов. Прямоугольники, соответствующие различным дипломам, не должны иметь общих внутренних точек.



Требуется написать программу, которая вычислит минимальный размер стороны доски, которая потребуется Пете для размещения всех своих дипломов.

Формат входных данных

Входной файл содержит три целых числа: w, h, n ($1 \leq w, h, n \leq 10^9$).

Формат выходных данных

В выходной файл необходимо вывести ответ на поставленную задачу.

Примеры

<code>diplomas.in</code>	<code>diplomas.out</code>
2 3 10	9

Задача F. Квадратный корень и квадратный квадрат

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Найдите такое число x , что $x^2 + \sqrt{x} = C$, с точностью не менее 6 знаков после точки.

Формат входных данных

В единственной строке содержится вещественное число $1.0 \leq C \leq 10^{10}$.

Формат выходных данных

Выведите одно число — искомый x .

Примеры

stdin	stdout
2.0000000000	1.0000000000
18.0000000000	4.0000000000

Задача G. Отрезок с небольшой суммой

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

Дан массив из n неотрицательных чисел a_i . Скажем, что отрезок этого массива $a[l..r]$ ($1 \leq l \leq r \leq n$) *хороший*, если сумма чисел на этом отрезке не больше s . Ваша задача — найти самый длинный хороший отрезок.

Формат входных данных

Первая строка содержит целые числа n и s ($1 \leq n \leq 10^5$, $1 \leq s \leq 10^{14}$). Вторая строка содержит числа a_i ($0 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно число — длину самого большого хорошего отрезка. Если таких отрезков нет, выведите 0.

Примеры

стандартный ввод	стандартный вывод
7 20 2 6 4 3 6 8 9	4

Задача Н. Мороженое

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

Вдоль моря узкой полоской тянется пляж. В некоторых точках пляжа расположены ларьки с мороженым. В один прекрасный день не все мороженщики вышли на работу. Распределите мороженщиков по ларькам так, чтобы минимальное расстояние между мороженщиками было как можно больше. Так они будут меньше мешать друг другу.

Формат входных данных

В первой строке вводятся числа N ($2 < N < 10\,001$) — количество ларьков и K ($1 < K < N$) — количество мороженщиков, вышедших на работу. Во второй строке задаются N натуральных чисел в порядке возрастания — координаты ларьков (координаты не превосходят 10^9).

Формат выходных данных

Выведите одно число — минимальное расстояние между соседними ларьками в оптимальной расстановке.

Примеры

<code>ice-cream.in</code>	<code>ice-cream.out</code>
5 3 1 2 3 100 1000	99

Задача I. Вырубка леса

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

Фермер Николай нанял двух лесорубов: Дмитрия и Федора, чтобы вырубить лес, на месте которого должно быть кукурузное поле. В лесу растут X деревьев.

Дмитрий срубает по A деревьев в день, но каждый K -й день он отдыхает и не срубает ни одного дерева. Таким образом, Дмитрий отдыхает в K -й, $2K$ -й, $3K$ -й день, и т.д.

Федор срубает по B деревьев в день, но каждый M -й день он отдыхает и не срубает ни одного дерева. Таким образом, Федор отдыхает в M -й, $2M$ -й, $3M$ -й день, и т.д.

Лесорубы работают параллельно и, таким образом, в дни, когда никто из них не отдыхает, они срубают $A + B$ деревьев, в дни, когда отдыхает только Федор — A деревьев, а в дни, когда отдыхает только Дмитрий — B деревьев. В дни, когда оба лесоруба отдыхают, ни одно дерево не срубается.

Фермер Николай хочет понять, за сколько дней лесорубы срубят все деревья, и он сможет засеять кукурузное поле.

Требуется написать программу, которая по заданным целым числам A , K , B , M и X определяет, за сколько дней все деревья в лесу будут вырублены.

Формат входных данных

Входной файл содержит пять целых чисел, разделенных пробелами: A , K , B , M и X ($1 \leq A, B \leq 10^9$, $2 \leq K, M \leq 10^{18}$, $1 \leq X \leq 10^{18}$)

Формат выходных данных

Выходной файл должен содержать одно целое число — искомое количество дней.

