**Общее задание:**

Реализовать заданный в варианте алгоритм на языке ассемблера NASM (<https://www.nasm.us/pub/nasm/releasebuilds/2.15.05/win64/>). Программу собирать для архитектуры x86-64 (см. настройки SASM).

Результат работы программы вывести на экран соответствующим макросом из стандартного комплекта SASM (<http://dman95.github.io/SASM/> , список макросов – внизу страницы).

Массивы реализовывать, как статические переменные в секции .data (или .bss).

При работе со строками максимальную длину строки установить не менее 255.

При работе со строками считать, что допустимый алфавит состоит из цифр, английских букв обоих регистров и символов []()+-\*/=&?!/\”’:;@., (ASCII-символы, при желании можно добавить кириллицу)

При работе с числовыми массивами максимальную длину массива установить не менее 100

Если в задании предполагается ввод массива пользователем, то пользователь сначала вводит размер массива, а затем сам массив.

Для массивов чисел, если не указано иное, тип элемента - int.

***Проверку ввода пользователя можно не делать.***

**Задания на лабораторную:**

**Легкий уровень:**

1. Для введенной пользователем строки вывести инвертированную строку и строку, состоящую только из четных элементов исходной строки.
2. Для введенной пользователем строки и числа N вывести строку, циклически сдвинутую на N вправо
3. Для введенной пользователем строки проверить, является ли она палиндромом
4. Для введенной пользователем строки проверить, сколько раз в ней встречается буква, с которой начинается ваша фамилия (регистр букв игнорировать)
5. Для введенного пользователем числа вывести представление числа по основанию 8.
6. Для введенного пользователем массива рассчитать минимум и максимум.
7. Проверить введенное пользователем число на простоту, вывести ответ.
8. Для введенного пользователем вектора вывести квадрат его нормы.
9. Для введенного пользователем массива чисел подсчитать количество *битов* со значением 1.
10. Для введенного пользователем массива рассчитать среднее.

**Средний уровень**

1. Зашифровать введенную пользователем строку шифром Цезаря.
2. Реализовать аналог функции strcmp. Продемонстрировать результат работы алгоритма на 2-х введенных строках.
3. Для введенных пользователем строк проверить, содержит ли первая строка вторую.
4. Для пользовательской строки вывести частоту встречаемости каждого символа. (помните, что символы - это целые числа в диапазоне 0-255).
5. Для пользовательского массива чисел рассчитать и вывести массив с результатом расчета скользящего среднего в окне размером 5.
6. Посчитать количество уникальных элементов в массиве чисел типа short.

**Сложный уровень:**

1. Отсортировать введенный пользователем массив пузырьком
2. Отсортировать введенный пользователем массив вставками
3. Отсортировать введенный пользователем массив выбором
4. Реализовать [линейный конгруэнтный генератор](https://ru.wikipedia.org/wiki/%D0%9B%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%BD%D0%B3%D1%80%D1%83%D1%8D%D0%BD%D1%82%D0%BD%D1%8B%D0%B9_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4) псевдослучайных чисел. Параметры генератора считать постоянными. Используя введенное пользователем число в качестве начальной точки, вывести 100 сгенерированных чисел.
5. Реализовать генератор псевдослучайных чисел [xorshift128](https://ru.wikipedia.org/wiki/Xorshift). Используя введенное пользователем число в качестве начальной точки, вывести 100 сгенерированных чисел.
6. Реализовать генератор псевдослучайных чисел [BlumBlumShub](https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC_%D0%91%D0%BB%D1%8E%D0%BC_%E2%80%94_%D0%91%D0%BB%D1%8E%D0%BC%D0%B0_%E2%80%94_%D0%A8%D1%83%D0%B1%D0%B0). Параметры p и q выбрать самостоятельно. В качестве результата использовать 1 или 2 младших байта полученного числа. Используя введенное пользователем число в качестве начальной точки, вывести 100 сгенерированных чисел.
7. Реализовать вычисление контрольной суммы [CRC32](https://ru.wikibooks.org/wiki/%D0%A0%D0%B5%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8_%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%BE%D0%B2/%D0%A6%D0%B8%D0%BA%D0%BB%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D0%B8%D0%B7%D0%B1%D1%8B%D1%82%D0%BE%D1%87%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4) для введенной строки. Вывести полученное значение.
8. Отсортировать введенный пользователем массив шейкерной сортировкой