

**Программирование, 9-й математический класс**  
**Листок 5: Синтаксис Python. Цикл while.**

Во всех задачах листка нельзя пользоваться операцией возведения в степень (\*\*), кроме случаев, когда требуется вычисление квадратного корня.

**A. Факториал**

Дано  $n \geq 0$ , вычислить  $n!$

Input	Output
1	1

**B. Сумма цифр числа**

Найти сумму цифр натурального числа.

Input	Output
12345	15

**C. Степень числа**

Дано целое число  $a$  и целое неотрицательное число  $n$ . Вычислить  $a^n$ .

Input	Output
2	8
3	

**D. Число Фибоначчи**

Найти  $n$ -е число Фибоначчи ( $\phi_0 = 0, \phi_1 = 1, \phi_n = \phi_{n-1} + \phi_{n-2}$ ).

Input	Output
6	8

**E. Наименьший натуральный делитель**

Дано целое число  $N$ , не меньшее 2. Выведите его наименьший натуральный делитель, отличный от 1. Время работы программы должно быть пропорционально  $\sqrt{N}$ .

Input	Output
15	3

**F. Точная степень двойки**

Дано натуральное число  $N$ . Выведите слово YES, если число  $N$  является точной степенью двойки, или слово NO в противном случае.

Операцией возведения в степень пользоваться нельзя. В этой задаче можно обойтись одним циклом и одним условным оператором. Условный оператор разрешается использовать только вне цикла.

Input	Output
1	YES

**G. Двоичный логарифм**

По данному натуральному числу  $N$  выведите такое наименьшее целое неотрицательное число  $k$ , что  $2^k \geq N$ .

Операцией возведения в степень пользоваться нельзя.

Input	Output
1	0

## Н. Номер числа Фибоначчи

Последовательность Фибоначчи определяется так:

$$\phi_0 = 0, \phi_1 = 1, \phi_2 = 1, \dots, \phi_N = \phi_{N-1} + \phi_{N-2}$$

Дано натуральное число  $A$ . Определите, каким по счету числом Фибоначчи оно является, то есть выведите такое число  $n$ , что  $\phi_n = A$ .

Если  $A$  не является числом Фибоначчи, выведите число  $-1$ .

Можно рассчитывать, что число 1 на вход подаваться не будет.

Input	Output
8	6

## I. Алгоритм Евклида

Алгоритм Евклида для вычисления наибольшего общего делителя двух натуральных чисел, формулируется так: нужно заменять большее число на остаток от деления большего на меньшее до тех пор, пока остаток не станет равно нулю; тогда второе число и есть НОД.

Напишите программу, которая реализует этот алгоритм.

Даны два натуральных числа:  $a$  и  $b$ .

Программа должна вывести в одной строке два числа: сначала наибольший общий делитель двух введенных чисел, а затем — количество шагов цикла, которые были выполнены.

Input	Output
21 14	7 2

## J. Исполнитель "Раздвоитель"

Исполнитель «Раздвоитель» преобразует натуральные числа. У него есть две команды: «Вычесть 1» и «Разделить на 2», первая команда уменьшает число на 1, вторая команда уменьшает число в два раза, если оно чётное, иначе происходит ошибка.

Дано два натуральных числа  $A$  и  $B$  ( $A > B$ ). Напишите алгоритм для Раздвоителя, который преобразует число  $A$  в число  $B$  и при этом содержит минимальное число команд. Команды алгоритма нужно выводить по одной в строке, первая команда обозначается, как  $-1$ , вторая команда как  $:2$ .

Input	Output
179 20	-1 :2 -1 :2 :2 -1 -1

## К. Утренняя пробежка

В первый день спортсмен пробежал  $x$  километров, а затем он каждый день увеличивал пробег на 70% от предыдущего значения. По данному числу  $y$  определите номер дня, на который пробег спортсмена составит не менее  $y$  километров.

На вход программа получает два действительных числа  $x$  и  $y$ .

Input	Output
10 30	4

## L\* Количество целочисленных точек в круге

Дано натуральное число  $R \leq 10^5$ . Определите количество целочисленных точек, находящихся внутри и на границе круга радиуса  $R$  с центром в начале координат.

Сложность алгоритма должна быть  $O(R)$ .

При решении этой задачи нельзя пользоваться функцией вычисления квадратного корня.

Input	Output
2	13

М\* *Быстрое возведение в степень*

Решите задачу С для возведения вещественного числа в неотрицательную степень, используя следующие тождества:

$$\begin{aligned}a^{2k} &= (a^2)^k \\ a^{2k+1} &= a^{2k} \cdot a\end{aligned}$$

Сформулируйте цикл, поддерживающий следующий инвариант:  $a^k \cdot res = a^N$  с такими первоначальными значениями:  $k = N$  и  $res = 1$ .

Если вы всё сделаете правильно, что программа сможет возводить в степень очень быстро, например, для возведения в степень  $n = 1000000$  потребуется не более 40 умножений.

Input	Output
2	2
1	

Н\* *Разложение на простые множители*

Выписать разложение числа  $N$  ( $2 \leq N \leq 2 \cdot 10^9$ ) на простые множители.

