

Задача А. Десятичная дробь

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Жюри загадало целые взаимно простые числа a и b ($0 \leq a < b \leq 10^9$). Вам дано число $\frac{a}{b}$ в форме десятичной дроби, округлённой ровно до 18 цифр после десятичной точки. Округление в этой задаче всегда происходит к ближайшему числу, при равенстве к большему. Найдите a и b .

Формат входных данных

В первой строке задано число n — количество тестовых случаев ($1 \leq n \leq 10^4$). Следующие n строк описывают тестовые случаи. Каждая из них содержит десятичную дробь, в которой ровно 18 цифр после десятичной точки. Гарантируется, что каждая дробь получена способом, описанным в условии.

Формат выходных данных

Выведите n строк. Строка с номером i должна содержать числа a и b ($0 \leq a < b \leq 10^9$), которые загадало жюри в i -м тестовом случае.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2	0 1
0.000000000000000000	2 3
0.666666666666666667	

Пояснение к примеру

В первом тестовом случае $\frac{0}{1} = 0$.

Во втором тестовом случае $\frac{2}{3} \approx 0.666\ 666\ 666\ 666\ 666\ 667$.

Задача В. Как уравнивать числа

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Два целых положительных числа, a и b , загаданы и держатся в секрете. Известно лишь, что $1 \leq a, b \leq 10^9$ и $a \neq b$.

С числами можно делать в любом порядке следующие операции:

- выбрать целое положительное число x и умножить число a на x ;
- выбрать целое положительное число x и умножить число b на x .

После каждой операции сообщается результат сравнения a и b . Сделайте числа a и b равными не более чем за 5000 операций.

Протокол взаимодействия

Это интерактивная задача. В каждом тесте начальные значения чисел a и b зафиксированы заранее. Гарантируется, что $1 \leq a, b \leq 10^9$ и $a \neq b$.

Решение выводит операции, каждую в отдельной строке. Операция записывается буквой, за которой следуют пробел и число:

- «a x »: умножить a на x ;
- «b x »: умножить b на x .

Число x должно быть целым, $1 \leq x \leq 2 \cdot 10^{18}$.

В ответ на каждую операцию решение получает строку с одним символом:

- «<», если $a < b$;
- «>», если $a > b$;
- «=», если $a = b$.

В последнем случае решение должно завершить работу. В остальных случаях можно выводить следующую операцию.

Если за 5000 операций не удалось сделать числа равными, решение должно завершить работу — тогда оно получит вердикт «Wrong Answer», а не какой-либо ещё.

Помните, что после вывода каждой операции нужно очищать буфер вывода (например, функцией «`fflush(stdout)`» в языке C, «`cout.flush()`» в C++ или «`sys.stdout.flush()`» в Python). В противном случае выведенная операция может не дойти до проверяющей программы, и тогда решение получит вердикт «Idleness Limit Exceeded».

Пример

Пустые строки добавлены для удобства чтения.

<i>стандартный ввод</i>	<i>стандартный вывод</i>	<i>секретные значения</i>
		$a = 3, \quad b = 5$
>	a 2	$a = 6, \quad b = 5$
	b 3	
<		$a = 6, \quad b = 15$
	a 5	
>		$a = 30, \quad b = 15$
	b 2	
=		$a = 30, \quad b = 30$

Задача С. Аддитивный класс — лайт

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Назовём *аддитивным классом* двух натуральных чисел a и b множество чисел вида $x \cdot a + y \cdot b$, где x и y — неотрицательные целые коэффициенты. Например, аддитивный класс чисел 3 и 5 — это множество $\{0, 3, 5, 6, 8, 9, 10, 11, 12, \dots\}$.

Сколько чисел в аддитивном классе a и b лежат в промежутке от ℓ до h , включительно?

Формат входных данных

В первой строке заданы через пробел два целых числа a и b ($1 \leq a, b \leq 10^7$).

Во второй строке заданы через пробел два целых числа ℓ и h ($0 \leq \ell \leq h \leq 10^{18}$).

