Разработка Программного Обеспечения Систем Управления

Отчет по лабораторной работе №3

Выполнил студент группы А-03-19:

Мосалов К.Д.

27.04.2020

**Цель работы**

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

Индивидуальное задание (1 вариант)

Добавил функцию позволяющую пользователю задавать ширину гистограммы из определенного диапазона, я сделал через if может быть лучше было бы сделать через switch. Введенную ширину использовал как максимальную ширину столбика для максимального кол-ва оценок, и относительно нее масштабировал другие, путем нахождения длины на одну оценку. Чтобы гистограмма всегда умещалась прибавляю к заданной ширине длину текста (50 пикселей). Добавил тест для функции hist\_scale на правильность нахождения коэффициента масштабирования, больше проверять в функции нечего(наверное), поэтому добавил тест функции поиска максимального кол-ва оценок одного вида (find\_maxinbins).

Пример результата работы программы

Входные данные:

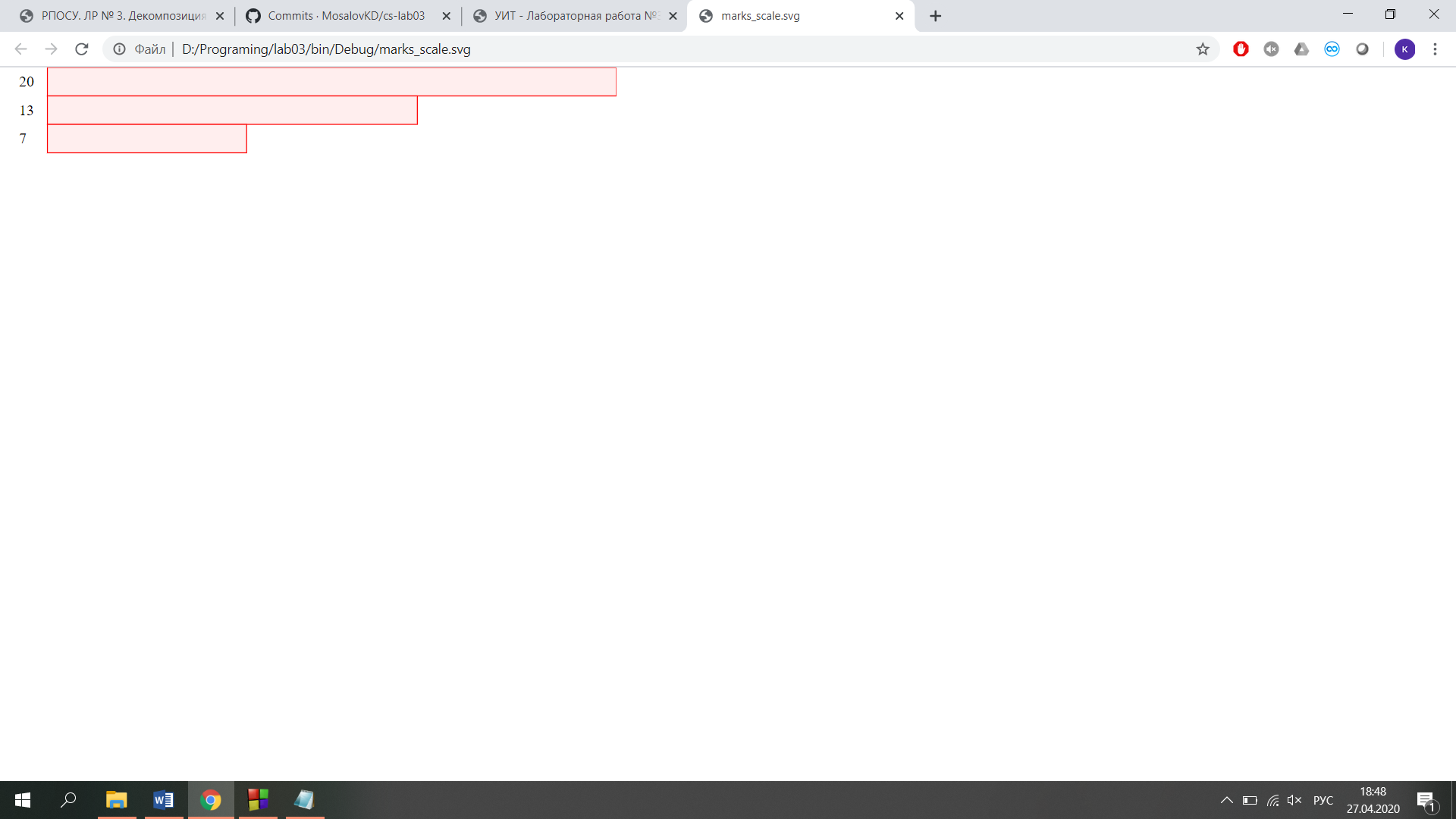
40

3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5

3

600

Результат:



**Код main.cpp**

#include <iostream>

#include <vector>

#include "histogram.h"

#include "svg.h"

using namespace std;

vector<double> input\_numbers(size\_t count)

{

vector<double> result(count);

for (size\_t i = 0; i < count; i++) {

cin >> result[i];

}

return result;

}

/\*

void show\_histogram\_text (const vector<size\_t>& bins)

{

const size\_t max\_width = 80;

const size\_t max\_ast = max\_width - 4;

double koeff = 1;

size\_t max\_count = 0;

for(size\_t bin:bins)

{

if(bin > max\_count)

max\_count = bin;

}

if (max\_count > max\_ast)

{

koeff = static\_cast <double> (max\_ast)/max\_count;

}

for(size\_t bin:bins)

{

if (bin < 10)

cout << " " << " ";

if (bin >= 10 && bin < 100)

cout << " ";

if (bin > 100)

{

cout << "error";

break;

}

cout << bin << "|";

