## Министерство транспорта Российской Федерации Федеральное агентство железнодорожного транспорта

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Дальневосточный государственный университет путей сообщения» Естественно-научный институт

Кафедра «Вычислительная техника и компьютерная графика»

Лабораторная работа №02

ЛР 01.03.02 ЯиМП.11.02.00–921

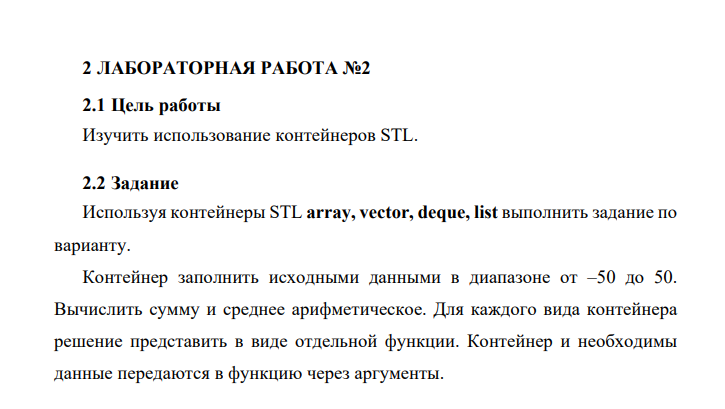
## Студент 921 гр. С.Н. Костенко

подпись, дата

## Ст. Преподаватель В.В. Халиман

подпись, дата

Общее задание

****

Дополнительное задание



Код решения

#include <iostream>

#include <array>

#include <vector>

#include <deque>

#include <list>

#include <numeric>

#include <cstdlib>

#include <ctime>

#include <algorithm>

using namespace std;

// Функция для вычисления суммы элементов

template <typename Container>

int calculateSum(const Container& container) {

return accumulate(container.begin(), container.end(), 0);

}

// Функция для вычисления среднего арифметического

template <typename Container>

double calculateAverage(const Container& container) {

if (container.empty())

return 0.0;

int sum = calculateSum(container);

return static\_cast<double>(sum) / container.size();

}

// Функция для деления каждого элемента на максимальный элемент

template <typename Container>

void divideMax(Container& container) {

if (container.empty())

return;

auto maxElem = \*max\_element(container.begin(), container.end());

for (auto& elem : container) {

elem /= static\_cast<double>(maxElem);

}

}

// Функция для вывода элементов

template <typename Container>

void printContainer(const Container& container) {

for (const auto& elem : container) {

cout << elem << " ";

}

cout << endl;

}

int main() {

setlocale(LC\_ALL, "ru");

srand(time(nullptr)); // генератор случайных чисел

array<double, 20> arr;

vector<double> vec;

deque<double> deq;

list<double> lst;

// Заполнение контейнеров случайными числами

cout << endl << "Array | Vector | Deque | List" << endl;

for (int i = 0; i < 20; ++i) {

int randomNum = rand() % 101 - 50; // Генерируем случайное число в диапазоне от -50 до 50

int randomNum2 = rand() % 101 - 50; // Генерируем случайное число в диапазоне от -50 до 50

int randomNum3 = rand() % 101 - 50; // Генерируем случайное число в диапазоне от -50 до 50

int randomNum4 = rand() % 101 - 50; // Генерируем случайное число в диапазоне от -50 до 50

arr[i] = randomNum;

vec.push\_back(randomNum2);

deq.push\_back(randomNum3);

lst.push\_back(randomNum4);

cout << randomNum << " | " << randomNum2 << " | " << randomNum3 << " | " << randomNum4 << endl;

}

int sumArr = calculateSum(arr);

double avgArr = calculateAverage(arr);

int sumVec = calculateSum(vec);

double avgVec = calculateAverage(vec);

int sumDeq = calculateSum(deq);

double avgDeq = calculateAverage(deq);

int sumLst = calculateSum(lst);

double avgLst = calculateAverage(lst);

divideMax(arr);

divideMax(vec);

divideMax(deq);

divideMax(lst);

cout << "Сумма и среднее арифметическое для array:\n";

cout << "Сумма: " << sumArr << "\n";

cout << "Среднее арифметическое: " << avgArr << "\n\n";

cout << "Результат после деления элемента на макисмальный для array: ";

printContainer(arr);

cout << "Сумма и среднее арифметическое для vector:\n";

cout << "Сумма: " << sumVec << "\n";

cout << "Среднее арифметическое: " << avgVec << "\n\n";

cout << "Результат после деления элемента на макисмальный для Vector: ";

printContainer(vec);

cout << "Сумма и среднее арифметическое для deque:\n";

cout << "Сумма: " << sumDeq << "\n";

cout << "Среднее арифметическое: " << avgDeq << "\n\n";

cout << "Результат после деления элемента на макисмальный для Deque: ";

printContainer(deq);

cout << "Сумма и среднее арифметическое для list:\n";

cout << "Сумма: " << sumLst << "\n";

cout << "Среднее арифметическое: " << avgLst << "\n";

cout << "Результат после деления элемента на макисмальный для List: ";

printContainer(lst);

return 0;

}

Результат

