Федеральное государственное бюджетное образовательное учреждение высшего образования "Национальный исследовательский университет "МЭИ"

ОТЧЕТ

По лабораторной работе № 3

# «Декомпозиция и контроль корректности программ»

ФИО студента Зобов Павел Алесандрович

Номер группы А-02-18

Имя преподавателя Мохов Андрей Сергеевич

Козлюк Дмитрий Александрович

Москва, «НИУ «МЭИ», 2019

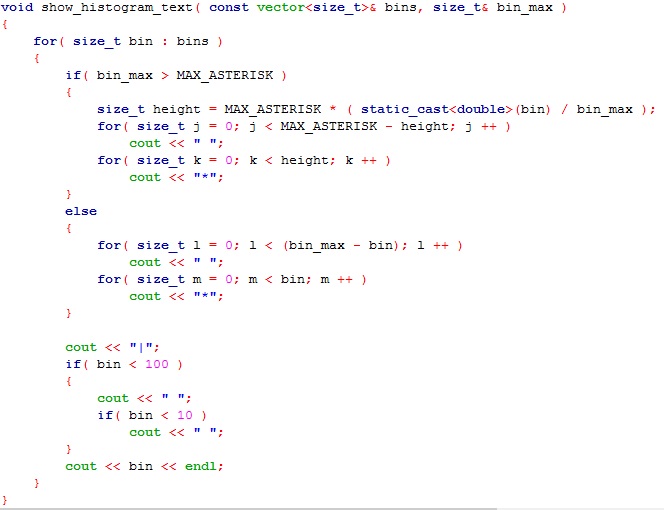
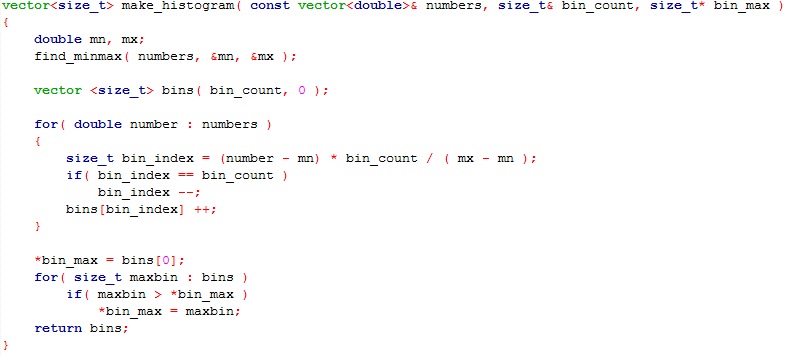
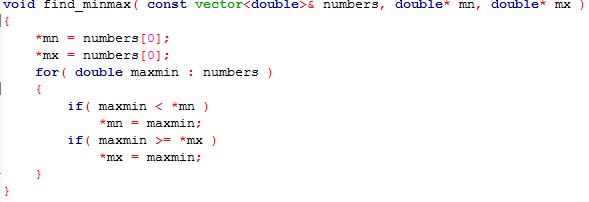
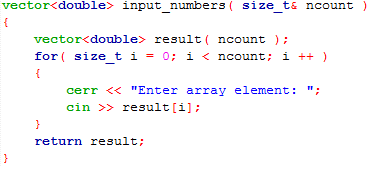
Ход работы

Цели работы:

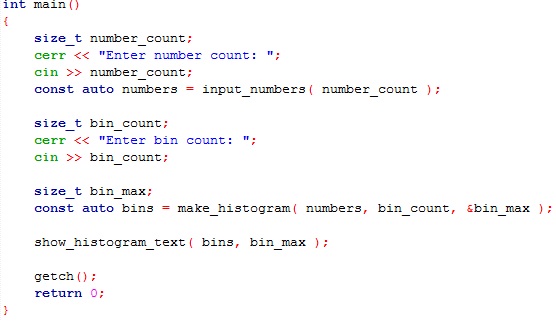
1. Уметь структурировать программу при помощи функций и структур.
2. Уметь писать модульные тесты.

