1. **Программа лояльности**

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Одна из форм реализации программы лояльности – предоставление клиентам накопительной дисконтной карты. В некоторой компании на каждый товар определено начисление бонусных баллов как доли от стоимости приобретённого товара. То есть на виртуальном счету, «привязанном» к дисконтной карте, накапливаются бонусные баллы при каждой покупке пользователем товара и предъявления этой карты.

Компанией при расчёте бонусов по каждому товару округление стоимости производится до сотых денежной единицы в пользу клиента.

Входные данные

На первой строке содержится одно единственное положительное число N (0 < N < 104) – количество записей в файле. Далее идут N строк, на каждой строке расположена одна запись.

Каждая запись содержит следующие данные: номер предъявленной карты – последовательность из 16 десятичных цифр, артикул товара (уникальный идентификатор товара) – последовательность из 9 десятичных цифр, название товара заключённого в двойные кавычки – может содержать только следующие символы: буквы латинского алфавита, пробелы и десятичные цифры (не более 50 символов), цена товара P (0.00 < P ≤ 10000.00) – всегда дано с точностью до 2‑х знаков после запятой, количество приобретённого товара Q (0 < Q < 1000) – количество товара может быть указано с десятичной составляющей до 3‑х знаков после запятой, процент начисления бонусов – целое число pr (0 ≤ pr ≤ 100).

Все данные в записях отделены друг от друга единственным пробелом. Гарантируется, что во всех записях один и тот же товар имеет одно и то же наименование.

Выходные данные

R1 строк (где R1 – количество уникальных карт во входных данных), в каждой из которых уникальный номер карты из входных данных и через пробел величина бонусных баллов на счету этой бонусной карты. Порядок следования карт может быть любым.

Далее R2 строк (где R2 – количество уникальных товаров во входных данных), в каждой из которых артикул, название товара соответствующее артикулу и, какое количество этого товара было продано. Название должно быть в двойных кавычках. Артикул и название должны быть отделены пробелом.

Примеры

|  |  |
| --- | --- |
| Ввод | Вывод |
| 5  1234567890123456 123456789 "abc" 1.01 10.9 2  0000000000000000 123456789 "abc" 10.9 1.01 3  0000000000000000 000000000 " e " 0.01 99.99 5  3141592653589793 123454789 "abc" 10000 1 6  3141592653589793 123654789 "AbC" 1 999.999 6 | 0000000000000000 0.39  1234567890123456 0.23  3141592653589793 660.00  000000000 " e " 99.990  123454789 "abc" 1  123456789 "abc" 11.91  123654789 "AbC" 999.999 |

1. **Непростая последовательность**

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Введём несколько определений для описания задачи: под последовательностью понимается упорядоченное множество всех натуральных чисел из диапазона [L, R] , где R и L также натуральные числа, при этом R ≥ L; непростая последовательность – это та, которая не содержит ни одного простого числа; простое число K – это натуральное число, которое имеет в точности два различных делителя из ряда натуральных чисел (1 и K).

Требуется по заданной через L и R последовательности определить, является ли она непростой.

Входные данные

Два целых положительных числа (1 ≤ L ≤ R ≤ 1014) записанные через пробел.

Выходные данные

«Yes» без кавычек, если последовательность непростая или любое простое число P (L ≤ P ≤ R) в качестве доказательства, что она такой не является.

Примеры

|  |  |
| --- | --- |
| Ввод | Вывод |
| 1 3 | 2 |
| 1 1 | Yes |
| 15 17 | 17 |
| 24 28 | Yes |
| 29 30 | 29 |

1. **Проверка бит**

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Вам дан ряд байт (целые числа от 0 до 255), и вам нужно определить, сколько бит находится в состоянии «1» в данных байтах с номерами от A до B (включительно).

Входные данные

В первой строке дано целое число N (1 ≤ N ≤ 104) – количество байт, в следующей строке через один пробел дано N целых чисел di (0 ≤ di ≤ 255) – значения байт. На следующей строке дано целое число M (1 ≤ M ≤ 105) – количество запросов. Далее на M строках даны пары целых числе Aj, Bj, разделённые одним пробелом (1 ≤ Aj, Bj ≤ N) – границы интервалов, на которых нужно подсчитать количество бит.

Выходные данные

Для каждого интервала на отдельной строке выведите количество бит в состоянии «1».

Пример

| Ввод | Вывод |
| --- | --- |
| 5  12 130 250 11 56  3  1 1  3 5  2 4 | 2  12  11 |

1. **Папа у Васи**

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Папа у Васи силён в …, нет, не в математике, а в гирях. Имеются четыре цельнолитые спортивные гири следующих номиналов масс: 8, 16, 24 и 32 кг, а также разновес из набора гирь, который шёл в комплекте к весам – 1, 2 и 4 кг (по одной гире каждой массы). Васе потребовалось на одну из чаш рычажных весов положить гири общей массой N кг.

Определите, можно ли получить массу в N кг имеющимися гирями и как её получить, если это возможно!

Входные данные

Одно целое положительное число N (1 ≤ N ≤ 100).

Выходные данные

«Impossible» без кавычек, если массу в N кг получить не удастся, в противном случае – перечислите веса всех гирь, одновременное выставление которых обеспечит необходимый вес.