size\_t height = bin \* koeff;

for (int i =0; i < height; i++)

cout << "\*";

cout << endl;

}

}

\*/

int main()

{

cerr << "Vvedite kol-vo chisel: ";

size\_t number\_count;

cin >> number\_count;

const auto numbers = input\_numbers(number\_count);

cerr << "Vvedite kol-vo korzin: ";

size\_t bin\_count;

cin >> bin\_count;

const auto bins = make\_histogram(numbers, bin\_count);

if (bins.size() == 0)

{

cout << "error";

return 0;

}

//show\_histogram\_text(bins);

show\_histogram\_svg(bins);

return 0;

}

**Код svg.cpp**

void hist\_scale (int& HIST\_WIDTH, double& BLOCK\_WIDTH, const vector<size\_t>& bins) // domashnee zadanie

{

size\_t kol\_vo\_chisel = 0;

for (size\_t bin : bins)

{

kol\_vo\_chisel = kol\_vo\_chisel + bin;

}

cerr << "Vvedite shirinu histogrammi:";

cin >> HIST\_WIDTH;

if (HIST\_WIDTH < 70)

{

cerr << endl << "error: shirina ochen' malen'kaya. Vvedite znachenie ot 70 do 800.";

cin >> HIST\_WIDTH;

}

if (HIST\_WIDTH > 800)

{

cerr << endl << "error: shirina slishkom bol'shaya. Vvedite znachenie ot 70 do 800.";

cin >> HIST\_WIDTH;

}

size\_t binmax = find\_maxinbins(bins);

BLOCK\_WIDTH = HIST\_WIDTH/binmax;

if (HIST\_WIDTH < 1/3.0 \* kol\_vo\_chisel \* BLOCK\_WIDTH)

{

cerr << endl << "error: shirina ochen' malen'kaya. Vvedite znachenie ot 70 do 800.";

cin >> HIST\_WIDTH;

BLOCK\_WIDTH = HIST\_WIDTH/binmax;

}

}

void svg\_end()

{

cout << "</svg>\n";

}

void show\_histogram\_svg(const vector<size\_t>& bins)

{

int HIST\_WIDTH = 0;

double BLOCK\_WIDTH; //(koeff)

size\_t binmax = find\_maxinbins(bins);

hist\_scale(HIST\_WIDTH, BLOCK\_WIDTH, bins);

const auto IMAGE\_WIDTH = HIST\_WIDTH + 50;

const auto IMAGE\_HEIGHT = 300;

const auto TEXT\_LEFT = 20;

const auto TEXT\_BASELINE = 20;

const auto TEXT\_WIDTH = 50;

const auto BIN\_HEIGHT = 30;

svg\_begin(IMAGE\_WIDTH, IMAGE\_HEIGHT);

double top = 0;

for (size\_t bin : bins)

