

Цикл while.

A. Количество чисел до первого нуля

Дана последовательность целых чисел по одному в строке. Сколько чисел было введено до первого нуля?

Input	Output
5	3
7	
6	
0	

B. Количество чисел между первыми двумя нулями

Дана последовательность целых чисел. Сколько чисел было введено между первым и вторым нулём? Гарантируется, что в последовательности есть как минимум два нуля.

Input	Output
5	1
0	
6	
0	
8	
8	

C. Сумма чисел до первого нуля

Дана последовательность чисел. Какова сумма чисел до первого нуля?

Input	Output
5	19
6	
8	
0	

D. Сумма чисел между первыми двумя нулями

Дана последовательность чисел. Какова сумма чисел между первым и вторым нулём?

Input	Output
0	14
6	
8	
0	
1	

E. Сумма больше ста (внимательно читайте комментарий к примеру!)

Дана последовательность целых чисел. Сколько чисел было введено перед тем, как их сумма впервые превысила 100? Гарантируется, что хотя бы один раз сумма введённых чисел превысила 100.

Input	Output
20	4
20	
20	
-50	
200	
30	
-10	
20	
30	

Комментарий к тесту: сумма впервые превысила 100 после ввода числа 200. Перед этим было введено 4 числа: 20, 20, 20 и -50.

F. *Сумма между 100 и 200 (внимательно читайте комментарий к примеру!)*

Дана последовательность чисел. Сколько чисел было введено перед тем, как их сумма впервые приняла значение от 100 до 200 (включительно)?

Input	Output
1	7
2	
3	
200	
-1	
-2	
-104	
50	
34	

Комментарий к тесту: сумма впервые попала на отрезок $[100, 200]$ после ввода числа 50. Перед этим было введено 7 чисел: 1, 2, 3, 200, -1, -2, -104.

G. *Сумма цифр числа*

Найти сумму цифр натурального числа.

Input	Output
12345	15

H. *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

I. *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

J. *Ближайшее число Фибоначчи*

Последовательность Фибоначчи определяется так:

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

Дано натуральное число A . Определите ближайшее к нему число Фибоначчи. Если два числа Фибоначчи находятся на равном расстоянии, вывести меньшее.

Input	Output
10	8
21	21

К. Алгоритм Евклида

Алгоритм Евклида для вычисления наибольшего общего делителя (gcd — greatest common divisor) двух натуральных чисел a и b ($a > b$) основан на следующем факте:

$$gcd(a, b) = gcd(a - b, b)$$

Можно сообразить, что многократно выполняемую операцию вычитания можно заменить операцией взятия остатка от деления:

$$gcd(a, b) = gcd(a \% b, b)$$

Напишите программу, которая вычисляет НОД двух данных натуральных чисел, не превосходящих 10^{18} .

На вход программе подаются два натуральных числа: a и b .

Программа должна вывести одно число — наибольший общий делитель двух введённых чисел.

Input	Output
21 14	7

В задачах L - Q 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):

```
int x;
cin >> x;          // чтение первого числа последовательности
...               // необходимые предварительные присваивания и т.п.
while (x != 0) {
    ...           // основной алгоритм, в котором нет ни одного
                // оператора чтения
    cin >> x;     // 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.)

L. Максимум последовательности

В последовательности целых чисел определить значение наибольшего элемента последовательности.

Input	Output
1 7 9 0	9

M. Второй максимум

Последовательность состоит из различных целых 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 9 0	7

N. Количество элементов, равных максимальному

Последовательность состоит из целых чисел и завершается числом 0. Определите количество элементов последовательности, равных её максимальному элементу.

Input	Output
1 4 4 3 0	2

О. *Количество перемен знака*

Определить число перемен знака в последовательности целых чисел, заканчивающейся нулём.

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

Р. *Сумма чисел между предпоследней и последней двойками*

Последовательность состоит из натуральных чисел и завершается числом 0. 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	

Q. *Длина наибольшей площадки*

Последовательность состоит из целых чисел и завершается числом 0. Определить длину наибольшей «площадки» в последовательности (т.е. подпоследовательности подряд идущих одинаковых чисел).

Input	Output
1	2
7	
7	
9	
1	
0	

Р. *Несколько последовательностей — количество элементов*

Задано несколько последовательностей, заканчивающихся нулём. Определить количество элементов в каждой последовательности.

В первой строке вводится натуральное число N . Затем вводится N последовательностей, заканчивающихся нулём (все элементы по одному в строке). Программа должна вывести N целых чисел — количества элементов в последовательностях чисел. Напомним, что ноль не является элементом последовательности, а обозначает её конец.

Input	Output
3	2
3	1
4	3
0	
3	
0	
3	
4	
5	
0	

S. *Несколько последовательностей — максимумы*

В условиях задачи R выведите максимумы каждой последовательности. Гарантируется, что все последовательности состоят хотя бы из одного элемента.

Input	Output
3	4
3	3
4	5
0	
3	
0	
3	
4	
5	
0	

T. *Несколько последовательностей — минимакс*

В условиях задачи R выведите минимальный из максимумов всех последовательностей. Гарантируется, что все последовательности состоят хотя бы из одного элемента.

Input	Output
3	-3
-1	
4	
0	
-4	
5	
0	
-3	
-4	
-5	
0	