Министерство образования Республики Беларусь

Управление образования

«Белорусский государственный университет информатики и радиоэлектроники»

Отчёт

По лабораторной работе №7

«**Шаблоны. Соединение классов в Qt**»

|  |  |
| --- | --- |
|  | Выполнил:  студент группы 262101  Калинков Даниил Русланович  Проверил:  Ловецкий Михаил Юрьевич |

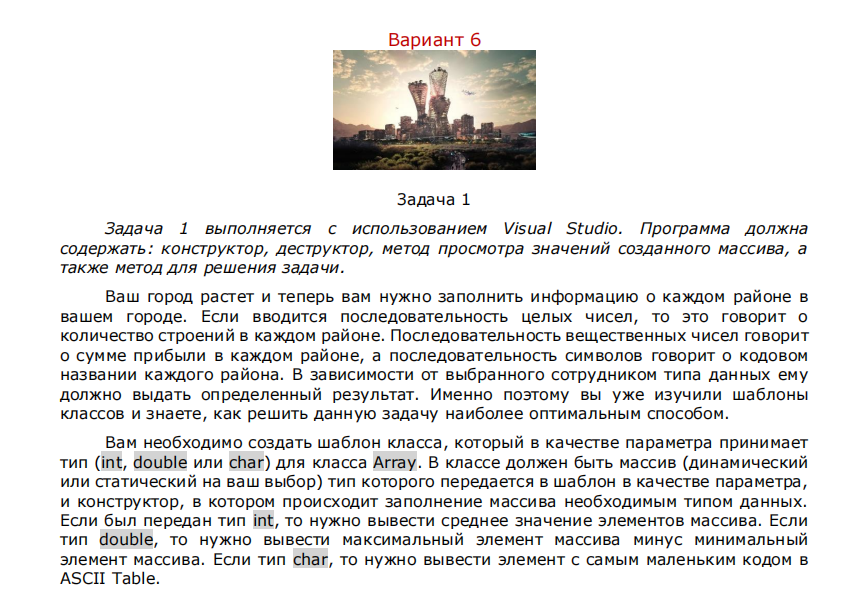
Минск, 2023

*Цель работы*: Изучить методику создания шаблонов функций, шаблонов

классов. Научиться соединять классы в Qt.

Вариант 6

Задание 1:



Реализация:

#include <iostream>

#include <algorithm>

#include <stdexcept>

template <typename T>

class Array {

private:

T\* data;

int size;

public:

Array(int arraySize) : size(arraySize) {

data = new T[size];

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

// Заполняем массив нулями или другими значениями по умолчанию в зависимости от типа T

data[i] = T();

}

}

// Деструктор для освобождения выделенной памяти

~Array() {

delete[] data;

}

// Оператор [] для доступа к элементам массива

T& operator[](int index) {

if (index >= 0 && index < size) {

return data[index];

}

else {

throw std::out\_of\_range("Index out of range");

}

}

const T& operator[](int index) const {

if (index >= 0 && index < size) {

return data[index];

}

else {

throw std::out\_of\_range("Index out of range");

}

}

// Операции в зависимости от типа T

void process() const {

if constexpr (std::is\_same\_v<T, int>) {

if (size > 0) {

int sum = 0;

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

sum += data[i];

}

double average = static\_cast<double>(sum) / size;

std::cout << "Average: " << average << std::endl;

}

else {

std::cout << "Array is empty!" << std::endl;

}

}

else if constexpr (std::is\_same\_v<T, double>) {

if (size > 0) {

double minElem = \*std::min\_element(data, data + size);

double maxElem = \*std::max\_element(data, data + size);

double diff = maxElem - minElem;

std::cout << "Max - Min: " << diff << std::endl;

}

else {

std::cout << "Array is empty!" << std::endl;

}

}

else if constexpr (std::is\_same\_v<T, char>) {

if (size > 0) {

char minChar = \*std::min\_element(data, data + size);

std::cout << "Min Char (ASCII Code): " << static\_cast<int>(minChar) << std::endl;

}

else {

std::cout << "Array is empty!" << std::endl;

}

}

else {

std::cout << "Operation not available for this data type." << std::endl;

}

}

};

int main() {

// Создаем объекты массивов с разными типами

Array<int> intArray(5);

Array<double> doubleArray(4);

Array<char> charArray(3);

