

Задача А. Обратный элемент по модулю

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

Обратным элементом к n в кольце вычетов по модулю m называется такой элемент x , что выполняется равенство $nx \equiv 1 \pmod{m}$.

Формат входных данных

Входной файл содержит два целых числа n и m ($1 \leq n, m \leq 10^9$).

Формат выходных данных

В выходной файл выведите обратный элемент к n в кольце вычетов по модулю m . Если этого элемента не существует, то выведите -1 .

Примеры

<code>inv.in</code>	<code>inv.out</code>
1 2	1
1 5	1
2 4	-1

Задача В. Диофантово уравнение

Имя входного файла: `dioph.in`
Имя выходного файла: `dioph.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайта

Даны натуральные числа a , b и c . Решите в целых числах уравнение $ax+by=c$. Среди множества решений следует выбрать такое, где x имеет наименьшее неотрицательное значение.

Формат входных данных

Входной файл содержит три целых числа a и b и c ($1 \leq a, b, c \leq 10^9$).

Формат выходных данных

В выходной файл выведите искомые x и y через пробел. Если решения не существует, выведите одну строку «Impossible».

Примеры

<code>dioph.in</code>	<code>dioph.out</code>
1 2 3	1 1

Задача С. Китайская теорема

Имя входного файла: `chine.in`
Имя выходного файла: `chine.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайта

Решите в целых числах систему уравнений

$$\begin{cases} x \equiv a \pmod{n} \\ x \equiv b \pmod{m}, \end{cases}$$

где n и m взаимно просты. Среди решений следует выбрать наименьшее неотрицательное число.

Формат входных данных

Первая строка входных данных содержит число N , $1 \leq N \leq 10^4$, — количество тестов, для которых нужно решить задачу.

Следующие N строк содержат по четыре целых числа a_i, b_i, n_i и m_i ($1 \leq n_i, m_i \leq 10^6$, $0 \leq a_i < n_i$, $0 \leq b_i < m_i$).

Формат выходных данных

Для каждого из тестов выведите искомое наименьшее неотрицательное число x_i .

Примеры

<code>chine.in</code>	<code>chine.out</code>
2	3
1 0 2 3	38
3 2 5 9	

Задача D. Проверка на простоту

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

Проверьте, являются ли данные числа простыми.

Формат входных данных

Программа получает на вход одно целое число k , $1 \leq k \leq 100$. Следующие k строк содержат по одному числу n_i , $2 \leq n_i \leq 10^{18}$.

Формат выходных данных

Для каждого из чисел n_i программа должна вывести одну строку: **YES** для простого числа и **NO** для составного.

Примеры

<code>prime.in</code>	<code>prime.out</code>
2	YES
3	NO
4	

Задача Е. Разложение на множители

Имя входного файла: `factor.in`
Имя выходного файла: `factor.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дано натуральное число N . Известно, что оно является произведением двух простых чисел. Разложите его на множители.

Формат входных данных

Программа получает на вход одно целое число N , $4 \leq N \leq 10^{18}$.

Формат выходных данных

Программа должна вывести два простых делителя числа N в порядке неубывания.

Примеры

<code>factor.in</code>	<code>factor.out</code>
15	3 5

Задача F. Циклический шифр

Имя входного файла: `circular-cipher.in`
Имя выходного файла: `circular-cipher.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Вам задан набор из n последовательностей. Каждая из последовательностей состоит из целых положительных чисел, не превосходящих m . Все числа внутри одной последовательности различны, но одно и то же число может встречаться в разных последовательностях. Длина i -й последовательности равна k_i .

Раз в секунду числа в каждой последовательности циклически сдвигаются на одну позицию влево, то есть числа на позициях $i > 1$ переходят на позиции $i - 1$, а первое число становится последним.

Каждую секунду будем выписывать первое число каждой последовательности в новый массив. Для всех чисел 1 до m найдем самый длинный **подотрезок** этого массива, все элементы которого равны этому числу.

Будем проделывать эту операцию на протяжении 10^{100} секунд. Для каждого числа от 1 до m определите самый длинный из подотрезков, найденных за это время.

Формат входных данных

В первой строке входных данных записаны два числа n и m ($1 \leq n, m \leq 100\,000$) — количество последовательностей и максимальное число, которое может встретиться в последовательностях.

В следующих n строках даны сами последовательности. В каждой строке сначала записано число k_i ($1 \leq k_i \leq 40$) — количество чисел в последовательности, а затем ещё k_i целых положительных чисел — сама последовательность. Гарантируется, что числа в каждой последовательности попарно различны и не превосходят m .

Суммарная длина всех последовательностей не превосходит 200 000.

Формат выходных данных

Выведите m чисел, i -е из которых равняется длине самого большого подотрезка, все числа в котором равны i и который встретился в выписываемом массиве за первые 10^{100} секунд.

Примеры

<code>circular-cipher.in</code>	<code>circular-cipher.out</code>
3 4	2
3 3 4 1	1
4 1 3 4 2	3
3 3 1 4	2
5 5	3
2 3 1	1
4 5 1 3 2	4
4 2 1 3 5	0
1 3	1
2 5 3	
4 6	0
3 4 5 3	0
2 6 3	2
2 3 6	1
3 3 6 5	1
	2