Примеры

стандартный ввод	стандартный вывод
2 4 3 3 25	7

Задача J. Космическое поселение

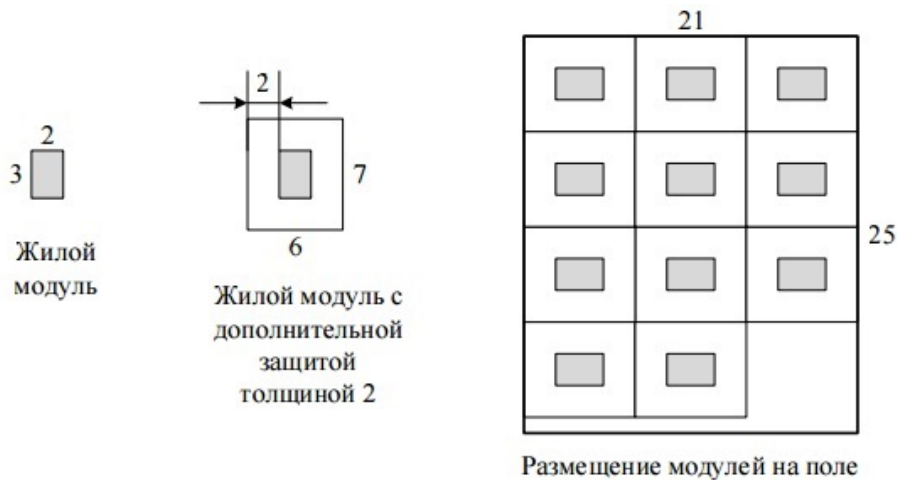
Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Для освоения Марса требуется построить исследовательскую базу. База должна состоять из n одинаковых модулей, каждый из которых представляет собой прямоугольник.

Каждый модуль представляет собой жилой отсек, который имеет форму прямоугольника размером $a \times b$ метров. Для повышения надежности модулей инженеры могут добавить вокруг каждого модуля слой дополнительной защиты. Толщина этого слоя должна составлять целое число метров, и все модули должны иметь одинаковую толщину дополнительной защиты. Модуль с защитой, толщина которой равна d метрам, будет иметь форму прямоугольника размером $(a+2d)(b+2d)$ метров.

Все модули должны быть расположены на заранее подготовленном прямоугольном поле размером $w \times h$ метров. При этом они должны быть организованы в виде регулярной сетки: их стороны должны быть параллельны сторонам поля, и модули должны быть ориентированы одинаково.

Требуется написать программу, которая по заданному количеству и размеру модулей, а также размеру поля для их размещения, определяет максимальную толщину слоя дополнительной защиты, который можно добавить к каждому модулю.



Формат входных данных

Входной файл содержит пять разделенных пробелами целых чисел: n , a , b , w и h ($1 \leq n \leq 10^{18}$). Гарантируется, что без дополнительной защиты все модули можно разместить в поселении описанным образом.

Формат выходных данных

Выходной файл должен содержать одно целое число: максимальную возможную толщину дополнительной защиты. Если дополнительную защиту установить не удастся, требуется вывести число 0.

Примеры

стандартный ввод	стандартный вывод
11 2 3 21 25	2
1 5 5 6 6	0

Задача К. Дремучий лес

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

Чтобы помешать появлению СЭС в лагере, администрация ЛКШ перекопала единственную дорогу, соединяющую «Берендеевы поляны» с Судиславлем, теперь проехать по ней невозможно. Однако, трудности не остановили инспекцию, хотя для СЭС остаётся только одна возможность — дойти до лагеря пешком. Как известно, Судиславль находится в поле, а «Берендеевы поляны» — в лесу.

- Судиславль находится в точке с координатами $(0, 1)$.
- «Берендеевы поляны» находятся в точке с координатами $(1, 0)$.
- Граница между лесом и полем — горизонтальная прямая $y = a$, где a — некоторое число ($0 \leq a \leq 1$).
- Скорость передвижения СЭС по полю составляет V_p , скорость передвижения по лесу — V_f . Вдоль границы можно двигаться как по лесу, так и по полю.

Администрация ЛКШ хочет узнать, сколько времени у неё осталось для подготовки к визиту СЭС. Она попросила вас выяснить, в какой точке инспекция СЭС должна войти в лес, чтобы дойти до «Берендеевых полян» как можно быстрее.

