Задача A. 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 ($1 \leqslant n, m \leqslant 10^9$).

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

stdin	stdout
1 2	1
1 5	1
2 4	-1

Задача B. 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 ($1 \leqslant a, b, c \leqslant 10^9$).

Формат выходных данных

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

Примеры

stdin	stdout
1 2 3	1 1

Задача C. Простые сложности

Имя входного файла:	`again.in`
Имя выходного файла:	`again.out`
Ограничение по времени:	5 секунды
Ограничение по памяти:	256 мегабайта

В этой жизни не всё так просто. Особенно числа. Вам дан набор чисел. Необходимо для каждого из них определить, является ли оно простым.

Формат входных данных

В первой строке входных данных содержится единственное число $1 \leqslant T \leqslant 5\,000$ — количество чисел, которые необходимо проверить на простоту. Далее содержится T целых положительных чисел, не превосходящих 10^{18}.

Формат выходных данных

В i-й строке выходных данных должно быть записано «`YES`», если i-е число является простым, и «`NO`» в противном случае.

Примеры

again.in	again.out
2 3 4	YES NO

Задача D. 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, $4 \leqslant N \leqslant 10^{18}$.

Формат выходных данных

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

Примеры

factor.in	factor.out
15	3 5

Задача E. 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:

$$\varphi(N) = \text{количество взаимно простых с } N \text{ чисел среди } 1, 2, \ldots, N$$

$$\tau(N) = \text{количество делителей числа N}$$

$$\sigma(N) = \text{сумма всех делителей числа N}$$

Формат входных данных

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

Формат выходных данных

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

Примеры

func.in	func.out
2	1 2 3

Задача F. 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 $C = M^e \bmod n$ (для этого необходимо только знание открытого ключа). Полученное зашифрованное сообщение C передается по каналу связи. Для его расшифровки необходимо вычислить значение $M = C^d \bmod n$, 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, $n \leqslant 10^9$, $e \leqslant 10^9$, $C < n$. 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 $(0 \leqslant M < n)$, которое было зашифровано такой криптосхемой.

Примеры

rsa.in	rsa.out
143 113 41	123
9173503 3 4051753	111111