

Задача А. Функция printlist

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

Напишите функцию `printlist`, которая получает на вход 4 аргумента:

- список `s`
- строку `before`: что печатать в начале списка
- строку `between`: что печатать между элементами списка
- строку `after`: что печатать после списка

и выводит на экран список `s` в указанном формате. Вставьте вашу функцию в начало следующей программы и сдайте ее на проверку:

```
s = input().split()
printlist(s, 'x=', '*', '.')
printlist(s, '', ' ', '')
printlist(s, '', '', '')
printlist(s, '"', '"', '"', '"')
```

Не забывайте, что список может состоять из 0 или 1 элемента!

Формат входных данных

Вам не надо об этом задумываться.

Формат выходных данных

И об этом тоже.

Примеры

stdin	stdout
abcd e fg	x=abcd*e*fg. abcd e fg abcdefg "abcd","e","fg"

Задача В. 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 письма:

```
def is_prime(n):
    # Jwo oISJi skjewo ...

chisla = int(input())
print(is_prime(chisla))
```

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

stdin	stdout
2	True
10	False

Задача С. Разложение на множители

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 0.25 секунда
Ограничение по памяти: 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

stdin	stdout
17	17
60	2 2 3 5

Задача D. 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

stdin	stdout
18 30 21	3

Задача E. Шестерёнки

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

Даны две сцепленные шестерёнки. У одной шестерёнки N зубцов, у другой — K . Требуется найти, какое минимальное число поворотов на один зубчик требуется сделать, чтобы шестерёнки вернулись в исходное состояние.

Формат входных данных

В единственной строке — два числа, N и K . $1 \leq N, K \leq 10^{1000}$.

Формат выходных данных

Выведите искомое количество поворотов.

Примеры

stdin	stdout
2 3	6
6 21	42

Задача F. Решето Эратосфена

Имя входного файла: `sieve.in`
Имя выходного файла: `sieve.out`
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 $1 \leq A \leq B \leq 100000$

Формат выходных данных

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

Примеры

sieve.in	sieve.out
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 G. Количество делителей

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
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 ($1 \leq n \leq 10^3$) — 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

Количество делителей

Примеры

stdin	stdout
3 * 4 * 3 ?	6
4 * 6 ? / 10 ?	4 Current number is not integer!