Формат выходных данных

В первой строке выведите количество элементов аддитивного класса a и b , лежащих в промежутке от ℓ до h включительно.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
1 2 3 4	2
3 5 0 12	9

Задача D. Аддитивный класс

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

В этой задаче нужно выяснить, сколько чисел из аддитивного класса a и b лежат на отрезке $[l, r]$.

Аддитивным классом двух целых положительных чисел a и b называется множество чисел, представимых в виде суммы нуля или более чисел, каждое из которых равно либо a , либо b . Формально это следующее множество: $\{x \cdot a + y \cdot b \mid x, y \in \mathbb{Z}, x, y \geq 0\}$.

По заданным числам a и b , а также целым числам l и r ($l \leq r$), найдите количество элементов e аддитивного класса a и b , для которых выполнено двойное неравенство $l \leq e \leq r$.

Формат входных данных

Входные данные содержат один или несколько тестовых случаев. В первой строке задано одно число t — количество тестовых случаев ($1 \leq t \leq 10\,000$). Далее заданы сами тестовые случаи.

Описание каждого случая состоит из двух строк. Первая из этих строк содержит два целых положительных числа a и b ($1 \leq a, b \leq 10^9$). Вторая строка содержит два целых числа l и r ($0 \leq l \leq r \leq 10^{18}$).

Формат выходных данных

Выведите ответ на каждый из t заданных тестовых случаев в отдельной строке. Ответом на тестовый случай является одно число — количество элементов аддитивного класса a и b , лежащих на отрезке $[l, r]$.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
2	2
3 5	0
6 8	
6 4	
13 13	

Пояснение к примеру

В примере два тестовых случая.

В первом тестовом случае $a = 3$ и $b = 5$. Аддитивный класс этих чисел содержит числа $0, 3, 5, 6 = 3 + 3, 8 = 3 + 5, 9 = 3 + 3 + 3, \dots$. Два из них принадлежат отрезку $[6, 8]$.

Во втором тестовом случае $a = 6$ и $b = 4$. Очевидно, аддитивный класс этих чисел может содержать только чётные числа, поэтому на отрезке $[13, 13]$ у него нет ни одного элемента.

Задача Е. Целая корова

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Корова находится на бесконечной плоскости в целой точке (x_0, y_0) . Трава растёт в круге с центром в целой точке (x_c, y_c) и целым радиусом r , а также на его границе.

Корова может сколько угодно раз выполнять следующую команду: пройти из текущей целой точки (x_1, y_1) в целую точку (x_2, y_2) , причём тратит на это время, равное Евклидову расстоянию между этими точками. Указанные точки могут совпадать.

Найдите такой набор команд, чтобы корова за минимальное возможное время оказалась в целой точке, где растёт трава.

Формат входных данных

В первой строке задано целое число t — количество тестовых случаев ($1 \leq t \leq 100$). В следующих t строках заданы тестовые случаи, по одному в строке. Каждый тестовый случай задаётся пятью целыми числами x_c, y_c, r, x_0, y_0 — координаты центра травяного круга, его радиус и исходные координаты коровы ($-10^9 \leq x_c, y_c, x_0, y_0 \leq 10^9, 1 \leq r \leq 10^9$).

