Федеральное государственное бюджетное образовательное учреждение высшего образования «Национальный исследовательский университет «МЭИ»

Институт информационных и вычислительных технологий

Кафедра Управления и интеллектуальных технологий

**Отчёт по лабораторной работе № 3**

**По курсу «Разработка программного обеспечения систем управления»**

**Декомпозиция и контроль корректности программ**

Выполнил студент группы А-02-20

Минаев Дмитрий Алексеевич

Проверил

Мохов А.С.

Козлюк Д.А.

Василькова П.Д.

Москва 2021

**Цель работы**

1. Уметь структурировать программу при помощи функций.
2. Уметь писать модульные тесты.
3. **Выделил все команды в функции.**

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

{

vector<double> result(count);

cerr << "Enter numbers: ";

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

{

cin >> result[i];

}

return result;

}

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

{

double min, max;

find\_minmax(numbers, min, max);

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

for (double number : numbers)

{

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

if (bin == bin\_count)

{

bin--;

}

bins[bin]++;

}

return bins;

}

int show\_histogram\_text(const vector<double> &numbers, size\_t bin\_count, vector<size\_t> &bins)

{

const size\_t SCREEN\_WIDTH = 80;

const size\_t MAX\_ASTERISK = SCREEN\_WIDTH - 4 - 1;

size\_t max\_count = 0;

for (size\_t count : bins)

{

if (count > max\_count)

{

max\_count = count;

}

}

const bool scaling\_needed = max\_count > MAX\_ASTERISK;

for (size\_t bin : bins)

{

if (bin < 100)

{

cout << ' ';

}

if (bin < 10)

{

cout << ' ';

}

cout << bin << "|";

size\_t height = bin;

if (scaling\_needed)

{

const double scaling\_factor = (double)MAX\_ASTERISK / max\_count;

height = (size\_t)(bin \* scaling\_factor);

}

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

{

cout << '\*';

}

cout << '\n';

}

return max\_count;

}

int shkala(const auto &max\_name, int &int\_shkal, int &j)

{

int kof\_shkal = max\_name/int\_shkal + 1;

for (j=0; j<kof\_shkal; j++)

{

cout<<"|";

for (int z=0; z<int\_shkal-1; z++)

{

cout<<"-";

}

}

cout<<"|"<<endl;

cout<<" ";

int chislo\_shkal=-int\_shkal;

for (j=0; j<kof\_shkal\*int\_shkal; j++)

{

chislo\_shkal = chislo\_shkal + int\_shkal;

if (chislo\_shkal <= int\_shkal || chislo\_shkal == kof\_shkal\*int\_shkal) //Проверка для вывода 1ого, 2ого и последнего числа под шкалой

{

cout<<chislo\_shkal;

}

if (chislo\_shkal < 10)

{

for (int z=0; z<int\_shkal-1; z++)

{

cout<<" ";

}

}

if ((chislo\_shkal>=10)&&(chislo\_shkal<100))

{

for (int z=0; z<int\_shkal; z++)

{

cout<<" ";

}

}

if (chislo\_shkal>=100)

{

for (int z=0; z<int\_shkal+1; z++)

{

cout<<" ";

}

}

}

}

1. **Создал модульный тест (для функции find\_minmax()).**

histogram.h:

#ifndef HISTOGRAM\_H\_INCLUDED

#define HISTOGRAM\_H\_INCLUDED

#include <vector>

using namespace std;

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

#endif // HISTOGRAM\_H\_INCLUDED

**histogram.cpp:**

#include "histogram.h"

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

{

min = numbers[0];

max = numbers[0];

for (double number : numbers)

{

if (number < min)

{

min = number;

}

if (number > max)

{

max = number;

}

}

}

**test.cpp:**

#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\_negativ() {

double min = 0;

double max = 0;

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

assert(min == -3);

assert(max == -1);

}

void

test\_odinakov() {

double min = 0;

double max = 0;

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

assert(min == 1);

assert(max == 1);

}

void

test\_one() {

double min = 0;

double max = 0;

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

assert(min == 1);

assert(max == 1);

}

int main()

{

test\_positive();

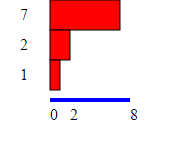
test\_negativ();

test\_odinakov();

test\_one();

}

1. **Пример SVG.**



**Оформление данного SVG:**

<?xml version='1.0' encoding='UTF-8'?>

<svg width='400' height='300' viewBox='0 0 400 300' xmlns='http://www.w3.org/2000/svg'>

<text x='20' y='20'>7</text><rect x='50' y='0' width='70' height='30' stroke='black' fill='red' /><text x='20' y='50'>2</text><rect x='50' y='30' width='20' height='30' stroke='black' fill='red' /><text x='20' y='80'>1</text><rect x='50' y='60' width='10' height='30' stroke='black' fill='red' />

<line x1='50' y1='100' x2='130' y2='100' stroke='blue' stroke-width='4'/>

<text x='50' y='120'>0</text><text x='70' y='120'>2</text><text x='130' y='120'>8</text>

</svg>