

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

«Московский государственный университет геодезии и картографии»
(МИИГАиК)

Факультет геоинформатики и информационной безопасности
Кафедра Прикладной информатики

Лабораторная работа №1
“Разработка калькулятора с GUI на C++”

Выполнил:

Доровин Андрей Владимирович

2023-ФГиИБ-ПИ-16

Проверил:

Лебедев Евгений Денисович

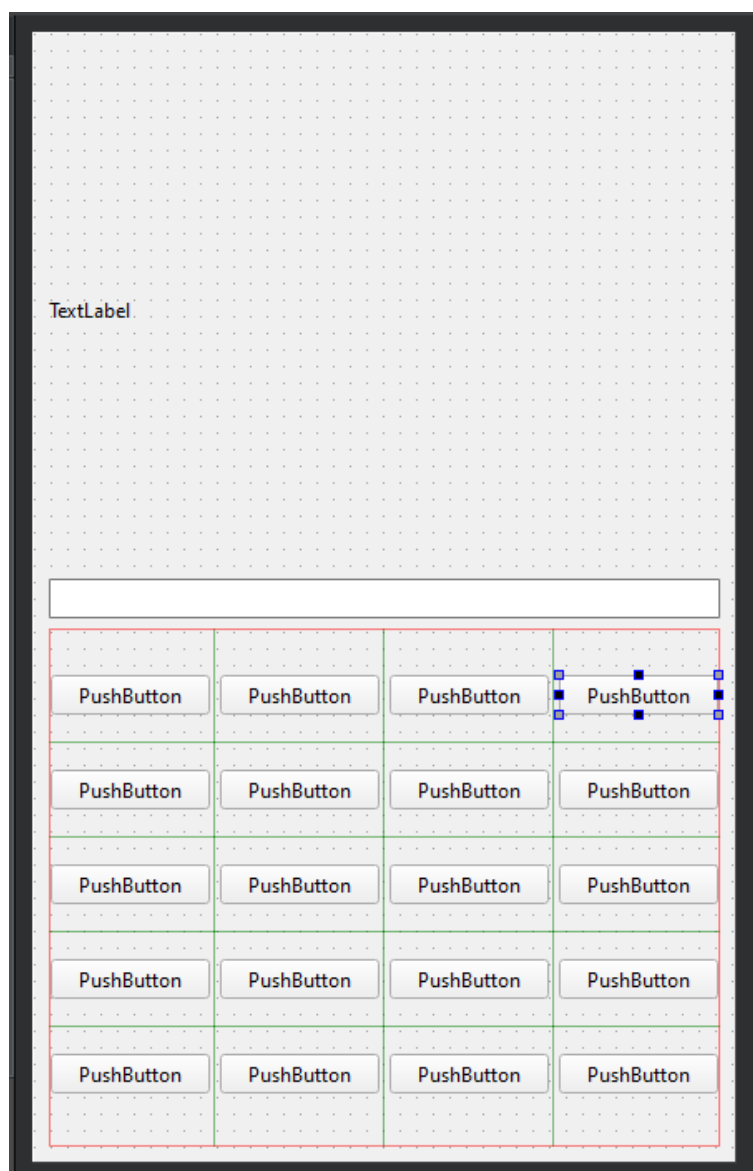
Москва 2023

Задание

Разработка приложения «Калькулятор», который будет выполнять основные арифметические операции (сложение, вычитание, умножение, деление). Калькулятор должен обладать простым и понятным пользовательским интерфейсом и обеспечивать корректное выполнение операций.

