

## Задача 1. Робот – мойщик окон

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Жена Петра очень уставала от мытья окон в квартире, и Петр купил ей робота – мойщика окон. Рабочая поверхность этого робота представляет собой квадрат со стороной  $A$ , соосно с центром которого вращается круглая щетка радиусом  $R$  ( $R \leq A/2$ ), которая, собственно, и моет окно.

Когда Петр принес покупку домой и разобрался с конструкцией, он сообразил, что полностью помыть окно робот не сможет, и часть окна жене все же придется мыть самой. Чтобы оправдать свою покупку перед женой, он решил написать программу, которая продемонстрирует ей, как мало работы теперь нужно делать. Однако с программой у него не заладилось, помогите ему, пожалуйста.

Стекло окна – прямоугольник  $C \times D$ . Стекло находится в раме, которая выступает над поверхностью окна и не дает роботу выехать за край (квадрат робота может коснуться рамы). Необходимо вычислить, какой процент составляет площадь, которую робот не сможет помыть (и придется мыть жене), от общей площади стекла.

### Формат входных данных

В единственной строке четыре отделенных друг от друга пробелами вещественных числа  $C$ ,  $D$ ,  $A$ ,  $R$  – размеры стекла, сторона квадрата рабочей поверхности робота, радиус щетки. Все эти числа положительные, лежат в диапазоне от 0.1 до 1 000 и приведены с точностью 1 знак после десятичной точки,  $R \leq A/2$ .

### Формат выходных данных

В единственной строке одно вещественное число – процент площади стекла, которую робот не сможет помыть, с точностью 3 знака после десятичной точки.

### Примеры

<i>standard input</i>	<i>standard output</i>
1000.0 100.0 10.0 3.0	4.392

## Задача 2. Раскладка товара

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 256 MiB

Директор супермаркета обнаружил, что товары на полках лежат хаотично, их плохо видно, полки смотрятся некрасиво. Он обратился к маркетологу за помощью. Маркетолог объяснил, что при поступлении нового товара сотрудники просто кладут его на полку в любое свободное место из-за спешки, а создание красивой раскладки требует много времени. Тогда директор обратился к программисту с просьбой помочь решить эту проблему. Программист сказал, что оценка красоты не является его специализацией и попросил маркетолога предоставить хоть какие-то критерии для определения красивой раскладки, например, зависимость красоты от положения и высоты товара. Маркетолог объяснил, что простая сортировка товаров по высоте не будет выглядеть красиво, и обсуждение остановилось. Тогда программист решил обратиться за помощью к математику, чтобы формализовать идеи маркетолога. После долгих обсуждений математик предложил следующий критерий для выбора наилучшего места для нового товара.

Допустим, на полке лежит  $N$  товаров, и каждый из них имеет свою высоту (обозначим их как  $A_1, A_2, \dots, A_N$ ). Красота  $i$ -го товара определяется по формуле  $A_i K_i / S$ , где  $K_i$  – количество последовательностей товаров на полке (без пропусков), в которых  $i$ -й товар является самым низким, а  $S = \sum_{m=1}^N A_m K_m$ .

Переставлять уже имеющиеся товары на полке – долгая задача, поэтому при добавлении нового товара с высотой  $X$  его можно поместить перед первым товаром, после последнего или между любыми двумя товарами, раздвинув их. Программист должен определить наилучшее место для нового товара по приведенному критерию (при этом то, что красота уже лежащих товаров может ухудшиться, не учитывается). И этот программист – Вы.

### Формат входных данных

В первой строке два разделенных пробелом целых положительных числа  $M$  и  $X$ , где  $M$  – число уже лежащих на полке товаров,  $1 \leq M \leq 1\,000$ ,  $X$  – высота добавляемого товара,  $1 \leq X \leq 10^9$ . Во второй строке последовательность  $M$  разделенных пробелами целых положительных высот лежащих на полке товаров, все высоты различны, не превышают  $10^9$  и не совпадают с  $X$ .

### Формат выходных данных

Два разделенных пробелом целых положительных числа – числитель и знаменатель несократимой дроби – наибольшая возможная красота добавляемого товара по приведенному условию критерию.