Сложность алгоритма  $O(\sqrt{N})$ .

Input	Output
18	2 3 3

О\* *Расширенный алгоритм Евклида*

Даны два натуральных числа  $a$  и  $b$ . Найдите их наибольший общий делитель  $d$  и любые два целых числа  $x$  и  $y$ , таких, что  $ax + by = d$ . Программа должна вывести числа  $d, x, y$ .

*Указание:* рассмотреть систему уравнений

$$\begin{cases} a \cdot 1 + b \cdot 0 = a \\ a \cdot 0 + b \cdot 1 = b \end{cases}$$

Затем для правых частей уравнений вычислять НОД( $a, b$ ), используя тождество

$$a \% b = a - b * (a // b)$$

а для левых частей выполнять соответствующие преобразования над коэффициентами при  $a$  и  $b$ .

Input	Output
1	1 0 1
1	

Р\* *Количество чисел с чётной суммой цифр на отрезке*

Подсчитайте количество натуральных чисел на отрезке от  $A$  до  $B$ , сумма цифр которых чётна.

Вводится два натуральных числа  $A$  и  $B$ ,  $1 \leq A \leq B \leq 10^9$ .

Выведите одно число — количество чисел, больших либо равных  $A$  и меньших либо равных  $B$ , имеющих чётную сумму цифр.

Input	Output
1	2
5	

В задачах Q - Z последовательность целых чисел вводится с клавиатуры, не может содержать число 0 и заканчивается нулём, который служит сигналом окончания ввода последовательности.

В этих задачах не разрешается запоминать последовательность, если вы вдруг уже знакомы с массивами.

Оформлять решение следует таким образом (с точностью до именования переменных):

```
x = int(input())      # чтение первого числа последовательности
...                  # необходимые предварительные присваивания и т.п.
while x != 0:
    ...              # основной алгоритм, в котором нет ни одного оператора чтения input()
    x = int(input()) # оператор чтения очередного элемента завершает тело цикла
...                  # печать результата
```

Кроме того, требуется давать используемым переменным осмысленные имена (например, если это длина, текущая длина, максимум/минимум, предыдущий и т.п.)

#### Q. Максимум последовательности

В последовательности натуральных чисел определить значение наибольшего элемента последовательности.

Input	Output
1	9
7	
9	
0	

#### R. Второй максимум

Последовательность состоит из различных натуральных чисел и завершается числом 0. Определите значение второго по величине элемента в этой последовательности.

Второй по величине элемент — максимальный элемент в последовательности, полученной из исходной удалением одного элемента, равного её максимуму.

Input	Output
1	7
7	
9	
0	

#### S. Количество элементов, равных максимуму

Определите количество элементов последовательности, равных её максимальному элементу.

Input	Output
1	2
4	
4	
3	
0	

#### T. Количество перемен знака

Определить число перемен знака в последовательности целых чисел.

Input	Output
-1	1
-3	
-3	
3	
4	
0	

#### U. Сумма чисел между предпоследней и последней двойками

Вычислить сумму всех элементов последовательности натуральных чисел между предпоследней и последней двойками (не включая сами двойки). Если двоек нет или она одна, вывести число -1.

Input	Output
3	3
2	
1	
2	
3	
2	
1	
0	
0	
0	
0	

V. *Длина наибольшей площадки*

Определить длину наибольшей «площадки» в последовательности (т.е. подпоследовательности подряд идущих одинаковых чисел).

Input	Output
1	2
7	
7	
9	
1	
0	

W. *Длина наибольшего монотонного фрагмента*

Определите наибольшую длину монотонного фрагмента последовательности (то есть такого фрагмента, где все элементы либо больше предыдущего, либо меньше).

Input	Output
1	4
6	
7	
9	
4	
1	
0	

X. *Количество локальных максимумов*

Элемент последовательности называется локальным максимумом, если он строго больше предыдущего и последующего элемента последовательности. Первый и последний элемент последовательности не являются локальными максимумами.

Дана последовательность натуральных чисел, завершающаяся числом 0. Определите количество строгих локальных максимумов в этой последовательности.

Input	Output
1	2
2	
1	
2	
1	
0	

Y. *Наименьшее расстояние между локальными максимумами*

Определите наименьшее расстояние между двумя локальными максимумами последовательности. Если в последовательности нет двух локальных максимумов, выведите число 0.

Начальное и конечное значение при этом локальными максимумами не считаются.

Input	Output
1	2
2	
1	
1	
2	
1	
2	
1	
2	
1	
0	

## Z. Стандартное отклонение

Дана последовательность натуральных чисел  $x_1, x_2, \dots, x_n$ . Стандартным отклонением называется величина:

$$\sqrt{\frac{(x_1 - s)^2 + (x_2 - s)^2 + \dots + (x_n - s)^2}{n - 1}}$$

где  $s = \frac{x_1 + x_2 + \dots + x_n}{n}$  — среднее арифметическое последовательности.

Вычислите стандартное отклонение для данной последовательности натуральных чисел, завершающейся числом 0.

Input	Output
1	4.16333199893
7	
9	
0	