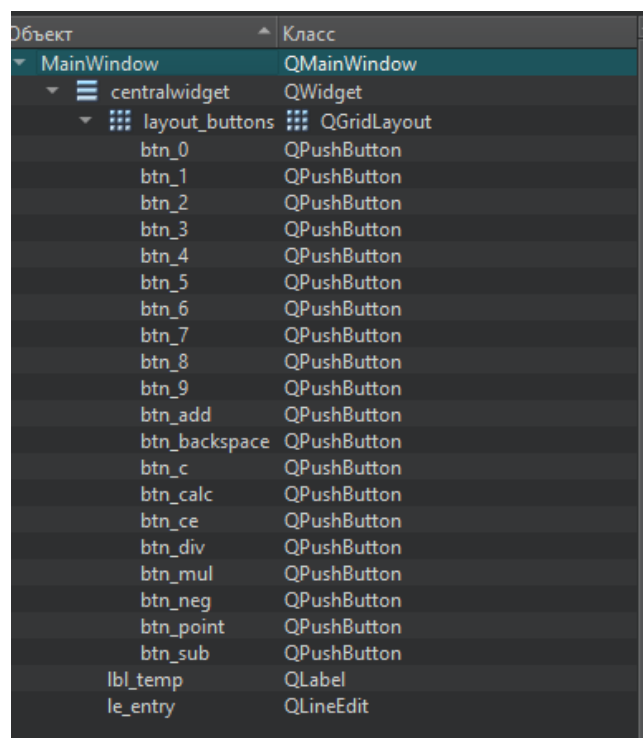
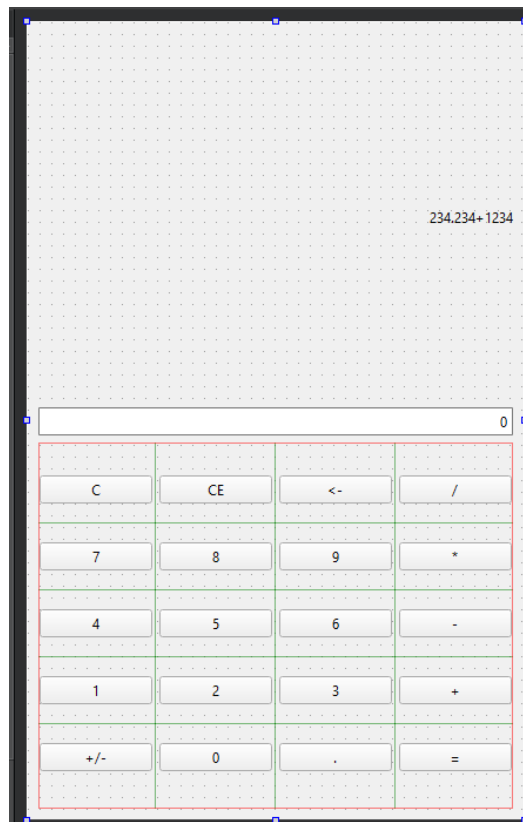
Глава 1. Разработка интерфейса.

Вначале был изменён WindowTitle на “калькулятор”. После я удалил menubar и statusbar. Так же добавил Line Edit, Label и Grid Layout. Все элементы разместил по вертикали (Lay Out Vertically). В гриде разместил 20 PushButton (4x5).



После я переименовал все кнопки и добавил к ним shortcut'ы. В поле Line Edit прописал 0 и сделал правое горизонтальное выравнивание, так же

поставил ему readOnly и приравнял maxLenght к 16. В Label записал выражение и сделал правое горизонтальное выравнивание. После переименовал все элементы, чтоб к ним было проще обращаться.



Далее я настроил вертикальную политику для элементов. Для Label и QLineEdit поставил maximum, а все btn переключил на Expanding. Далее я

изменил styleSheet у MainWindow, вписав туда следующее:

```
QWidget {  
color: white;  
background-color: #121212;  
font-family: Rubik;  
font-size: 16pt;  
font-weight: 600;  
}  
QPushButton {  
background-color: transparent;  
border: none;  
}  
QPushButton:hover {  
background-color: #666;  
}  
QPushButton:pressed {  
background-color: #888;  
}
```

Так же я изменил styleSheet у Label и LineEdit, вписав туда:

```
color: #888;
```