# Часть 1. Декомпозиция программы функциями

# Оформим ввод массива чисел, функцию поиска минимума и максимума, и функции расчета и вывода гистограммы в отдельные функции:

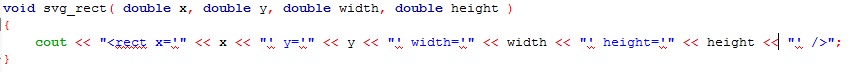
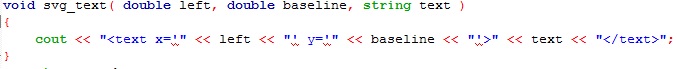


При этом многократно используем передачу по ссылке и по указателю, в том числе и константную ссылку на массив **numbers**, так как после ввода он больше не изменяется.

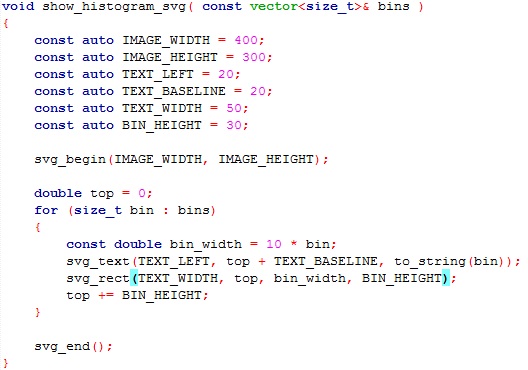


# Часть 2. Вывод гистограммы как изображения в формате SVG

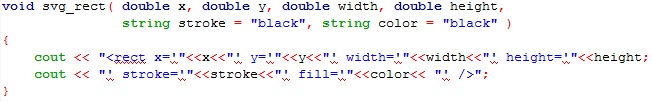
# Реализуем функции вывода текста (подписей к столбцам гистограммы) и прямоугольников (столбцов гистограммы):



Используем обе функции, для того, чтобы построить гистограмму. Откажемся от использования «магических констант» и зарегистрируем длину и ширину изображения, ширину и высоту текстового поля и столбцов и т.д.:

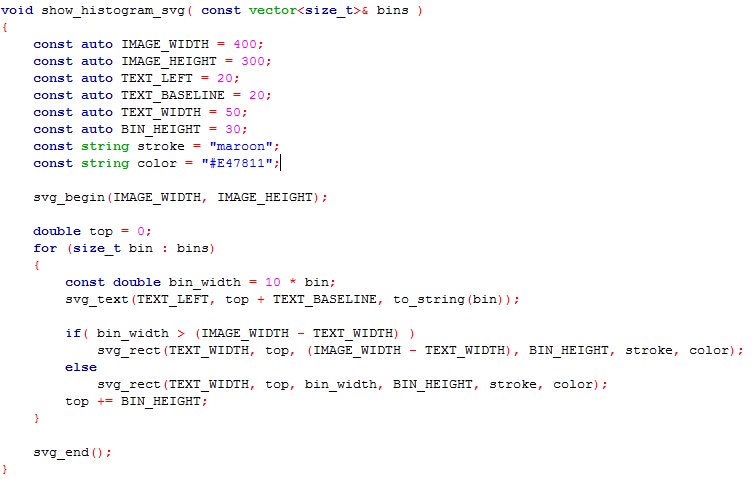


Добавим цвета прямоугольника по умолчанию – черный для обводки и заливки, а в вызове функции отрисовки прямоугольника – стандартный бордовый для обводки и оранжевый из RGB палитры:





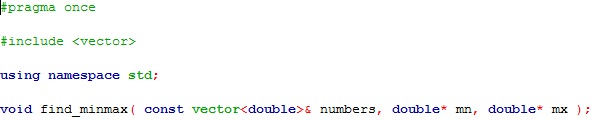
Реализуем масштабирование длины столбцов. Для этого добавим проверку на то, превышает ли рассчитанная длина столбца максимальную длину столбца (ширина изображения минус ширина текстового поля). В случае превышения максимальной длины будет отрисовываться столбец с длиной, равной максимальной. Для удобства были также зарегистрированы строковые константы цвета обводки и заливки::