### Примеры

<i>standard input</i>	<i>standard output</i>
2 2	1 2
3 5	

## Задача 3. Картография

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 MiB

Сегодня электромобили становятся все более популярными, ведь они считаются самым экологичным видом автотранспорта. Однако у электромобилей есть своя особенность – ограниченный запас хода, и найти место для их зарядки не так просто. Вы – сотрудник картографического сервиса, и вам предстоит разработать программу, которая поможет водителям электромобилей определить, смогут ли они доехать из пункта  $A$  до пункта  $B$ , имея запас хода не более  $W$  километров. Для этого у Вас есть карта, на которой указаны все точки зарядки (в них можно подзарядиться в пути), а также дороги, соединяющие эти точки и их длины.

### Формат входных данных

В первой строке три целых числа  $N$ ,  $2 \leq N \leq 10^5$ ,  $M$ ,  $1 \leq M \leq 10^5$ ,  $K$ ,  $1 \leq K \leq 10^5$ , – число точек зарядки, число дорог, число запросов от водителей, соответственно.

Далее  $M$  строк по три разделенных пробелами целых числа  $u$ ,  $v$ ,  $w$ , которые описывают дороги,  $u$  и  $v$  – порядковые номера точек зарядки (несовпадающие,  $1 \leq u, v \leq N$ ),  $w$  – длина дороги (положительная, не превышает  $10^9$ ). Все дороги с двусторонним движением.

Далее  $K$  строк вида  $A$ ,  $B$ ,  $W$ , которые описывают запросы водителей,  $A$  и  $B$  – порядковые номера точек зарядки, откуда и куда хочет доехать водитель (несовпадающие,  $1 \leq A, B \leq N$ ),  $W$  – запас хода его электромобиля (положительный, не превышает  $10^9$ ).

### Формат выходных данных

В первой строке одно целое число – количество запросов, для которых существует путь между запрошенными точками зарядки, не содержащий дорог, длина которых превышает запас хода электромобиля. Если это число положительное, то во второй строке отсортированные по возрастанию разделенные пробелами порядковые номера таких запросов, для которых существует путь.

### Примеры

<i>standard input</i>	<i>standard output</i>
4 3 3	2
1 2 1	1 3
2 3 5	
4 3 10	
1 3 5	
4 3 5	
4 3 10	

## Задача 4. Подарки

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Мартышка, Слононок, Удав и Попугай готовят подарки. Они собрали много разных фруктов и хотят разложить их в наборы по  $M$  штук таким образом, чтобы все фрукты в каждом наборе были различны. Зная, сколько у них есть фруктов каждого вида, определите, сколько подарочных наборов они смогут собрать.

### Формат входных данных

В первой строке два целых числа через пробел:  $K$ ,  $1 \leq K \leq 10^5$ , – количество разных видов фруктов, и  $M$ ,  $1 \leq M \leq 10^5$ , – количество разных фруктов, которое должно быть в каждом подарочном наборе.

Во второй строке  $K$  целых положительных чисел через пробел, обозначающих количество имеющихся фруктов каждого вида. Эти числа не превышают  $10^6$ .

### Формат выходных данных

В первой и единственной строке одно целое неотрицательное число – максимальное количество фруктовых подарочных наборов, которое можно составить.

### Примеры

<i>standard input</i>	<i>standard output</i>
3 3 2 3 3	2
5 4 2 2 3 5 6	3

## Задача 5. Шифрование Попугая-Удава

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Попугай придумал новый способ шифрования строки, который заключается в следующем. Чтобы зашифровать произвольную строку  $S$  берется целое положительное число  $X$  – параметр шифрования. Далее символ с индексом  $i$  строки  $S$  ставится в позицию  $i \cdot X$  новой строки  $S'$ , а остальные символы строки  $S'$  заполняются произвольными латинскими буквами.

Когда Попугай рассказал об этом способе Мартышке и Слононку, они тут же захотели вместе что-то зашифровать, но Мартышка захотела вместо  $X$  взять число  $P$ , а Слононок –  $Q$ , что привело к проблемам. Если символы строки  $S'$ , которые выбираются произвольным образом, они смогли договориться делать одинаковыми, то, когда на одну и ту же позицию строки  $S'$  Мартышка и Слононок должны были поставить разные символы из строки  $S$ , было совершенно неясно, что делать.