// Заполняем массивы какими-то значениями

intArray[0] = 10;

intArray[1] = 20;

intArray[2] = 30;

intArray[3] = 40;

intArray[4] = 50;

doubleArray[0] = 1.1;

doubleArray[1] = 2.2;

doubleArray[2] = 3.3;

doubleArray[3] = 4.4;

charArray[0] = 'a';

charArray[1] = 'b';

charArray[2] = 'c';

// Выполняем операции для каждого массива

intArray.process();

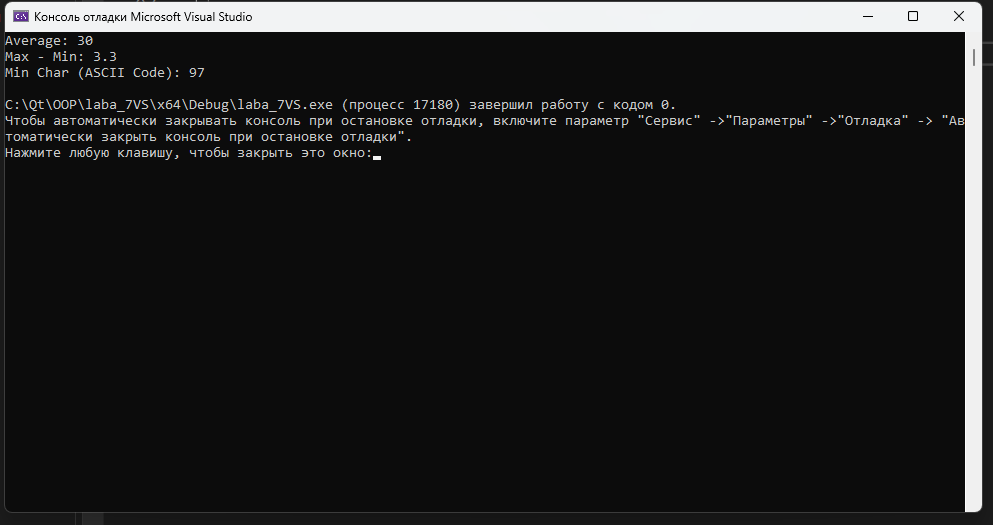
doubleArray.process();

charArray.process();

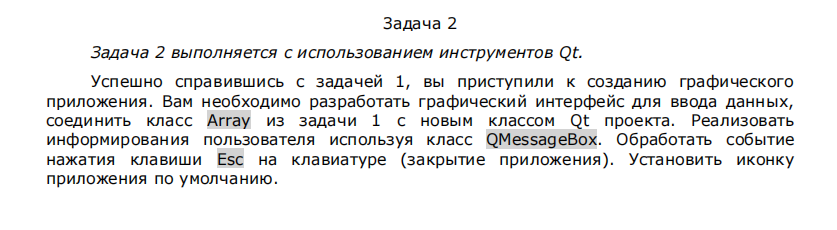
return 0;

}

результат:



Задание 2:



Задание выпонено:

**mainwindow.h**

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include <second.h>

#include <QMessageBox>

#include <QKeyEvent>

QT\_BEGIN\_NAMESPACE

namespace **Ui** { class **MainWindow**; }

QT\_END\_NAMESPACE

class **MainWindow** : public QMainWindow

{

Q\_OBJECT

public:

**MainWindow**(QWidget \*parent = nullptr);

~***MainWindow***();

public slots:

void **test**();

void ***keyPressEvent***(QKeyEvent\* e);

signals:

void **secondSignal**(QString str);

private slots:

void **on\_pushButton\_clicked**();

void **on\_pushButton\_2\_clicked**();

private:

Ui::MainWindow \*ui;

Second \*second;

};

#endif // MAINWINDOW\_H

**mainwindow.cpp**

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include <QDebug>

MainWindow::**MainWindow**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

second = new Second(this);

connect(ui->pushButton, SIGNAL(clicked()),this,SLOT(test()));

connect(this, &MainWindow::secondSignal,second,&Second::secondSlot);

}

MainWindow::~***MainWindow***()

{

delete ui;

}

void MainWindow::***keyPressEvent***(QKeyEvent \*e)

{

if(e->key()==Qt::Key\_Escape)

{

ui->label\_2->setText("Клавиша ESC");

}

}

void MainWindow::**test**()

{

qDebug() << "слот тест запусчен";

emit secondSignal(ui->lineEdit->text());

}

void MainWindow::**on\_pushButton\_clicked**()

{

}

void MainWindow::**on\_pushButton\_2\_clicked**()

