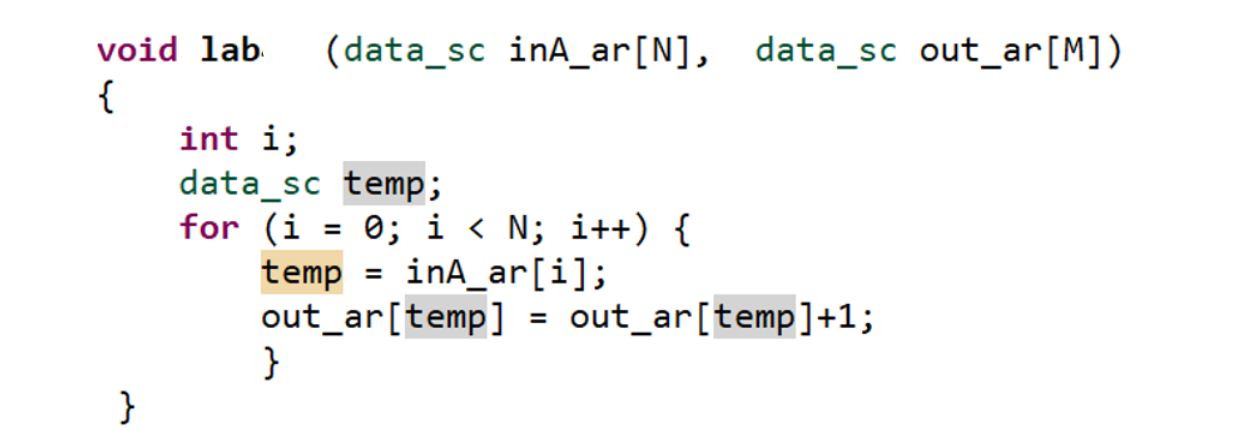
* Создать на языке Си функцию,
  + Последовательно считывающую элементы массива, заполненного случайными целыми числами (тип int), имеющими значения от **0 до M-1**. Размер массива - **N** элементов.
  + Функция подсчитывает количество появлений чисел от 0 до **M-1** во входном массиве и записывает полученные значения в соответствующие элементы (с 0 до **M-1)**  выходного массива, имеющего **M** элементов типа int.

*Т.е. по входному массиву размером N, заполненного случайными целыми числами от* **0 до M-1** *строится гистограмма в выходном массиве размером M элементов.*

* + В файле lab3\_z1.h должны быть определены **M**  и **N** и тип данных data\_sc, имеющий тип int.

Например, можно примерно так (возможны любые другие варианты):



* Создать на языке Си тест для проверки работы функции. Тест должен обеспечивать
  + Формирование исходного входного массива, заполненного случайными числами.
    - N=8192, M=8192
  + Запуск функции 2 раза
    - Перед следующим запуском функции не забудьте обнулить формируемый функцией выходной массив.
  + Проверку правильности полученного функцией результата (сравнение полученного функцией массива и ожидаемого массива, вычисленного в тесте с использованием другого, не такого как в функции, алгоритма построения гистограммы ) и формирование признака успешного/неуспешного выполнения для каждого запуска функции.
* Отладить функцию и тест (при неправильном результате в любом из запусков функции должен сообщать об ошибке).
* Создать скрипт автоматизирующий процесс:
  + Создания проекта lab3\_z1,
  + Назначения функции lab3\_z1
  + Подключения файла lab3\_z1.c (папка source ),
  + Подключения файла lab3\_z1\_test.c (папка source),
  + Создания решении
    - sol1, для которого
      * Port Level I/O interface по умолчанию
      * Микросхема: xa7a12tcsg325-1Q
      * Период тактового сигнала: 6нс, uncertainty 0.1нс.
      * Си моделирование
      * Синтез
      * С/RTL cosimulation (с опцией полной трассировки)
    - sol2, для которого
      * Port Level I/O interface для входного массива **ap\_fifo**
      * Микросхема: xa7a12tcsg325-1Q
      * Период тактового сигнала: 6нс, uncertainty 0.1нс.
      * Синтез
      * С/RTL cosimulation (с опцией полной трассировки)
* Отладить и проверить работу созданного скрипта.
* После выполнения скрипта открыть GUI
* Убедиться, что созданы все решения
* Используя средства HLS сравнить полученные решения
  + использованные интерфейсы
  + аппаратные затраты
  + результаты планирования (Schedule viewer и Performance profile)
* В GUI Vivado HLS cоздайте решение sol3, в котором установите Port Level I/O interface для выходного массива **ap\_fifo**
* Осуществите синтез и объясните полученный результат.

Исследование времени выполнения на ПК

* **Используются исходные коды и результаты исследования проведенного ранее.**
* На базе использованного выше Си теста создать отдельный, модернизированный, тест для проверки времени выполнения синтезируемой функции на ПК:
  + добавить в тест операторы измерения **времени выполнения** синтезируемой функции (например, как-то так: <https://solarianprogrammer.com/2019/04/17/c17-programming-measuring-execution-time-delaying-program/> ).
  + Увеличить количество запусков синтезируемой функции до 32. Для каждого запуска измерить время, найти среднее значение и вывести как результат.
  + Точность измерения времени (наносекунды).
  + Провести исследование времени выполнения синтезируемой функции на Вашем ПК
    - Осуществить компиляцию модернизированного теста и запустить его как отдельное приложение
    - В отчете привести:
      * Параметры Вашего ПК: тип процессора, частота работы процессора, объем ОЗУ
      * результаты измерения времени выполнения
* Оформить отчет, который должен включать
  + Задание
  + Раздел с описанием исходного кода функции
  + Раздел с описанием теста
  + Раздел с описание созданного командного файла
  + Раздел с описанием результатов сравнения решений (со снимками экрана из Vivado HLS)
  + Раздел с анализом результатов (со снимками экрана с заполненной таблицей и полученным графиком)
    - Анализ и выбор оптимального (критерий максимальная производительность) решения
  + Результаты исследования времени выполнения на ПК и сравнение с аппаратными решениями.
  + Выводы

Архив должен включать всю рабочую папку проекта (включая модернизированный тест и скомпилированное приложение), отчет