

Задача А. Будильники

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

64 мегабайта

Петя и Ваня живут в одной комнате главного корпуса ЛКШ и каждое утро встают пораньше, чтобы как следует размяться на альтернативной зарядке. Разумеется, мальчики ставят оба своих будильника на одно и то же время, поскольку сигнала одного будильника недостаточно, чтобы разбудить среднестатистического школьника. При этом, в случае, если Пете и Ване все-таки удастся проспать, Петин будильник будет звонить каждые n минут, а Васин — каждые m минут. Напишите программу, которая поможет вычислить, через сколько минут оба будильника прозвонят одновременно во второй раз.

Формат входных данных

В единственной строке даны два целых числа n и m ($1 \leq n, m \leq 10^{1000}$).

Формат выходных данных

Ваша программа должна вывести одно число — время в минутах, через которое оба будильника прозвонят одновременно во второй раз.

Примеры

стандартный ввод	стандартный вывод
2 3	6
6 21	42

Задача В. Планета Арифмет

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

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 ($2 \leq N \leq 10^9$).

Формат выходных данных

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

Примеры

стандартный ввод	стандартный вывод
2	True
10	False

Задача С. Разложение на множители

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 64 мегабайта

64 мегабайта

Дано число. Требуется разложить его на простые множители.

Формат входных данных

Вводится число N ($2 \leq N \leq 10^9$).

Формат выходных данных

Выведите через пробел разложение на простые множители в порядке возрастания множителей.

Примеры

стандартный ввод	стандартный вывод
17	17
60	2 2 3 5

Задача D. МегаНОД

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

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.

Пример

стандартный ввод	стандартный вывод
18 30 21	3

Задача Е. Часовая башня

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Многие ученики Хогвартса разозлились на то, что Амбридж запретила собираться в группы из трёх и более учеников, и решили придумать, как ей насолить. Один из планов состоял в том, чтобы прийти в Часовую башню и сломать имеющиеся там часы.

Часы состоят из N последовательно соединённых шестерней, то есть первая шестерня зацепляет вторую, вторая зацепляет третью и так далее... $(N - 1)$ -я шестерня зацепляет последнюю N -ю.

Перед тем как сломать часы, ученики сначала хотят разобраться, как они работают. Им интересно, сколько раз нужно полностью повернуть первую шестерню, чтобы все остальные шестерни вернулись в изначальное положение.

Формат входных данных

В первой строке вводится целое число N ($2 \leq N \leq 10$). В следующей строке вводится N целых чисел a_i — число зубчиков в каждой шестерне ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите минимальное число полных поворотов первой шестерни, после которой все остальные шестерни тоже возвращаются в изначальное положение. Гарантируется, что произведение первого числа и ответа не превышает 10^9 .

Пример

стандартный ввод	стандартный вывод
3 10 20 30	6

Задача F. Решето Эратосфена

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

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

Формат входных данных

В единственной строке вводятся два числа $1 \leq A \leq B \leq 1000000$

Формат выходных данных

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

Примеры

стандартный ввод	стандартный вывод
2 2	2
1 100	2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67

Задача G. Факториал

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Факториалом $N!$ числа называется произведение всех натуральных чисел от 1 до N . Также, для удобства, считается, что $0! = 1$. В этой задаче от вас требуется представить число $N!$ в виде произведения $p_1^{q_1} * p_2^{q_2} * \dots * p_m^{q_m}$, где p_i - различные простые, а q_i - положительные целые числа.

Формат входных данных

На вход подаётся единственное целое число N ($1 \leq N \leq 1\,000\,000$), факториал которого необходимо разложить на множители.

Формат выходных данных

В первой строке нужно вывести количество множителей в разложении. В остальных — множители, по одному в строке в порядке возрастания основания. При этом в каждой строке нужно через пробел указать основание и положительную степень каждого множителя.

Пример

стандартный ввод	стандартный вывод
6	3 2 4 3 2 5 1

Задача Н. Марсианские факториалы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

64 мегабайта

В 3142 году очередная экспедиция на Марс обнаружила в одной из пещер таинственные знаки. Они однозначно доказывали существование на Марсе разумных существ. Однако смысл этих таинственных знаков долгое время оставался неизвестным. Недавно один из ученых, профессор Очень-Умный, заметил один интересный факт: всего в надписях, составленных из этих знаков, встречается ровно K различных символов. Более того, все надписи заканчиваются на длинную последовательность одних и тех же символов.

Вывод, который сделал из своих наблюдений профессор, потряс всех ученых Земли. Он предположил, что эти надписи являются записями факториалов различных натуральных чисел в системе счисления с основанием K . А символы в конце — это конечно же нули, ведь, как известно, факториалы больших чисел заканчиваются большим количеством нулей. Например, в нашей десятичной системе счисления факториалы заканчиваются на нули, начиная с $5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5$. А у числа $100!$ в конце следует 24 нуля в десятичной системе счисления и 48 нулей в системе счисления с основанием 6 — так что у предположения профессора есть разумные основания!

Теперь ученым срочно нужна программа, которая по заданным числам N и K найдет количество нулей в конце записи в системе счисления с основанием K числа $N! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot (N-1) \cdot N$, чтобы они могли проверить свою гипотезу. Вам придется написать им такую программу!

Формат входных данных

В первой строке входных данных содержатся числа N и K , разделенные пробелом, ($1 \leq N \leq 10^9$, $2 \leq K \leq 1000$).

Формат выходных данных

Выведите число X — количество нулей в конце записи числа $N!$ в системе счисления с основанием K .

Примеры

стандартный ввод	стандартный вывод
5 10	1
100 6	48
100 10	24
3 10	0

Задача I. Количество делителей

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Для улучшения качества обучения в параллели С' преподаватели решили ввести индекс счастья ученика. В начале смены индекс счастья каждого ученика равен 1. Каждый последующий день смены индекс счастья может увеличиться или уменьшиться в m раз. Статистика показывает, что количество решенных задач на практике прямо пропорционально количеству делителей индекса счастья, поэтому бывают дни, когда индекс счастья не изменяется, а преподаватели считают количество делителей у индекса счастья.

Но так как процесс умножения, деления и подсчета количества делителей достаточно тяжелый, то вам поручено автоматизировать процесс вычислений.

Формат входных данных

В первой строке задается число n ($1 \leq n \leq 10^4$) — количество дней в смене. Каждая из последующих n строк содержит строчку запроса. Строчка запросов имеет один из трех видов: «?», «* m », «/ m » ($1 \leq m \leq 10^{100}$), все простые числа, на которые делится m , не превосходят 10^4 . На запросы второго и третьего вида нужно, соответственно, умножить индекс счастья на m и разделить на m . На запрос первого вида нужно вывести количество делителей индекса счастья, если индекс целый и строчку "Current number is not integer!", если индекс не целый.

Количество операций «* m », «/ m » в сумме не превышает 10^4 .

Формат выходных данных

Для каждого запроса типа «?» выведите одно число — количество делителей, или "Current number is not integer!", если соответствующее число не целое.

Примеры

стандартный ввод	стандартный вывод
3 * 4 * 3 ?	6
4 * 6 ? / 10 ?	4 Current number is not integer!

Задача J. Сильно составные числа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Выведите все числа в диапазоне от 2 до N , у которых есть хотя бы три различных простых делителя.

Формат входных данных

Вводится одно натуральное число N ($N \leq 10000000$).

Формат выходных данных

Выведите через пробел в возрастающем порядке все искомые числа.

Пример

стандартный ввод	стандартный вывод
50	30 42