Удав предложил компромиссное решение – на место несовпадающих символов ставить прочерк ('-'), а те символы, которые не совпали, ставить на следующую расчетную позицию: Мартышке – через  $P$  символов, а Слононку – через  $Q$  символов. В результате, позиции всех далее идущих символов также сдвигались на  $P$  и  $Q$ , соответственно.

Зная строку, которую Мартышка и Слононок хотят зашифровать, а также выбранные ими числа  $P$  и  $Q$ , определите, сколько прочерков будет в строке, полученной методом шифрования Попугая с модификацией Удава.

### Формат входных данных

В первой строке два целых положительных числа через пробел:  $P$  и  $Q$ ,  $1 \leq P, Q \leq 10^4$ , – значения параметра шифрования  $X$ , которые выбрали Мартышка и Слононок.

Вторая строка состоит из строчных латинских букв 'a' – 'z' и представляет собой строку, которую они хотят зашифровать. Длина этой строки находится в диапазоне от 1 до  $10^5$  символов.

### Формат выходных данных

В первой и единственной строке одно целое неотрицательное число – количество прочерков, которое получится при шифровании заданной строки методом Попугая с модификацией Удава, если использовать в качестве параметров шифрования числа  $P$  и  $Q$ .

### Примеры

<i>standard input</i>	<i>standard output</i>
1 2 abcabcabc	4

## Задача 6. Вдоль Удава

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Мартышка и Слоненок решил поиграть. Они развернули Удава, разбили его длину на  $N$  отрезков так, что 1-ый отрезок – это голова Удава, а  $N$ -ый – кончик его хвоста, поставили Попугая на голову Удава и стали по очереди двигать его к хвосту. Победителем считался тот, кто по окончании своего хода поставит Попугая на кончик хвоста Удава. При этом за один ход Попугая можно было двигать только на определенное количество отрезков. Мартышка настояла на том, что именно она в начале игры будет определять все возможные варианты ходов, согласившись при этом, что среди предложенных вариантов обязательно будет перемещение на соседний отрезок. Слоненку же Мартышка разрешила выбрать, кто будет ходить первым.

Зная количество отрезков, помогите Слоненку определить, первым или вторым ему надо ходить, чтобы выиграть.

### Формат входных данных

В первой строке два целых числа через пробел:  $N$ ,  $2 \leq N \leq 10^4$ , – количество отрезков, на которые разбили Удава, и  $K$ ,  $1 \leq K \leq 100$ , – количество вариантов ходов, предложенных Мартышкой.

Во второй строке через пробел  $K$  целых чисел от 1 до  $N - 1$ , определяющие, насколько отрезков можно сдвинуть Попугая от головы Удава к кончику хвоста за один ход. Гарантируется, что все эти числа различны, а первое из них – единица.

### Формат выходных данных

В первой и единственной строке ‘1’, если Слоненок должен ходить первым, чтобы гарантированно победить, и ‘2’, если для гарантированной победы он должен ходить вторым.

### Примеры

<i>standard input</i>	<i>standard output</i>
4 2 1 2	2
5 2 1 2	1

## Задача 7. Игрушка

Имя входного файла:	<i>standard input</i>
Имя выходного файла:	<i>standard output</i>
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 MiB

Для обучения детей работе с джойстиком Вам предложили написать игру. На экране прямоугольный параллелепипед, разделенный по вертикали на слои горизонтальными пластинами с нарисованными клетками (размером  $N_x \times N_y$  клеток), всего слоев  $N_z$ . В некоторых пластинах вырезаны отверстия, для каждого отверстия указан номер слоя и номер клетки по каждой координате. Некоторые отверстия могут быть в соседних клетках.

Ребенок с помощью джойстика управляет роботом, начальное положение которого в верхнем слое с координатами  $(1, 1, N_z)$ . У джойстика 4 кнопки, на которых написано 'L', 'R', '+', '-'. Команда 'L' – повернуть на месте против часовой стрелки на 90 градусов, 'R' – повернуть на месте по часовой стрелке на 90 градусов, '+' – сдвинуться на клетку вперед, '-' – на клетку назад. Робот на экране перемещается по конструкции, и, если попадет на клетку с отверстием, то перемещается на один слой ниже. По бокам конструкции стенок нет, и при перемещении за размеры конструкции робот выпадет. Если робот в нижнем слое попадет в отверстие, он тоже выпадет из конструкции.