{

double bin\_width = BLOCK\_WIDTH \* bin;

if (bin == binmax)

{

bin\_width = HIST\_WIDTH;

}

svg\_text(TEXT\_LEFT, top + TEXT\_BASELINE, to\_string(bin));

svg\_rect(TEXT\_WIDTH, top, bin\_width, BIN\_HEIGHT, "red", "#ffeeee");

top += BIN\_HEIGHT;

}

svg\_end();

}

**Код histogram.cpp**

#include "histogram.h"

#include <iostream>

void find\_minmax(const vector<double>& numbers, double& min, double& max)

{

if (numbers.size() == 0)

{

return;

}

min = numbers[0], max = numbers[0];

for (double number:numbers)

{

if (number < min)

min = number;

else

{

if (number > max)

max = number;

}

}

}

vector<size\_t> make\_histogram (const vector<double>& numbers, size\_t bin\_count)

{

double min = 0, max = 0;

find\_minmax(numbers, min, max);

vector<size\_t> foobins (bin\_count);

for (double number:numbers)

{

size\_t bin\_index = (size\_t)((number - min)\*bin\_count/(max - min));

if (number == max)

{

bin\_index--;

}

foobins[bin\_index]++;

}

return foobins;

}

size\_t find\_maxinbins(const vector<size\_t>& bins)

{

size\_t max = bins[0];

for (size\_t bin:bins)

{

if(bin > max)

max = bin;

}

return max;

}

**Код test.cpp**

#include "svg.h"

#include "histogram.h"

#include <cassert>

void test\_positive()

{

double min = 0;

double max = 0;

find\_minmax({1, 2, 3}, min, max);

assert(min == 1);

assert(max == 3);

}

void test\_otric()

{

double min = 0;

double max = 0;

find\_minmax({-10, -12, -3}, min, max);

assert(min == -12);

assert(max == -3);

}

void test\_sim()

{

double min = 0;

double max = 0;

find\_minmax({1, 1, 1}, min, max);

assert(min == 1);

assert(max == 1);

}

void test\_izodnogo()

{

double min = 0;

double max = 0;

find\_minmax({1}, min, max);

assert(min == 1);

assert(max == 1);

}

void test\_pustogo()

{

double min = 0;

double max = 0;

find\_minmax({}, min, max);

assert(min == 0);

assert(max == 0);

}

void test\_shirini()

{

int HIST\_WIDTH = 0;

double BLOCK\_WIDTH = 0;

hist\_scale (HIST\_WIDTH, BLOCK\_WIDTH, {10, 6, 8});

assert(BLOCK\_WIDTH == HIST\_WIDTH/10);

}

void test\_poiska\_max\_kol\_va\_ocenok()

{

size\_t max = find\_maxinbins({10, 4, 2});

assert(max == 10);

}

int main()

{

test\_positive();

test\_otric();

test\_sim();

test\_izodnogo();

test\_pustogo();

test\_shirini();

test\_poiska\_max\_kol\_va\_ocenok();

return 0;

}

**Код histogram.h**

#pragma once

#include <vector>

using namespace std;

void find\_minmax(const vector<double>& numbers, double& min, double& max);

vector<size\_t> make\_histogram (const vector<double>& numbers, size\_t bin\_count);

size\_t find\_maxinbins(const vector<size\_t>& bins);

**Код svg.h**

#pragma once

#include "histogram.h"

#include <string>

void svg\_begin(double width, double height);

void svg\_text(double left, double baseline, string text);

void svg\_rect(double x, double y, double width, double height);

void hist\_scale (int& HIST\_WIDTH, double& BLOCK\_WIDTH, const vector<size\_t>& bins); // domashnee zadanie

void svg\_end();

void show\_histogram\_svg(const vector<size\_t>& bins);

Ссылка на репозитарий

<https://github.com/MosalovKD/cs-lab03>