МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный университет геодезии и картографии»

(МИИГАиК)

Факультет геоинформатики и информационной безопасности

Кафедра Прикладной информатики

**Лабораторная работа №1  
 «Разработка калькулятора с GUI на С++»**

Выполнил: Ширшов Даниил Федорович

Студент группы: 2023-ФГиИБ-ПИ-1

Проверил:

Лебедев Евгений Денисович

Москва 2023

Оглавление

1. Задание……………………………………….3
2. Глава 1. Разработка интерфейса……………3
3. Глава 2. Разработка калькулятора…………6
4. Тестирование………………………………18

**Задание**

Разработка приложения «калькулятор», который будет

выполнять основные арифметические операции (сложение, вычитание,

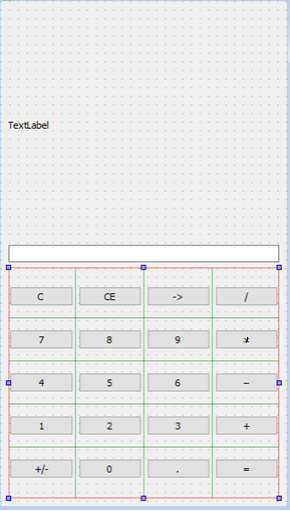
умножение, деление). Калькулятор должен обладать простым и

понятным пользовательским интерфейсом и обеспечивать корректное

выполнение операций.

**Глава 1. Разработка интерфейса**

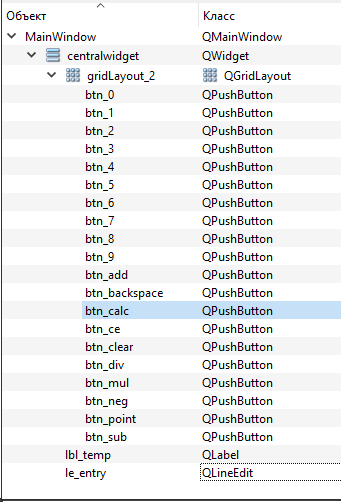
На начале проекта были добавлены поля ввода(Label) и вывода(Line Edit) , а также основные кнопки калькулятора(Grid Layout)



Далее было использовано выравнивание по середине (Lay Out Vertically) и были изменены свойства, используемых элементов:

1. Проставление горячих клавиш для всех кнопок
2. Line Edit:
   1. правое горизонтальное выравнивание для текста
   2. свойство readOnly.
   3. максимальной длины в 16 символов
   4. вертикальная политика : Maximum
3. Lable:
   1. правое горизонтальное выравнивание для текста
   2. вертикальная политика : Maximum
4. Кнопки:
   1. вертикальная политика : Expanding

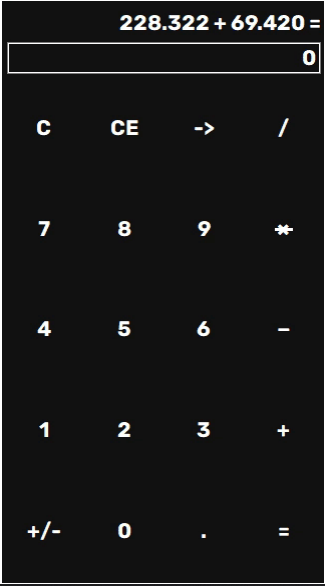
Далее были изменены названия элементов, чтобы в коде было проще обращаться к ним.



Используемая цветовая палитра:

1. Почти черный #121212 для фона.
2. Белый #FFF для текста кнопок и поля ввода.
3. Серый #666 для фона кнопок при наведении
4. Серый посветлее #888 для текста временного выражения и фона кнопок при нажатии.

Промежуточный вариант калькулятора

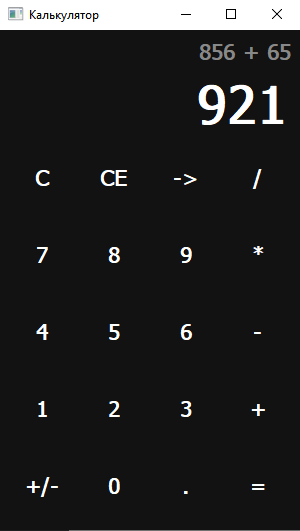


Следующие изменения Line Edit, Label и размера окна приложения:

1. Line Edit:
   1. font-size: 40pt;
   2. border: none;
2. Label:
   1. color: #888;
3. Окно приложения
   1. Размер 300 х 500

**Глава 2. Разработка калькулятора**

Итоговый вариант приложения



**Функции кнопок:**

1. Кнопки «0-9» добавляют соответствующую цифру в строку
2. Кнопка «.» добавляет «.» в строку
3. Кнопка «\*» добавляет «\*» в строку
4. Кнопка «/» добавляет «/» в строку
5. Кнопка «-» добавляет «-» в строку
6. Кнопка «+» добавляет «+» в строку
7. Кнопка «->» стирает последний добавленный элемент
8. Кнопка «CE» стирает все записи
9. Кнопка «С» стирает всё после использованного «=»
10. Кнопка «+/-» меняют знак числа
11. Кнопка «=» выполняет вычисления

**Листинг кода 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\_clear\_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

**Листинг кода mainwindow.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() + " ");

}

QString MainWindow::calculateExpression(const QString &expression) {

// Выводим токены для отладки

qDebug() << "Expression:" << expression;

QStringList tokens = expression.split(" ");

qDebug() << "Tokens:" << tokens;

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";

}

}

void MainWindow::on\_btn\_calc\_clicked() {

QString expression = ui->lbl\_temp->text();

QString result = calculateExpression(expression);

ui->le\_entry->setText(result);

last\_state = expression;

}

void MainWindow::on\_btn\_clear\_clicked()

{

ui->lbl\_temp->clear();

ui->lbl\_temp->setText(ui->lbl\_temp->text() + last\_state);

}

**Тестирование**

|  |  |  |  |
| --- | --- | --- | --- |
| Описание теста | вводные  данные | выходные данные | результат |
| Сложение чисел | 1 + 9 = | 10 | сумма чисел |
| Вычитание чисел | 9 – 1 = | 8 | разность чисел |
| Деление на 0 | 8 / 0 | Error: Division | На ноль делить нельзя, выводится ошибка |
| Неверный порядок | 6 \* \* 8 | 0 | Ожидается, что числа и операторы будут чередоваться. Следовательно неверное решение |
| Сложение дробных чисел | 7.5 + 2.5 | 10 | Сумма чисел |
| Разность двух дробных чисел | 7.5 – 4.5 | 3 | Разность чисел |
| Умножение больших чисел | 53535353 \* 10000000 | 35354е+14 | Большое число |
| Смена знака | 4 \* 5 = +/- | -20 | Число меняет знак |
| Стереть всё | 7 + 10 = СЕ |  | Всё стирается |
| Стереть всё после = | 8 – 2 = + 6 С | 6 | Всё стирается после = |
| Сумма дробного и целого чисел | 0.5 + 10 | 10.5 | Сумма чисел |
| Разность дробного и целого чисел | 5.5 – 3 = | 2.5 | Разность чисел |
| Стереть последний введённый символ | 55 -> | 5 | Стирает последний введённый символ |
| Сумма трёх чисел | 5 + 5 + 10 = | 20 | Сумма чисел |
| Проверка порядка выполнения операторов | 5 + 10 / 2 \* 3 | 20 | Верный порядок выполнения операторов |