{

QMessageBox::critical(this, "CriticalDamage","это гг братан");

QMessageBox::question(this, "medicine is powerless here","хоЧешь продолжИть?");

}

**second.h**

#ifndef SECOND\_H

#define SECOND\_H

#include <QObject>

#include <QMainWindow>

#include <QWidget>

class **Second** : public QObject

{

Q\_OBJECT

public:

explicit **Second**(QObject \*parent = nullptr);

signals:

public slots:

void **secondSlot**(QString str);

};

#endif // SECOND\_H

**second.cpp**

#include "second.h"

#include "second.h"

#include <QDebug>

Second::**Second**(QObject \*parent)

:QObject{*parent*}

{

}

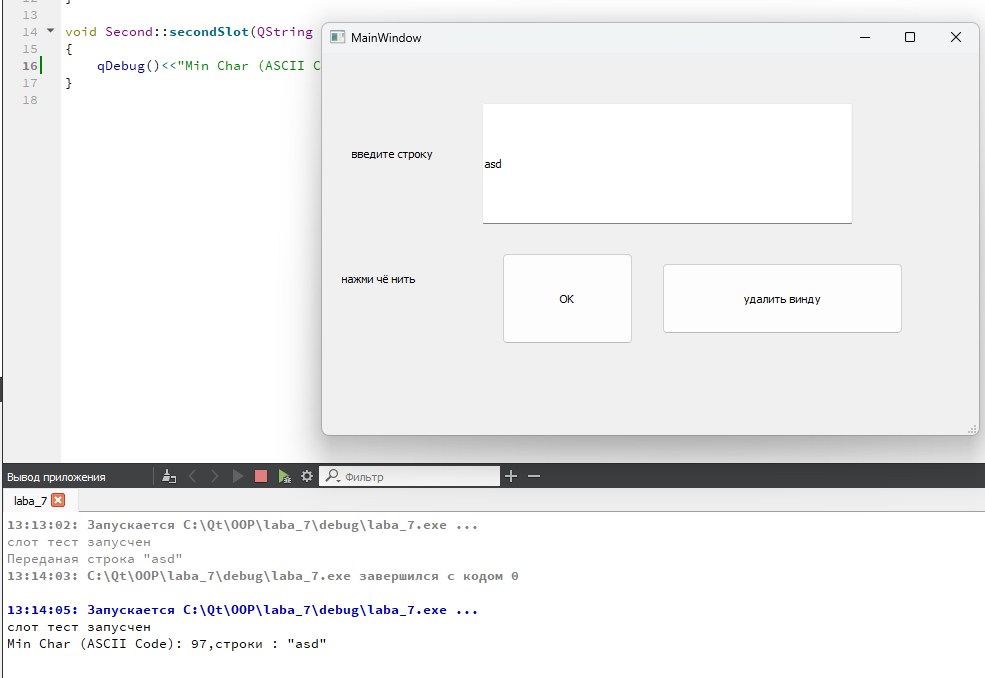
void Second::**secondSlot**(QString str)

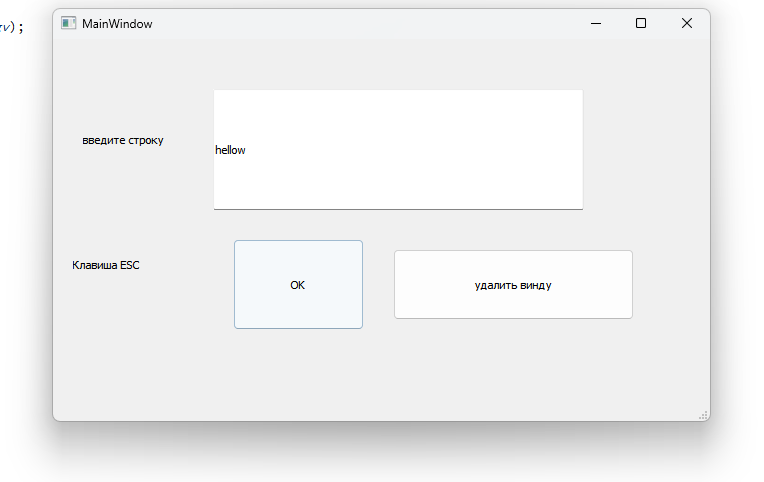
{

qDebug()<<"Min Char (ASCII Code): 97,строки :"<<str;

}

**Оно работает:**



После нажания клавиши ESC   


Сообщения об ошибках работают