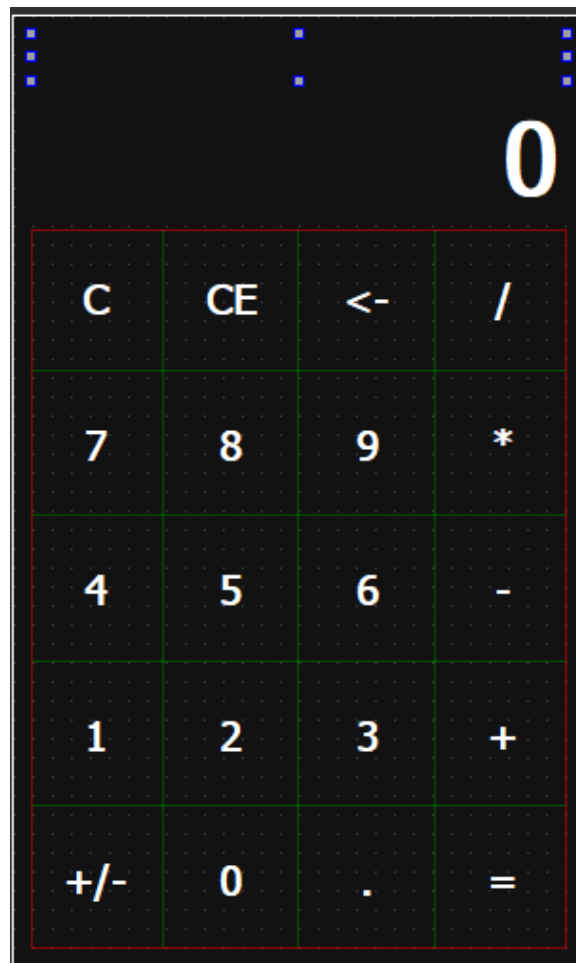
И

```
font-size: 40pt;
```

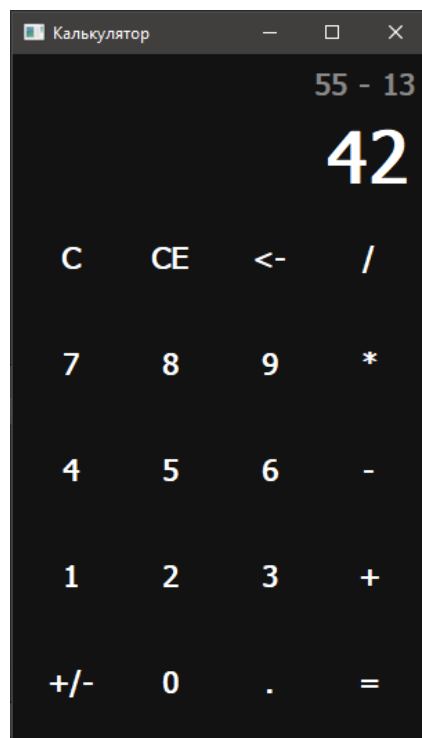
```
border: none;
```

Соответственно.

Из Label убрал текст, указал минимальный и текущий размер основному окну на 300x500 и поставил параметр PointingHand на кнопку С.



Глава 2. Разработка калькулятора.