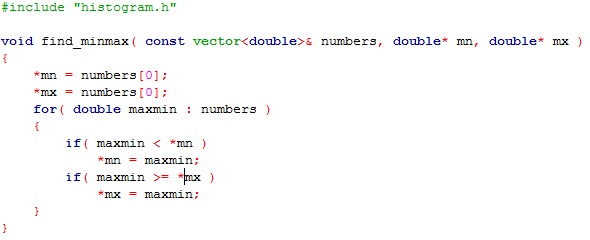
# Часть 3. Модульное тестирование

# Вынесем функцию find\_minmax() в отдельный модуль. Для этого создадим заголовочный файл с прототипом этой функции, и файл реализации с телом функции. В качестве «стража включения» в заголовочном файле используем директиву #pragma once:

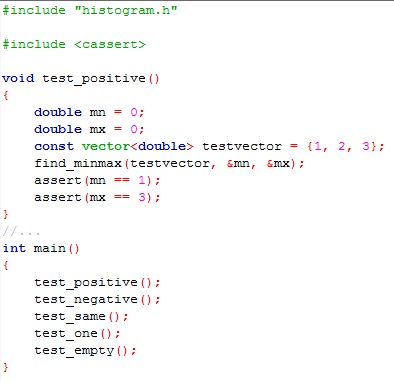
# Заголовочный файл



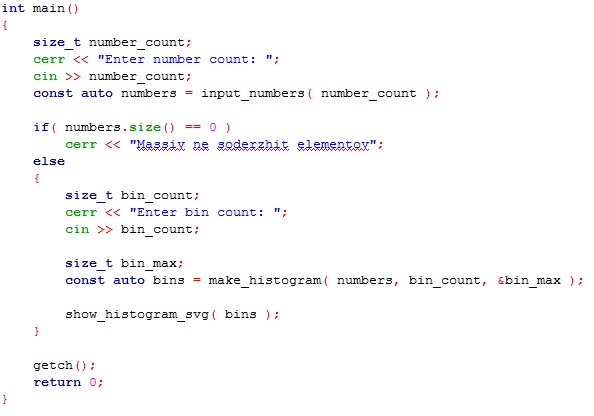
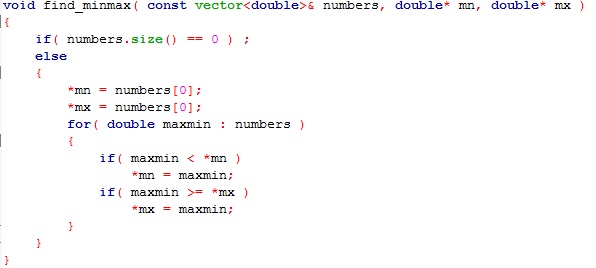
Файл реализации



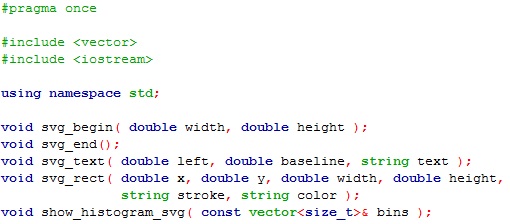
Создадим новый проект. Напишем программу для тестирования функции **find\_minmax().** Протестируем в ней массивы из положительных, отрицательных, одинаковых элементов, массив из одного элемента, пустой массив:



При тестировании пустого массива мы убеждаемся, что этот случай не учтен в программе и код работает некорректно. Для этого добавим в модуль и в главную программу проверку на нулевую длину массива:



Внесем также в отдельный модуль все функции для работы с SVG. Так будет выглядеть заголовочный файл:



В файл реализации перенесем все SVG-функции из **main.cpp** и подключим вышеописанный заголовочный файл **svg.h,** который также необходимо подключить в **main.cpp**.

Ссылка на репазитарий: https://github.com/ZobovPA/laab03