**Задача 1-1-Сколько раз встречается максимум**

На стандартном потоке ввода задается натуральное число N (N > 0), после которого следует последовательность из N целых чисел.

На стандартный поток вывода напечатайте, сколько раз в этой последовательности встречается максимум.

Указание: использовать массивы запрещается.

Пример №1

Данные на входе: 6 1 2 3 2 0 3

Данные на выходе: 2

Пример №2

Данные на входе: 3 5 2 -1

Данные на выходе: 1

**Задача 1-2-Шифр Цезаря**

На стандартном потоке ввода задается целое неотрицательное число N и последовательность допустимых символов в кодировке ASCII, оканчивающаяся точкой. Допустимые символы - латинские буквы 'a' ... 'z', 'A' ... 'Z' и пробел. Требуется закодировать латинские буквы ('a' ... 'z', 'A' ... 'Z') шифром Цезаря, пробелы вывести без изменения. Число N задает сдвиг в шифре. Шифр Цезаря заключается в следующем преобразовании. Пусть буквы алфавита пронумерованы от 0 до K - 1, где K - число символов в алфавите. Тогда символ с номером n кодируется символом с номером p = (n + N) mod K (mod - операция взятия остатка). На стандартном потоке вывода напечатать зашифрованное сообщение, оканчивающееся точкой. Преобразование требуется выполнять независимо для заглавных и строчных латинских букв.

Указание: использовать массивы запрещается.

Пример №1

Данные на входе: 0 Hello World.

Данные на выходе: Hello World.

Пример №2

Данные на входе: 1 Aa.

Данные на выходе: Bb.

Пример №3

Данные на входе: 3 ppc.

Данные на выходе: ssf.

**Задача 1-3-Циклический сдвиг**

На вход программе подается беззнаковое 32-битное целое число N и натуральное число K (1 ≤ K ≤ 31). Требуется циклически сдвинуть биты числа N вправо на K битов и вывести полученное таким образом число.

Пример №1

Данные на входе: 2 1

Данные на выходе: 1

Пример №2

Данные на входе: 1 2

Данные на выходе: 1073741824

Пример №3

Данные на входе: 3 1

Данные на выходе: 2147483649

Пример №4

Данные на входе: 12 2

Данные на выходе: 3

Пример №5

Данные на входе: 19 3

Данные на выходе: 1610612738

**Задача 1-4-Извлечение бит**

На вход программе подается беззнаковое 32-битное целое число N и натуральное число K (1 ≤ K ≤ 31). Требуется взять K младших битов числа N и вывести полученное таким образом число.

Пример №1

Данные на входе: 2 1

Данные на выходе: 0

Пример №2

Данные на входе: 12 3

Данные на выходе: 3

Пример №3

Данные на входе: 3 5

Данные на выходе: 2147483649

Пример №4

Данные на входе: 1024 3

Данные на выходе: 0

**Задача 1-5-Поиск последовательности бит**

На вход программе подается беззнаковое 32-битное целое число N и натуральное число K (1 ≤ K ≤ 31). Требуется взять K подряд идущих битов числа N так, чтобы полученное число было максимальным. Программа должна вывести полученное число.

Пример №1

Данные на входе: 2 1

Данные на выходе: 1

Пример №2

Данные на входе: 12 3

Данные на выходе: 6

Пример №3

Данные на входе: 3 5

Данные на выходе: 3

Пример №4

Данные на входе: 1024 3

Данные на выходе: 4

**Задача 1-6-Подсчет битов**

На вход программе подается беззнаковое 32-битное целое число N. Требуется найти количество единичных битов в двоичном представлении данного числа.

Пример №1

Данные на входе: 15

Данные на выходе: 4

Пример №2

Данные на входе: 16

Данные на выходе: 1

Пример №3

Данные на входе: 17

Данные на выходе: 2

**Задача 1-7-Инверсия старших**

На вход программе подается беззнаковое 32-битное целое число N. Требуется изменить значения всех битов старшего байта числа на противоположные и вывести полученное таким образом число.

Пример №1

Данные на входе: 1

Данные на выходе: 4278190081

Пример №2

Данные на входе: 4278190081

Данные на выходе: 1

Пример №3

Данные на входе: 2739690055

Данные на выходе: 1548507719

**Задача 1-8-Не пара**

Преподаватели Geek Brains создали набор, содержащий N (1 ≤ N ≤ 1 000 000) натуральных чисел, не превосходящих 1 000 000. Известно, что ровно одно число в этом наборе встречается один раз, а остальные — ровно по два раза. Помогите им найти это число. Входные данные: в первой входной строке вводится число N, затем по одному числу в строке вводятся N натуральных чисел, не превосходящих 2 000 000 000. Выходные данные: ваша программа должна определить число, встречающееся один раз, и вывести его на экран.

Указание: использовать массивы запрещается.

Пример №1

Данные на входе: 3 12 15 12

Данные на выходе: 15

Пример №2

Данные на входе: 5 1 2 0 2 1

Данные на выходе: 0

**Задача 1-9-Весы**

В стартер боксе Geek Brains есть электронные чашечные весы (вместимость каждой чашки 1 тонна) и гири массами 1г, 3г, 9г, 27г, и т.д. по одной штуке каждой массы. Менеджер из рекламного отдела взял такие весы и не знает, можно ли взвесить что-нибудь тяжелое и если можно, сколько гирь придется положить на весы (на любую чашку). Надо помочь ему.

Формат входных данных: единственное натуральное число – масса взвешиваемого товара в граммах (не больше 1000000000).

Формат выходных данных: единственное целое число – общее количество гирь или число –1 если данный товар взвесить невозможно.

Пример №1

Данные на входе: 8

Данные на выходе: 2

**Задача 1-10-Упаковать массив**

В программе реализована структура данных:

struct pack\_array {

uint32\_t array; // поле для хранения упакованного массива из 0 и 1

uint32\_t count0 : 8; // счетчик нулей в array

uint32\_t count1 : 8; // счетчик единиц в array

}

Необходимо реализовать только одну функцию (всю программу писать не надо!!!), которая упаковывает переданный ей массив из 32-ух элементов 0 и 1 в указанную структуру данных.

Функция должна строго соответствовать прототипу: void array2struct(int [], struct pack\_array \*)

Загрузите только одну функцию или просто вставьте ее текст void array2struct(int [], struct pack\_array \*)

Пример №1

Данные на входе: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Данные на выходе:

Поля структуры

array = ffff0000

count0 = 16

count1 = 16

Пример №2

Данные на входе: 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1

Данные на выходе:

Поля струткуры

array = ffa70009

count0 = 17

count1 = 15

**Задача 1-11-Распаковать порядок**

Необходимо реализовать только одну функцию, которой на входе передаётся вещественное число в типе float, она возвращает порядок который хранится в поле EXP в виде десятичного целого числа.

Функция должна строго соответствовать прототипу int extractExp(float)

Вставьте только текст функции int extractExp(float)

Пример №1

Данные на входе: 1.0

Данные на выходе: 127

Пример №2

Данные на входе: 2.0

Данные на выходе: 128

Пример №3

Данные на входе: 3.99999

Данные на выходе: 128

Пример №4

Данные на входе: 0.001

Данные на выходе: 117