Кнопки: 1 2 3 4 5 6 7 8 9 0 - + * / . добавляют соответствующий символ в строку

Кнопка <- удаляет один символ строки

Кнопка CE очищает все записи

Кнопка C очищает всё после последней операции =

Кнопка = вычисляет результат строки

Кнопка +/- меняет знак числа

Листинг кода вmainwindow.cpp

```
#include "mainwindow.h"
```

```
#include "../ui_mainwindow.h"
```

```
QString last_state = "";
```

```
MainWindow::MainWindow(QWidget *parent)
```

```
    : QMainWindow(parent)
```

```
    , ui(new Ui::MainWindow)
```

```
{
```

```
    ui->setupUi(this);
```

```
}
```

```
MainWindow::~MainWindow()
```

```
{
```

```
    delete ui;
```

```
}
```

```
void MainWindow::on_btn_ce_clicked()
```

```
{
```

```
    ui->le_entry->clear();
```

```
    ui->lbl_temp->clear();  
}
```

```
void MainWindow::on_btn_backspace_clicked()  
{  
    QString text_entry = ui->le_entry->text();  
    QString text_temp = ui->lbl_temp->text();  
    text_entry.chop(1);  
    text_temp.chop(1);  
    ui->le_entry->setText(text_entry);  
    ui->lbl_temp->setText(text_temp);  
}
```

```
void MainWindow::on_btn_point_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + ".");  
  
}
```

```
void MainWindow::on_btn_sub_clicked()  
{  
    QString text = ui->le_entry->text();  
    if (text.isEmpty()) {  
        return;  
    }  
    if (text.at(0) == '-') {  
        text.remove(0, 1);  
    }  
}
```

```
    } else {  
        text.prepend("-");  
    }  
    ui->le_entry->setText(text);  
}
```

```
void MainWindow::on_btn_0_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_0->text());  
}
```

```
void MainWindow::on_btn_1_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_1->text());  
}
```

```
void MainWindow::on_btn_2_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_2->text());  
}
```

```
void MainWindow::on_btn_3_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_3->text());  
}
```

```
void MainWindow::on_btn_4_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_4->text());  
}
```



```
}
```

```
void MainWindow::on_btn_5_clicked()
```

```
{
```

```
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_5->text());
```

```
}
```

```
void MainWindow::on_btn_6_clicked()
```

```
{
```

```
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_6->text());
```

```
}
```

```
void MainWindow::on_btn_7_clicked()
```

```
{
```

```
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_7->text());
```

```
}
```

```
void MainWindow::on_btn_8_clicked()
```

```
{
```

```
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_8->text());
```

```
}
```

```
void MainWindow::on_btn_9_clicked()
```

```
{
```

```
    ui->lbl_temp->setText(ui->lbl_temp->text() + ui->btn_9->text());
```

```
}
```

```
void MainWindow::on_btn_div_clicked()
```

```
{
```

```
    ui->lbl_temp->setText(ui->lbl_temp->text() + " " + ui->btn_div->text() +  
    " ");  
}
```

```
void MainWindow::on_btn_mul_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + " " + ui->btn_mul->text()  
+ " ");  
}
```

```
void MainWindow::on_btn_neg_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + " " + ui->btn_neg->text() +  
    " ");  
}
```

```
void MainWindow::on_btn_add_clicked()  
{  
    ui->lbl_temp->setText(ui->lbl_temp->text() + " " + ui->btn_add->text()  
+ " ");  
}
```

```
void MainWindow::on_btn_c_clicked()  
{  
    ui->lbl_temp->clear();  
    ui->lbl_temp->setText(ui->lbl_temp->text() + last_state);  
}
```

```
void MainWindow::on_btn_calc_clicked()  
{
```

```

    QString expression = ui->lbl_temp->text();
    QString result = calculateExpression(expression);
    ui->le_entry->setText(result);
    last_state = expression;
}

```

```

QString MainWindow::calculateExpression(const QString &expression) {
    QStringList tokens = expression.split(" ");
    QList<QString> numbers;
    QList<QString> operators;
    for (const QString &token : tokens) {
        if (token != "+" && token != "-" && token != "*" && token != "/") {
            // Если токен не является оператором, добавляем его в список
чисел
            numbers.push_back(token);
        } else {
            // Если токен - оператор, добавляем его в список операторов
            operators.push_back(token);
        }
    }
    // Выполняем операции умножения и деления
    for (int i = 0; i < operators.size(); ++i) {
        if (operators.at(i) == "*" || operators.at(i) == "/") {
            // Извлекаем первое число
            double a = numbers.at(i).toDouble();
            // Извлекаем оператор
            QString op = operators.at(i);
            // Извлекаем второе число

