

Задача А. НОД

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Формат входных данных

Входной файл содержит два натуральных числа a и b ($1 \leq a, b \leq 10^{18}$);

Формат выходных данных

В выходной файл выведите наибольший общий делитель этих двух этих чисел.

Примеры

<code>stdin</code>	<code>stdout</code>
3 5	1
3 9	3

Задача В. Решето Эратосфена

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

По введённым числам A и B вывести все простые числа в интервале от A до B включительно.

Формат входных данных

В единственной строке вводятся два числа $1 \leq A \leq B \leq 100\,000$

Формат выходных данных

Вывести в одну строку все простые числа в интервале от A до B включительно

Примеры

<code>sieve.in</code>	<code>sieve.out</code>
2 2	2
1 100	2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

Задача С. Делители

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

Вася любит простые числа, а особенно он любит простые числа, на которые делится его самое любимое число K .

Формат входных данных

На вход подается единственное целое положительное число K отличное от 1, не превосходящее $3 * 10^9$.

Формат выходных данных

В первой строке выведите количество чисел, которые Вася особенно любит. Во второй строке выведите в порядке возрастания все простые числа, которые Вася особенно любит.

Примеры

<code>onedivisor.in</code>	<code>onedivisor.out</code>
5	1 5
12	2 2 3
30	3 2 3 5

Задача D. Возведение в степень

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Формат входных данных

Во входном файле даны три натуральных числа A, B, M ($1 \leq A, B \leq 10^9, 2 \leq M \leq 10^9$), записанные на одной строке через пробел.

Формат выходных данных

В выходной файл выведите одно число, равное $A^B \bmod M$ (`mod` означает взятие остатка при делении).

Примеры

<code>stdin</code>	<code>stdout</code>
2 3 100	8

Замечание

При решении данной задачи нельзя использовать встроенную операцию возведения в степень.

Задача Е. МегаНОД

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Дано несколько чисел. Найти самое большое число, на которое делятся все несколько чисел.

Формат входных данных

В единственной строке даны через пробел несколько чисел ($1 \leq \text{несколько} \leq 1000$, $1 \leq \text{каждое} \leq 10^9$).

Формат выходных данных

Выведите искомое число.

Примеры

<code>stdin</code>	<code>stdout</code>
18 30 21	3

Задача F. Обратный элемент по модулю

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
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>stdin</code>	<code>stdout</code>
1 2	1
1 5	1
2 4	-1

Задача G. Планета Арифмет

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

Где-то в далекой галактике есть планета Арифмет. Населяют ее небольшие, человеко-подобные существа — Арифметяне. В толще планеты находится единственный ценный ресурс — Числы. Числы — это большие красивые кристаллы, состоящие из слипшихся Цифров. Каждая Числа имеет эквивалент в обычных, Земных числах. Многие века Арифметяне добывают Числы, которые необходимы им, чтобы выжить. Но однажды на их планету упала Белая Вычислительная коробка С Надкушенным Яблоком и сказала им человеческим голосом: «Простые Числы вкуснее». Однако после падения коробка повредилась и забыла, что такое Простые Числы. Вам, как самому умному Арифметянину предстоит разобраться с этой коробкой и заставить ее говорить, вкусная ли данная ей Числа. Единственная зацепка, которая у вас есть — это уцелевшие на коробке древние письмена:

```
def is_prime(n):  
    # Jwo oISJi skjewo ...  
  
number = int(input())  
print(is_prime(number))
```

Восстановите повреждённую часть!

Формат входных данных

В коробку вводится одно число N ($1 \leq N \leq 10^9$).

Формат выходных данных

Коробка должна сказать `True`, если Числа вкусная, и `False`, если Числа невкусная.

Примеры

<code>stdin</code>	<code>stdout</code>
2	True
10	False

Задача Н. Диофантово уравнение

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
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>stdin</code>	<code>stdout</code>
1 2 3	1 1

Задача I. RSA. Взлом RSA

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

В 1977 году Ronald Linn Rivest, Adi Shamir и Leonard Adleman предложили новую криптографическую схему RSA, используемую до сих пор. RSA является криптосистемой с открытым ключом: зашифровать сообщение может кто угодно, знающий общеизвестный открытый ключ, а расшифровать сообщение — только тот, кто знает специальный секретный ключ.