Ваша задача вычислить, где окажется робот после выполнения серии команд.

### Формат входных данных

В первой строке 5 разделенных пробелами целых чисел,  $N_x$ ,  $1 \leq N_x \leq 1000$ ,  $N_y$ ,  $1 \leq N_y \leq 1000$ ,  $N_z$ ,  $1 \leq N_z \leq 1000$ ,  $M$ ,  $0 \leq M \leq 10\,000$ ,  $K$ ,  $1 \leq K \leq 10\,000$ , – где  $N_x$ ,  $N_y$ ,  $N_z$  – размеры конструкции,  $M$  – число отверстий,  $K$  – число команд.

Далее  $M$  строк с тремя разделенными пробелами координатами отверстий (все координаты целые положительные, не превышают размер по соответствующей координате). Гарантируется, что отверстия не повторяются.

Далее  $K$  команд вида «символ число». Символы и смысл команд описаны в условии, число целое положительное, отделено от символа пробелом, не превышает 99, и означает количество последовательных повторов команды.

Начальное положение робота  $(1, 1, N_z)$ , он направлен в положительном направлении по оси  $X$ . Ось  $Y$  направлена относительно оси  $X$  против часовой стрелки.

### Формат выходных данных

Если в процессе движения робот выпал из конструкции, вывести  $-1$ . Иначе вывести три координаты конечного положения через пробелы.

## Примеры

<i>standard input</i>	<i>standard output</i>
3 3 3 2 8 2 2 3 2 3 2 L 1 + 2 R 1 + 1 L 1 - 1 + 1 - 2	2 1 1
3 3 3 2 2 2 2 3 2 3 2 R 1 + 2	-1

## Задача 8. Произведение цифр

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Определите минимальное  $K$ -значное число, произведение цифр которого равно  $N$ .

### Формат входных данных

В первой и единственной строке два целых числа через пробел:  $N$ ,  $1 \leq N \leq 10^9$ , и  $K$ ,  $1 \leq K \leq 10^3$ .

### Формат выходных данных

В первой и единственной строке вывести минимальное  $K$ -значное число, произведение цифр которого равно  $N$ . Если такое число не существует, вывести  $-1$ .

### Примеры

<i>standard input</i>	<i>standard output</i>
225 4	1559
225 2	-1

## Задача 9. Ламповый куб

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

В каждой единичной ячейке куба со стороной  $N$  расположена лампочка. Включать лампочки в кубе можно только с помощью команд одного из трех типов: « $x=A, y=B$ », « $y=A, z=B$ » и « $z=A, x=B$ », где  $A$  и  $B$  – произвольные целые числа от 1 до  $N$ . Каждая такая команда включает лампочки в соответствующем ряду куба при нумерации рядов от 1 до  $N$  вдоль каждой из трех его осей, которые обозначены как  $x$ ,  $y$  и  $z$ .

Если лампочку включают повторно, ее состояние не меняется и она продолжает светить.

Зная последовательность команд, определите, сколько лампочек будет включено после ее выполнения.

### Формат входных данных

В первой строке два целых положительных числа через пробел:  $N$  – размер куба и  $K$  – количество выполняемых команд,  $1 \leq N \leq 10^8$ ,  $0 \leq K \leq 10^4$ .

Далее  $K$  строк, каждая из которых является командой одного из трех типов: « $x=A, y=B$ », « $y=A, z=B$ » и « $z=A, x=B$ », где  $A$  и  $B$  – целые числа,  $1 \leq A, B \leq N$ .

Гарантируется, что все команды различны.

### Формат выходных данных

В первой и единственной строке одно целое неотрицательное число – количество лампочек, включенных в результате выполнения заданной последовательности команд.