Формат выходных данных

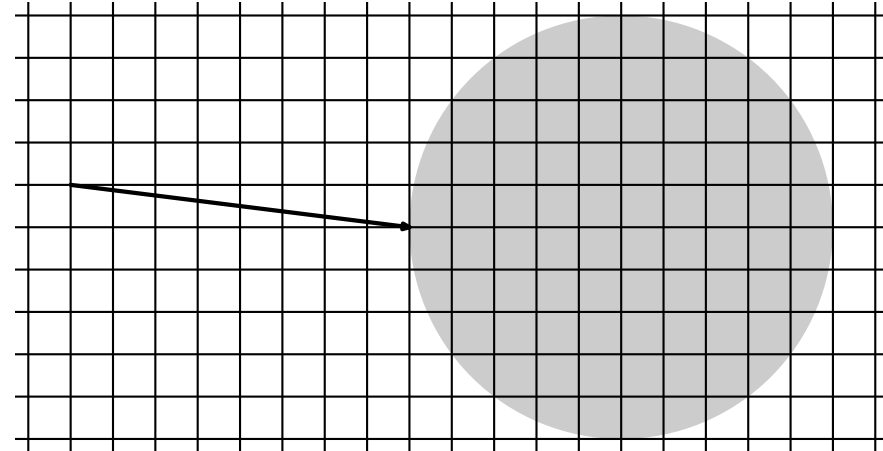
В ответ на каждый тестовый случай выведите две строки. В первой выведите целое число k — количество ходов коровы ($0 \leq k \leq 1\,000\,000$). Во второй выведите $2(k+1)$ чисел — путь коровы: $x_0 y_0 \dots x_k y_k$. Если оптимальных ответов несколько, выведите любой из них.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3	0
1 2 1 1 2	1 2
3 2 5 -10 3	1
0 0 1 10 0	-10 3 -2 2
	3
	10 0 5 0 5 0 1 0

Пояснение к примеру

Картинка соответствует второму тестовому случаю.



Задача F. Целые точки

Имя входного файла: *стандартный ввод*
 Имя выходного файла: *стандартный вывод*
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 512 мегабайт

По трём данным целым числам p , q и n найдите количество точек на координатной плоскости, имеющих целые координаты и удовлетворяющих неравенствам

$$0 \leq x \leq n \text{ и } 0 \leq y \leq x \cdot \sqrt{\frac{p}{q}}$$

Формат входных данных

На первой строке ввода задано целое число t — количество тестов ($1 \leq t \leq 100\,000$). За ней следует t строк, каждая из которых содержит один тест. Каждый тест состоит из трёх целых чисел p , q и n , разделённых одиночными пробелами ($1 \leq p, q \leq 100$; $0 \leq n \leq 10^9$).

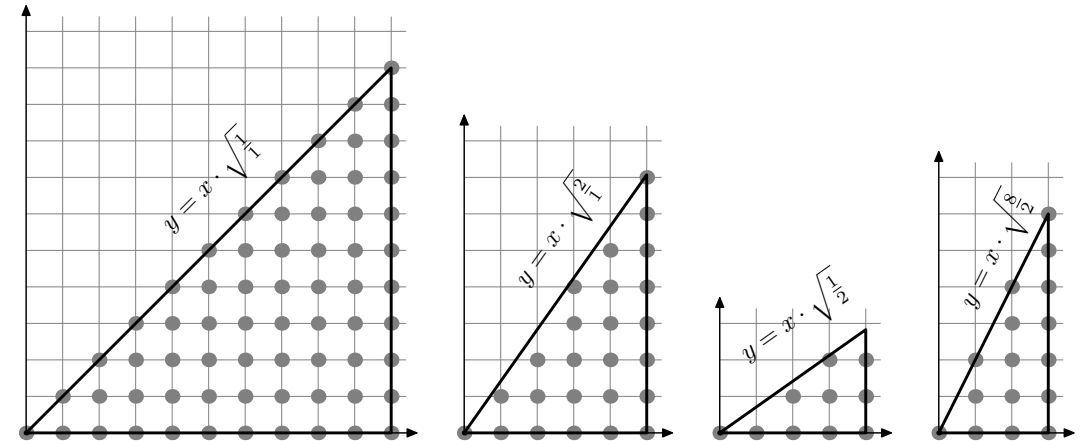
Формат выходных данных

В ответ на каждый тест выведите количество точек с целыми координатами, удовлетворяющих ограничениям, на отдельной строке.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
4	66
1 1 10	25
2 1 5	10
1 2 4	16
8 2 3	

Иллюстрация



Задача G. Игральная кость

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Обычная игральная кость — это кубик, на каждой грани которого написано какое-то число от 1 до 6, причём все эти числа встречаются ровно по одному разу. Кость называется *сбалансированной*, если сумма чисел, написанных на любых двух противоположных гранях, равна 7.

Игральную кость, лежащую на столе, начинают перекачивать через рёбра, и считают очки, выпавшие на ней: после каждого перекачивания к сумме очков добавляется число, написанное на той грани кости, которая оказалась сверху.