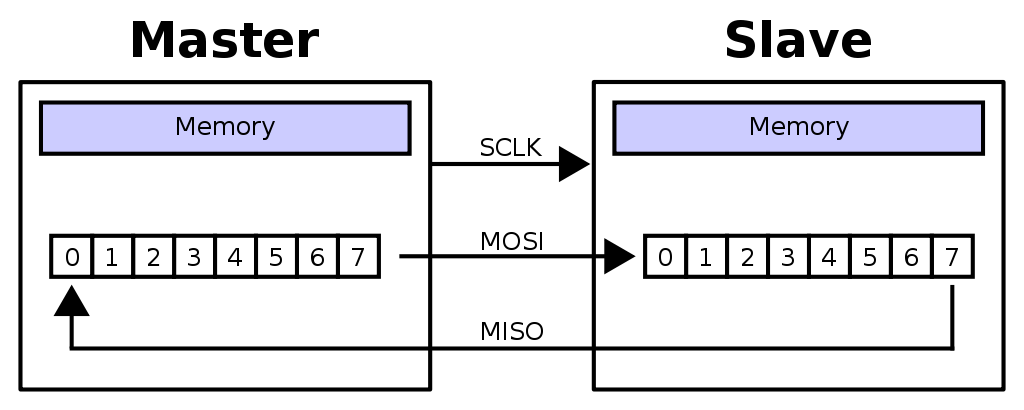
Пример

|  |  |
| --- | --- |
| Ввод | Вывод |
| 25 | 24 1 |
| 33 | 8 24 1 |
| 100 | Impossible |

1. **Шина SPI**

Ограничение по времени: **4000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

SPI (англ. Serial Peripheral Interface, SPI bus – последовательный периферийный интерфейс, шина SPI) – последовательный синхронный стандарт передачи данных в режиме полного дуплекса, предназначенный для обеспечения простого и недорогого высокоскоростного сопряжения микроконтроллеров и периферии.

Подлежащие передаче данные ведущее и ведомое устройства помещают в сдвиговые регистры. После этого ведущее устройство начинает генерировать импульсы синхронизации на линии SCLK, что приводит к взаимному обмену данными. Передача данных осуществляется бит за битом от ведущего по линии MOSI и от ведомого по линии MISO. Передача каждого байта осуществляется от старшего бита к младшим. После окончания передачи очередного байта каждым из устройств в регистры сдвига загружаются следующие байты, и процедура передачи повторяется.

Определите, что находится в регистрах сдвига каждого из устройств после N импульсов синхронизации на линии SCLK.

Входные данные

Первая строка содержит число M (1 ≤ M ≤ 104) – количество байт, предназначенных для передачи в каждом направлении.

Во второй строке через пробел перечислено M чисел, каждое из которых определяет значение очередного байта, предназначенного для передачи от Master к Slave.

В третьей строке через пробел перечислено M чисел, каждое из которых определяет значение очередного байта, предназначенного для передачи от Slave к Master.

В четвёртой строке указано одно число K (1 ≤ K ≤ 104) – количество запросов, на которые надо дать ответы.

На следующей строке идёт K чисел через пробел – запросы. Каждый запрос – число Ni (0 ≤ Ni < 8∙M) , являющееся количеством импульсов от начала передачи.

Выходные данные

Содержать K строк, каждая из которых содержит ответ для соответствующего запроса – два числа через пробел, соответственно, состояние сдвиговых регистров Master и Slave.

Примеры

|  |  |
| --- | --- |
| Ввод | Вывод |
| 2  255 1  0 128  4  0 4 8 9 | 255 0  240 15  1 128  3 0 |
| 2  255 1  0 128  5  9 4 0 8 10 | 3 0  240 15  255 0  1 128  6 0 |

1. **Арифметический корень**

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Достаточно часто из-под знака радикала, когда он соответствует операции взятия арифметического корня второй степени (квадратного корня), требуется вынести множитель.

Пусть имеется выражение sqrt(x), где sqrt – функция взятия квадратного корня, x – подкоренное числовое значение (x – целое неотрицательное число). Требуется написать программу, которая будет выводить одно целое положительное значение b, если b2 = x, или выводить два целых числа b и d, если b2 · d = x, при этом b и d должны быть больше 1 (единицы) и d должно быть минимально возможным. Если уложиться в указанные ограничения невозможно, следует вывести одно слово «Impossible» (без кавычек).

Входные данные

Одно число x (1 ≤ x ≤ 109).

Выходные данные

В единственной строке ответ на задачу. В случае необходимости вывода двух чисел, отделить их друг от друга пробелом, сначала вывести b, затем d.

Примеры

|  |  |
| --- | --- |
| Ввод | Вывод |
| 7 | Impossible |
| 12 | 2 3 |
| 4 | 2 |

1. **Базис**

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Координаты двух точек A и B заданы на плоскости в базисе, определённом двумя неколлинеарными векторами g и f. Вектора g и f заданы в прямоугольной системе координат. Требуется найти длину отрезка в прямоугольной системе координат, концами которого являются точки A и B.

Входные данные

В первой строке дано через пробел два целых числа gx и gy (–109 ≤ gx, gy ≤ 109) – координаты вектора g. Координаты вектора f даны аналогичным образом во второй строке. В третьей и четвёртой строке, подобным же образом, указаны координаты двух точек в базисе векторов g и f.

Выходные данные

Одно единственное число – длина отрезка образованного точками A и B. Ответ требуется вывести с точностью не хуже 10-6.

Примеры

|  |  |
| --- | --- |
| Ввод | Вывод |
| 1 1  1 -1  1 0  0 1 | 2 |
| 1 0  0 1  1 1  0 0 | 1.4142136 |
| 2 3  -1 -2  1 -2  6 8 | 5.000000 |