```

```
double b = numbers.at(i + 1).toDouble();
// Выполняем операцию в зависимости от оператора
if (op == "*") {
    numbers[i] = QString::number(a * b);
} else if (op == "/") {
    if (b == 0.0) {
        return "Error: Division by zero";
    }
    numbers[i] = QString::number(a / b);
}
// Удаляем использованный оператор и второе число
operators.removeAt(i);
numbers.removeAt(i + 1);
// Уменьшаем индекс, чтобы не пропустить следующий
оператор
--i;
}
}
// Выполняем операции сложения и вычитания
while (!operators.isEmpty()) {
    // Извлекаем первое число
    double a = numbers.takeFirst().toDouble();
    // Извлекаем оператор
    QString op = operators.takeFirst();
    // Извлекаем второе число
    double b = numbers.takeFirst().toDouble();
    // Выполняем операцию в зависимости от оператора
    if (op == "+") {
```

```

        numbers.push_front(QString::number(a + b));
    } else if (op == "-") {
        numbers.push_front(QString::number(a - b));
    }
}

// В списке должен остаться один элемент - результат выражения
if (numbers.size() == 1) {
    QString result = numbers.takeFirst();
    qDebug() << "Result:" << result;
    return result;
} else {
    return "Error";
}
}

```

Листинг кода в mainwindow.h

```

#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
#include <QWidget>
#include <QVBoxLayout>
#include <QLineEdit>
#include <QLabel>
#include <QGridLayout>
#include <QPushButton>

QT_BEGIN_NAMESPACE
namespace Ui {

```

```
class MainWindow;
```

```
}
```

```
QT_END_NAMESPACE
```

```
class MainWindow : public QMainWindow
```

```
{
```

```
    Q_OBJECT
```

```
public:
```

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

```
    ~MainWindow();
```

```
private slots:
```

```
    void on_btn_c_clicked();
```

```
    void on_btn_ce_clicked();
```

```
    void on_btn_backspace_clicked();
```

```
    void on_btn_point_clicked();
```

```
    void on_btn_1_clicked();
```

```
    void on_btn_0_clicked();
```

```
    void on_btn_2_clicked();
```

```
    void on_btn_3_clicked();
```

void on_btn_4_clicked();

void on_btn_5_clicked();

void on_btn_6_clicked();

void on_btn_7_clicked();

void on_btn_8_clicked();

void on_btn_9_clicked();

void on_btn_sub_clicked();

void on_btn_div_clicked();

void on_btn_mul_clicked();

void on_btn_neg_clicked();

void on_btn_add_clicked();

void on_btn_calc_clicked();

private:

Ui::MainWindow *ui;

QWidget *centralWidget;

QVBoxLayout *layout;

QLineEdit *le_entry;

```

QLabel *lbl_temp;
QGridLayout *gridLayout_buttons;
QPushButton *btn_0;
QPushButton *btn_1;
QPushButton *btn_2;
QPushButton *btn_3;
QPushButton *btn_4;
QPushButton *btn_5;
QPushButton *btn_6;
QPushButton *btn_7;
QPushButton *btn_8;
QPushButton *btn_9;
QPushButton *btn_div;
QPushButton *btn_mul;
QPushButton *btn_neg;
QPushButton *btn_add;
QPushButton *btn_calc;
QString calculateExpression(const QString &expression);

};
#endif // MAINWINDOW_H

```

Описание	Входные данные	Выходные данные	Ожидаемый результат
Сумма целых	5+5	10	10
Сумма дробных	5.5 + 4	9.5	9.5
Сумма дробных с целым ответом	2.5 + 1.5	4	4
Разность целых	5 – 4	1	1
Разность дробных	4.5-2	2.5	2.5
Деление нацело	10 \ 2	5	5

Деление с остатком	$5 \setminus 3$	1.66666666667	1.6666666667
Умножение положительных	$5 * 5$	25	25
Умножение отрицательного на положительное	$-5 * 5$	-25	-25
Деление отрицательного на положительное	$-5 \setminus 2$	-2.5	-2.5
Проверка порядка вычислений	$5 + 5 * 2$	15	15
Умножение двух дробных	$5.2 * 5.2$	27.04	27.04
Деление на 0	$5 \setminus 0$	Error: Division	Error
Умножение на 0	$-5 * 0$	0	0
Деление дробных	$5.1 \setminus 5.1$	1	1