

Функции и рекурсия.

А. Сумма делителей числа

Для данного натурального числа N ($1 \leq N < 10^{10}$) требуется вычислить сумму его делителей, меньших самого числа.

Решение оформите в виде функции `SumDivisors`, принимающей на вход натуральное число и возвращающей сумму его делителей, отличных от него самого.

Решение должно иметь сложность $O(\sqrt{N})$.

Input	Output
12	16

В. Дружественные числа

Дружественные числа — это два различных натуральных числа, таких, что сумма всех делителей одного числа (меньших самого этого числа) равна другому числу, и наоборот (дружественными являются, например, 220 и 284).

Напишите функцию `IsFriend`, которая проверяет пару чисел на «дружественность» и возвращает логическое значение `true`, если пара чисел дружественная и `false` в противном случае. Функция `IsFriend` должна использовать функцию `SumDivisors` из предыдущей задачи.

Ваша программа должна вывести строку `YES`, если пара чисел является дружественной, и строку `NO` в противном случае.

Input	Output
220 284	YES

С. Дружественные числа в диапазоне

Дружественные числа — это два натуральных числа, таких, что сумма всех делителей одного числа (меньших самого этого числа) равна другому числу, и наоборот (дружественными являются, например, 220 и 284).

Напишите программу, которая находит все пары не равных друг другу дружественных чисел в заданном диапазоне. Используйте функцию, которая вычисляет сумму делителей числа.

На вход программе подаётся две строки с натуральными числами a и b ($a < b$).

Программа должна вывести пары различных дружественных чисел, каждое из которых находится на отрезке $[a, b]$.

В каждой паре сначала выводится меньшее число. Пары чисел должны выводиться в порядке возрастания первого числа из пары и разделяться запятой. Каждая пара заключена в скобки.

В случае, если таких пар в указанном диапазоне нет, вывести число 0.

Input	Output
1000 5000	(1184, 1210) (2620, 2924)

D. Сумма чисел

Дана последовательность чисел, завершающаяся числом 0. Найдите сумму всех этих чисел, не используя цикл и массивы.

Input	Output
1 2 3 0	6

E. Разворот последовательности

Дана последовательность целых чисел, заканчивающаяся числом 0. Выведите эту последовательность в обратном порядке.

Решение этой задачи при помощи рекурсии позволяет обойтись без списков, строк и прочих структур данных для сохранения всех введённых чисел.

Input	Output
1 2 3 0	0 3 2 1

Ф. Алгоритм Евклида

Даны два неотрицательных целых числа A и B , ($1 \leq A, B \leq 10^{18}$). Требуется найти их наибольший общий делитель при помощи уже известного вам алгоритма Евклида.

На вход программе подаётся два целых неотрицательных числа. Программа должна вывести одно число — ответ на вопрос задачи.

Input	Output
12 16	4

Г. Быстрое возведение в степень

Дано вещественное число $a \neq 0$ и неотрицательное целое n ($n \leq 10^9$). Вычислите a^n .

Указание: воспользуйтесь тождествами $a^{2n} = (a^2)^n$ и $a^{2n+1} = a^{2n} \cdot a$

Input	Output
1.000000001 1000000000	2.7182820387553908

Н. Количество вызовов функции (числа Фибоначчи)

Как известно, очередное число Фибоначчи равно сумме предыдущих двух. Первое и второе число Фибоначчи равны единице.

Программист Вася написал вычисление n -ого числа Фибоначчи с помощью рекурсивной функции, которая выглядит следующим образом:

```
def fibonacci(n):  
    if n < 3:  
        return 1  
    else:  
        return fibonacci(n - 1) + fibonacci(n - 2)
```

Сколько раз запустится эта функция прежде, чем будет получено значение?

На вход программе подаётся одно натуральное число. Программа должна вывести одно натуральное число — ответ на вопрос задачи. Гарантируется, что ответ не превосходит 10^{18} .

Input	Output
3	3
10	109
57	730870592323

И. Фишки

Дана полоска из клеток, пронумерованных от 1 до N . На каждом ходе разрешено поставить фишку на клетку (если её там еще нет) или снять фишку с клетки (если она там есть). При этом, можно выбрать не любую клетку, а только клетку под номером 1 или клетку с номером на 1 больше, чем у первой занятой клетки.

Изначально полоска пуста. Требуется занять все клетки.

Программа должна вывести последовательность номеров клеток, с которыми совершается действие. Если фишка снимается, то номер клетки должен выводиться со знаком минус.

Количество действий не должно превышать 10^4 . Если существует несколько возможных решений задачи, то разрешается вывести любое.

Input	Output
3	1 2 -1 3 1

Ж. Ханойские башни

Головоломка «Ханойские башни» состоит из трех стержней, пронумерованных числами 1, 2, 3. На стержень 1 надета пирамидка из n дисков различного диаметра в порядке убывания диаметра (наверху самый маленький диск). Диски можно перекладывать с одного стержня на другой по одному, при этом диск нельзя класть на диск меньшего диаметра. Необходимо переложить всю пирамидку со стержня 1 на стержень 3 за минимальное число перекладываний.

Напишите программу, которая решает головоломку; для данного числа дисков n печатает последовательность перекладываний в формате $a\ b\ c$, где a — номер перекладываемого диска, b — номер стержня с которого снимается данный диск, c — номер стержня на который надевается данный диск.

Например, строка $1\ 2\ 3$ означает перемещение диска номер 1 со стержня 2 на стержень 3. В одной строке печатается одна команда. Диски пронумерованы числами от 1 до n в порядке возрастания диаметров.

Input	Output
2	1 1 2 2 1 3 1 2 3

К* Ремонт в Ханое

Решите задачу U со следующим ограничением: запрещено перекладывать диски со стержня 1 на стержень 3 и наоборот.

Вам не нужно находить минимальное решение, но количество совершённых перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10.

Input	Output
2	1 1 2 1 2 3 2 1 2 1 3 2 1 2 1 2 2 3 1 1 2 1 2 3

Л* Циклические башни

Решите задачу U со следующим ограничением: диск со стержня 1 можно перекладывать только на стержень 2, со стержня 2 на 3, а со стержня 3 на 1.

Вам не нужно находить минимальное решение, но количество совершённых перемещений не должно быть больше 200000, при условии, что количество дисков не превосходит 10.

Input	Output
2	1 1 2 1 2 3 2 1 2 1 3 1 2 2 3 1 1 2 1 2 3