Желающий использовать систему RSA для получения сообщений должен сгенерировать два простых числа p и q , вычислить $n = pq$ и сгенерировать два числа e и d такие, что $ed \equiv 1 \pmod{(p-1)(q-1)}$ (заметим, что $(p-1)(q-1) = \varphi(n)$). Числа n и e составляют открытый ключ и являются общеизвестными. Число d является секретным ключом, также необходимо хранить в тайне и разложение числа n на простые множители, так как это позволяет вычислить секретный ключ d .

Сообщениями в системе RSA являются числа из \mathbb{Z}_n . Пусть M — исходное сообщение. Для его шифрования вычисляется значение $C = M^e \pmod n$ (для этого необходимо только знание открытого ключа). Полученное зашифрованное сообщение C передается по каналу связи. Для его расшифровки необходимо вычислить значение $M = C^d \pmod n$, а для этого необходимо знание секретного ключа.

Вы перехватили зашифрованное сообщение C и знаете только открытый ключ: числа n и e . “Взломайте” RSA — расшифруйте сообщение на основе только этих данных.

Формат входных данных

Программа получает на вход три натуральных числа: n , e , C , $n \leq 10^9$, $e \leq 10^9$, $C < n$. Числа n и e являются частью какой-то реальной схемы RSA, т.е. n является произведением двух простых и e взаимно просто с $\varphi(n)$. Число C является результатом шифрования некоторого сообщения M .

Формат выходных данных

Выведите одно число M ($0 \leq M < n$), которое было зашифровано такой криптосхемой.

Примеры

<code>rsa.in</code>	<code>rsa.out</code>
143 113 41	123
9173503 3 4051753	111111

Задача J. Китайская теорема

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

Входной файл содержит четыре целых числа a , b , n и m ($1 \leq n, m \leq 10^6$, $0 \leq a < n$, $0 \leq b < m$).

Формат выходных данных

В выходной файл выведите искомое наименьшее неотрицательное число x .

Примеры

<code>stdin</code>	<code>stdout</code>
1 0 2 3	3
3 2 5 9	38

Задача К. Вычислите функции

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

Дано число N . Требуется вычислить следующие функции для него:

$\varphi(N)$ = количество взаимно простых с N чисел среди $1, 2, \dots, N$

$\tau(N)$ = количество делителей числа N

$\sigma(N)$ = сумма всех делителей числа N

Формат входных данных

Во входном файле содержится единственное число $1 \leq N \leq 10^9$.

Формат выходных данных

В единственную строку выходного файла выведите через пробел три числа — значения $\varphi(N)$, $\tau(N)$, $\sigma(N)$.

Примеры

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

Задача L. Разложение на простые множители - 2

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

По введённому числу N выведите его разложение на простые множители.

Формат входных данных

В единственной строке вводится единственное число $2 \leq N \leq 1\,000\,000$.

Формат выходных данных

Выведите в каждой строке по два числа. Первое число — делитель, второе — степень, с которой он входит в разложение числа. Если степень равна 0, то выводить делитель и степень не надо. Пары чисел выводить по возрастанию делителя. Внимательно прочитайте примеры.

Примеры

<code>stdin</code>	<code>stdout</code>
24	2 3 3 1
2	2 1

Замечание

В тесте 1 : $24 = 2^3 * 3^1$

В тесте 2 : $2 = 2^1$

Задача М. Обратный элемент по простому модулю

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

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

Формат входных данных

Входной файл содержит два целых числа a и p ($1 \leq a, p \leq 10^9$), причем p — простое.

Формат выходных данных

В выходной файл выведите обратный элемент к a в кольце вычетов по модулю p .

Примеры

<code>inv.in</code>	<code>inv.out</code>
1 2	1