Какая минимальная и максимальная сумма очков могла получиться при заданной последовательности перекачиваний, если известно только, что кость сбалансирована, а какие именно числа написаны на каких гранях, неизвестно?

Формат входных данных

В первой строке задана строка, описывающая перекачивания. Она непуста, содержит только символы «D» (перекатить вниз по столу), «L» (перекатить влево по столу), «R» (перекатить вправо по столу) и «U» (перекатить вверх по столу) и имеет длину не более 10 000 символов.

Формат выходных данных

В первой строке выведите два числа через пробел — минимальную и максимальную сумму выпавших на кубике чисел.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
RUL	6 15
UD	3 11

Задача Н. Катание кубика — лайт

Имя входного файла: *стандартный ввод*
 Имя выходного файла: *стандартный вывод*
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 512 мегабайт

В этой задаче требуется найти сумму чисел, отпечатавшихся на плоскости при катании кубика вдоль отрезка.

Рассмотрим плоскость, на которой введена декартова система координат. Плоскость разбита на клетки прямыми вида $x = c$ и $y = c$ для целых чисел c .

Зафиксируем два взаимно простых положительных целых числа a и b . Рассмотрим отрезок прямой между точками $(0, 0)$ и (a, b) на нашей плоскости. Выделим клетки, с которыми отрезок имеет больше одной общей точки. В силу взаимной простоты a и b эти клетки образуют путь, связный по стороне, из клетки с левым нижним углом $(0, 0)$ в клетку с правым верхним углом (a, b) .

Возьмём игральную кость — кубик со стороной 1, различным граням которого соответствуют различные целые числа от 1 до 6. Поставим кубик на первую клетку пути и будем катить его по пути до последней клетки, перекачивая через стороны. Каждый раз, когда кубик оказывается на клетке, на ней отпечатывается число с его нижней грани.

По заданным a и b , а также описанию кубика в начальном положении, найдите сумму чисел, отпечатавшихся на плоскости.

Формат входных данных

В первой строке ввода задано два целых числа a и b ($1 \leq a, b \leq 10\,000$). Гарантируется, что числа a и b взаимно просты.

Во второй строке записано шесть целых чисел — числа на гранях кубика. Гарантируется, что каждое число от 1 до 6 оказалось ровно на одной грани кубика. Никакие другие свойства расположения чисел на гранях не гарантируются. Грани перечислены в следующем порядке: передняя, верхняя, правая, левая, нижняя, задняя.

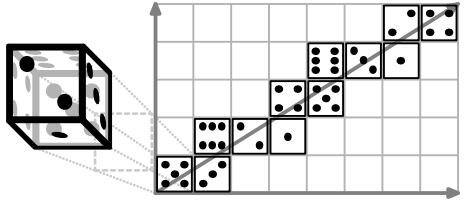
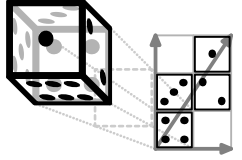
В начальном положении на плоскости кубик стоит на своей нижней грани. При движении из начального положения в соседнюю клетку в положительном направлении вдоль оси Ox кубик перекачивается на правую грань. При движении из начального положения в соседнюю клетку в положительном направлении вдоль оси Oy кубик перекачивается на заднюю грань.

Формат выходных данных

Выведите одно целое число — сумму чисел, отпечатавшихся на плоскости

при движении кубика вдоль заданного отрезка.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>	<i>пояснение</i>
8 5 1 2 3 4 5 6	42	
2 3 6 1 2 5 4 3	10	

Пояснения к примерам

К примерам прилагаются картинки. Слева на картинке — кубик в его начальном положении. Справа — след кубика при его движении вдоль заданного отрезка.

Задача I. Катание кубика

Имя входного файла: *стандартный ввод*
 Имя выходного файла: *стандартный вывод*
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 512 мегабайт

В этой задаче требуется найти сумму чисел, отпечатавшихся на плоскости при катании кубика вдоль отрезка.