### Примеры

<i>standard input</i>	<i>standard output</i>
10 3 x=1,y=1 x=2,y=5 y=5,z=10	29

## Задача 10. 38 попугаев

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Да, в попугаях удав был гораздо длиннее. Измерив еще пару раз, сколько кто в длину, ширину и даже в высоту в попугаях, Мартышка, Слононок, Удав и Попугай устали и заскучали. Измерять больше было некого и нечего. Но тут умный Слононок предложил: “А давайте не просто измерять, а складывать и составлять уравнения!”. “Это как?” – спросила любопытная Мартышка. “Ну... Вот я, например, в длину  $A$  мартышек плюс  $B$  попугаев. А какова минимально возможная длина Попугая в желудях?” – ответил Слононок. “А почему я в желудях?” – обиделся Попугай. “Ну... потому, что это самое маленькое, чем мы сегодня что-то измеряли”, – объяснил Слононок. “Давайте всех измерять в желудях!” – предложила Мартышка. “Так я это и хотел!” – обрадовался понятливости Мартышки Слононок. “Только для решения твоей задачи чего-то не хватает”, – добавил Удав. “Конечно, я забыл сказать, что моя длина в желудях  $C$ ” – ответил Слононок, – “этого уже достаточно.”. “Да,” – сказал Удав, – “если, конечно, все мы в желудях имеем целые положительные длины”. После того, как Мартышке объяснили, как решается задача, Попугай предложил ее усложнить, используя не только сложение, но и вычитание. Слононок согласился, добавив, что для упрощения записи задачи надо разрешить в записи “ $A$  мартышек +  $B$  попугаев =  $C$  желудей” отрицательные значения  $A$  или  $B$ . Помогите Мартышке научиться решать такую задачу!

### Формат входных данных

В единственной строке три разделенных пробелами ненулевых целых числа,  $A$ ,  $B$  и  $C$ , из условия задачи. Все числа по модулю не превышают 100 000,  $A$  и  $B$  не могут быть одновременно отрицательными,  $C$  всегда положительное.

### Формат выходных данных

В единственной строке одно целое положительное число – минимально возможная длина Попугая в желудях, или  $-1$ , если задача не имеет решения.

### Примеры

<i>standard input</i>	<i>standard output</i>
38 5 43	1
38 5 40	-1

## Задача 11. Прямоугольники

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

На координатной плоскости расположены два прямоугольника, вершины которых имеют целые координаты, причем эти прямоугольники, возможно, накладываются друг на друга. Зная размеры этих прямоугольников по каждой из осей и периметр фигуры, полученной их объединением, определите максимально возможную площадь их пересечения.

Напомним, что периметр – это общая длина границы фигуры. В случае, если прямоугольники не накладываются, будем считать периметром объединения сумму их периметров.

### Формат входных данных

В первой строке и единственной строке 5 целых положительных чисел через пробел –  $W_1, H_1, W_2, H_2, P$ , где  $W_1$  – размер первого прямоугольника по оси  $X$ ,  $H_1$  – размер первого прямоугольника по оси  $Y$ ,  $W_2$  – размер второго прямоугольника по оси  $X$ ,  $H_2$  – размер второго прямоугольника по оси  $Y$ ,  $P$  – периметр фигуры, являющейся объединением двух заданных прямоугольников. При этом  $1 \leq W_1, W_2, H_1, H_2 \leq 10^8$ ,  $4 \leq P \leq 6 \cdot 10^8$ .

### Формат выходных данных

В первой и единственной строке одно целое неотрицательное число – максимально возможная площадь пересечения заданных прямоугольников.

### Примеры

<i>standard input</i>	<i>standard output</i>
13 5 8 9 56	12

## Задача 12. Один разрез

Имя входного файла: *standard input*  
Имя выходного файла: *standard output*  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 MiB

Нитку сложили пополам, потом еще раз пополам и еще раз пополам, после чего получившийся шнурок разрезали один раз в произвольном месте. После измерения кусков нитки, образовавшихся после разрезания, оказалось, что все они имеют целочисленную длину. Часть замеров потерялась, и Вам необходимо по длине двух кусков определить, какой длины могла быть нитка изначально.

### Формат входных данных

В первой строке и единственной строке два целых числа  $L_1$  и  $L_2$  через пробел – длины двух кусков нитки, полученных в результате разреза,  $1 \leq L_1, L_2 \leq 100$ , причем  $L_1$  может быть равно  $L_2$ .

### Формат выходных данных

В первой и единственной строке через пробел целые числа в порядке возрастания – возможные длины исходной нитки. Если таких длин бесконечно много, то вывести минимальную и через пробел слово «INFINITY».

### Примеры

<i>standard input</i>	<i>standard output</i>
3 1	20 28
1 1	12 INFINITY