Формат входных данных

В первой строке входного файла содержатся два положительных целых числа — V_p и V_f ($1 \leq V_p, V_f \leq 10^5$). Во второй строке содержится единственное вещественное число — координата по оси Oy границы между лесом и полем a ($0 \leq a \leq 1$)

Формат выходных данных

В единственной строке выходного файла выведите вещественное число с точностью не менее 7 знаков после запятой — координата по оси Ox точки, в которой инспекция СЭС должна войти в лес.

Примеры

<code>stdin</code>	<code>stdout</code>
5 3 0.4	0.783310604
5 5 0.5	0.500000000

Задача L. Эклеры

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

После успешной сдачи всех зачетов Вася купил себе в подарок коробку, содержащую n сладких эклеров. Вася решил каждое утро есть некоторое одинаковое число эклеров, пока они все не закончатся. Однако сосед Васи, Петя, заметил принесенную Васей коробку и тоже решил насладиться вкусом эклеров.

Теперь процесс поедания эклеров выглядит следующим образом: сначала Вася выбирает число k , одинаковое для всех дней. Затем утром он съедает k эклеров из коробки (или доедает все эклеры, если их осталось меньше k), после этого Петя вечером съедает 10% **оставшихся** эклеров. Если эклеры еще не закончились, то на следующий день Вася опять съедает k эклеров, а Петя — 10% от оставшихся и так далее.

Если число эклеров не делится на 10, то Петя округляет «свою» долю в меньшую сторону, например, если в коробке было 97 эклеров, то Петя съест только 9 из них. В частности, если в коробке уже меньше 10 эклеров, то Петя не будет их есть вообще.

Определите, какое наименьшее число k может выбрать Вася такое, что он съест не менее половины всех n эклеров, которые были в коробке изначально. Заметьте, что число k должно быть натуральным.

Формат входных данных

В первой строке содержится натуральное число n ($1 \leq n \leq 10^{18}$) — изначальное количество эклеров.

Формат выходных данных

Вывести единственное число — наименьшее значение k , удовлетворяющее Васю.

Примеры

стандартный ввод	стандартный вывод
68	3
1	1

Задача М. Велогонка

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

Велосипедисты, участвующие в шоссейной гонке, в некоторый момент времени, который называется начальным, оказались в точках, удалённых от места старта на $x_1, x_2, \dots, x_n$ метров (n — общее количество велосипедистов). Каждый велосипедист движется со своей постоянной скоростью $v_1, v_2, \dots, v_n$ метров в секунду. Все велосипедисты движутся в одну и ту же сторону.

Репортёр, освещающий ход соревнований, хочет определить момент времени, в который расстояние между лидирующим в гонке велосипедистом и замыкающим гонку велосипедистом станет минимальным, чтобы с вертолётá сфотографировать сразу всех участников велогонки.

Требуется написать программу, которая по заданному количеству велосипедистов n , заданным начальным положениям велосипедистов $x_1, x_2, \dots, x_n$ и их скоростям $v_1, v_2, \dots, v_n$, вычислит момент времени t , в который расстояние l между лидирующим и замыкающим велосипедистом будет минимальным.

Формат входных данных

Первая строка входного файла содержит целое число n — количество велосипедистов. В последующих n строках указаны по два целых числа: x_i — расстояние от старта до i -го велосипедиста в начальный момент времени ($0 \leq x_i \leq 10^7$) и v_i — его скорость ($0 \leq v_i \leq 10^7$).

Формат выходных данных

В выходной файл необходимо вывести два вещественных числа: t — время в секундах, прошедшее от начального момента времени до момента, когда расстояние в метрах между лидером и замыкающим будет минимальным, l — искомое расстояние.

Числа t и l должны иметь абсолютную или относительную погрешность не более 10^{-6} , что означает следующее. Пусть выведенное число равно x , а в правильном ответе оно равно y . Ответ будет считаться правильным, если значение выражения $|x - y| / \max(1, |y|)$ не превышает 10^{-6} .

Примеры

bicycle.in	bicycle.out
3 0 40 30 10 40 30	1.00000000000000000000 30.00000000000000000000000000
5 90 100 100 70 100 70 110 60 120 35	0.50000000000000000000 5.00000000000000000000000000