Рассмотрим плоскость, на которой введена декартова система координат. Плоскость разбита на клетки прямыми вида $x = c$ и $y = c$ для целых чисел c .

Зафиксируем два взаимно простых положительных целых числа a и b . Рассмотрим отрезок прямой между точками $(0, 0)$ и (a, b) на нашей плоскости. Выделим клетки, с которыми отрезок имеет больше одной общей точки. В силу взаимной простоты a и b эти клетки образуют путь, связный по стороне, из клетки с левым нижним углом $(0, 0)$ в клетку с правым верхним углом (a, b) .

Возьмём игральную кость — кубик со стороной 1, различным граням которого соответствуют различные целые числа от 1 до 6. Поставим кубик на первую клетку пути и будем катить его по пути до последней клетки, перекачивая через стороны. Каждый раз, когда кубик оказывается на клетке, на ней отпечатывается число с его нижней грани.

По заданным a и b , а также описанию кубика в начальном положении, найдите сумму чисел, отпечатавшихся на плоскости.

Формат входных данных

В первой строке ввода задано два целых числа a и b ($1 \leq a, b \leq 10^{18}$). Гарантируется, что числа a и b взаимно просты.

Во второй строке записано шесть целых чисел — числа на гранях кубика. Гарантируется, что каждое число от 1 до 6 оказалось ровно на одной грани кубика. Никакие другие свойства расположения чисел на гранях не гарантируются. Грани перечислены в следующем порядке: передняя, верхняя, правая, левая, нижняя, задняя.

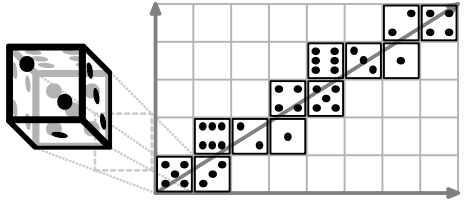
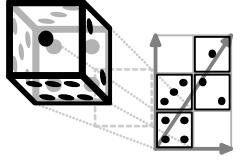
В начальном положении на плоскости кубик стоит на своей нижней грани. При движении из начального положения в соседнюю клетку в положительном направлении вдоль оси Ox кубик перекачивается на правую грань. При движении из начального положения в соседнюю клетку в положительном направлении вдоль оси Oy кубик перекачивается на заднюю грань.

Формат выходных данных

Выведите одно целое число — сумму чисел, отпечатавшихся на плоскости

при движении кубика вдоль заданного отрезка.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>	<i>пояснение</i>
8 5 1 2 3 4 5 6	42	
2 3 6 1 2 5 4 3	10	

Пояснения к примерам

К примерам прилагаются картинки. Слева на картинке — кубик в его начальном положении. Справа — след кубика при его движении вдоль заданного отрезка.

Задача J. Небольшой многоугольник

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

На плоскости с целочисленной решёткой нарисован многоугольник с площадью не более 1000 (не обязательно выпуклый). Необходимо выдать список всех точек с целыми координатами, покрытых этим многоугольником (если точка лежит на границе, то она считается покрытой).

Формат входных данных

В первой строке записано число вершин многоугольника n ($1 \leq n \leq 256$). В последующих n строках содержатся целые координаты вершин x_i и y_i , не превосходящие по модулю 10^9 . Координаты задаются в порядке обхода. Гарантируется, что входные данные корректны и задают многоугольник.

В каждом тесте может быть несколько (не более 10) наборов входных данных. Наборы следуют один за другим и отделяются переводом строки. Ввод завершается числом 0.

Формат выходных данных

Для каждого набора в первой строке выведите число целых точек, покрытых многоугольником, — P . Далее в P строках выведите x - и y -координаты покрытых точек, разделяя их одним пробелом. Точки можно выводить в произвольном порядке.

Вывод для различных наборов входных данных должен быть разделён одной пустой строкой.

Пример

<i>стандартный ввод</i>	<i>стандартный вывод</i>
3	3
0 1	0 1
0 0	0 0
1 0	1 0
4	9
0 0	0 0
0 2	0 1
2 2	2 2
2 0	0 2
	1 0
0	1 2
	2 0
	1 1
	2 1