

ПЕТРОЗАВОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНСТИТУТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
КАФЕДРА ИНФОРМАТИКИ И МАТЕМАТИЧЕСКОГО ОБЕСПЕЧЕНИЯ

Направление подготовки бакалавриата

Отчет по практике

РАЗРАБОТКА ПРИЛОЖЕНИЯ
«КАЛЬКУЛЯТОР»

Выполнили: Лашко Г.М. Бровкин А.П.
студенты 1 курса группы 22107

Г. М. Лашко _____
подпись

А. П. Бровкин _____
подпись

Содержание

Введение	3
1 Требования к приложению	4
2 Проектирование приложения	5
3 Реализация приложения	6
Заключение	10

Введение

Цель проекта: Сделать рабочую версию калькулятора

Задачи проекта: Создать дизайн, реализовать нажатие на цифры, реализовать математические функции

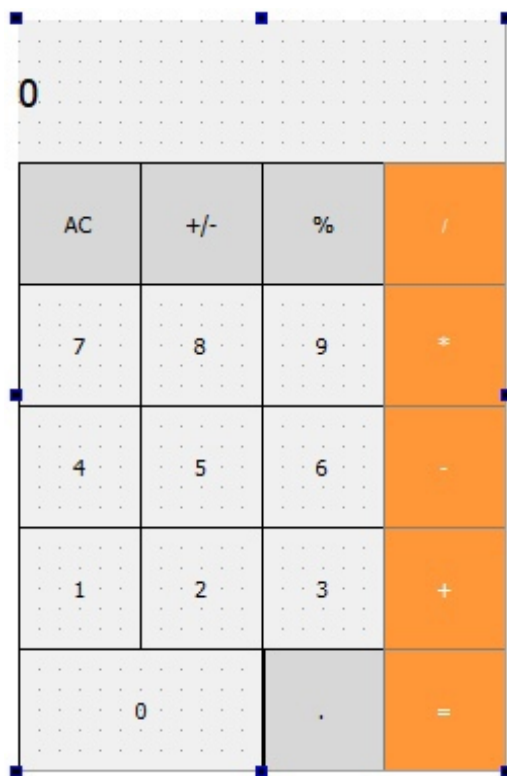


Рис. 1: Дизайн калькулятора

1 Требования к приложению

Приложение должно выполнять все функции стандартного калькулятора и выводить результат вычисления на экран

2 Проектирование приложения

Приложение состоит из нескольких модулей:

*Функция кнопок в которой занесены все кнопки с сигналами и слотами для них

- Тело калькулятора

*Функции операций процента и смены знака

- Реализация процента - деление числа на 100

- Реализация смены знака - в строке вывода меняется знак числа

*Функция кнопки точка

- Заключается в том, чтобы можно было ставить точку (делать число десятичным), ставится не более одного

*Функции математических операция

- Реализация плюса, минуса, деления (при делении на 0, выводит "скрытое"сообщение) и умножения

*Функция сброса калькулятора

- Сбрасывает окно вывода, выводя специальное для этого сообщение

*Функция результата

- взаимодействия всех операций с модулями чисел

3 Реализация приложения

Приложение написано на языке C++

Используемые библиотеки: <QMainWindow>; <QApplication>

Кол-во модулей: 20

Кол-во функций: 6

Строк кода: 144 (не считая 493 строк в mainwindow.ui)

Функция кнопок в которой занесены все кнопки с сигналами и слотами для них

```
1 MainWindow::MainWindow(QWidget *parent)
2     : QMainWindow(parent)
3     , ui(new Ui::MainWindow)
4 {
5     ui->setupUi(this);
6
7     connect(ui->pushButton_0 , SIGNAL(clicked()), this, SLOT(digits_numbers()));
8     connect(ui->pushButton_1, SIGNAL(clicked()), this, SLOT(digits_numbers()));
9     connect(ui->pushButton_2, SIGNAL(clicked()), this, SLOT(digits_numbers()));
10    connect(ui->pushButton_3, SIGNAL(clicked()), this, SLOT(digits_numbers()));
11    connect(ui->pushButton_4, SIGNAL(clicked()), this, SLOT(digits_numbers()));
12    connect(ui->pushButton_5, SIGNAL(clicked()), this, SLOT(digits_numbers()));
13    connect(ui->pushButton_6, SIGNAL(clicked()), this, SLOT(digits_numbers()));
14    connect(ui->pushButton_7, SIGNAL(clicked()), this, SLOT(digits_numbers()));
15    connect(ui->pushButton_8, SIGNAL(clicked()), this, SLOT(digits_numbers()));
16    connect(ui->pushButton_9, SIGNAL(clicked()), this, SLOT(digits_numbers()));
17    connect(ui->pushButton_plus_minus, SIGNAL(clicked()), this, SLOT(operations()));
18    connect(ui->pushButton_proc, SIGNAL(clicked()), this, SLOT(operations()));
19    connect(ui->pushButton_plus, SIGNAL(clicked()), this, SLOT(math_operations()));
20    connect(ui->pushButton_minus, SIGNAL(clicked()), this, SLOT(math_operations()));
21    connect(ui->pushButton_div, SIGNAL(clicked()), this, SLOT(math_operations()));
22    connect(ui->pushButton_mult, SIGNAL(clicked()), this, SLOT(math_operations()));
23
24    ui->pushButton_div->setCheckable(true);
25    ui->pushButton_mult->setCheckable(true);
26    ui->pushButton_plus->setCheckable(true);
27    ui->pushButton_minus->setCheckable(true);
28
```

29 }

Функции операций процента и смены знака

```
1 void MainWindow::operations()
2 {
3     QPushButton *button = (QPushButton *)sender();
4
5     double all_numbers;
6     QString new_label;
7
8     if(button->text() == "+/-"){
9         all_numbers = (ui->label->text()).toDouble();
10        all_numbers = all_numbers * -1;
11        new_label = QString::number(all_numbers, 'g', 15);
12
13        ui->label->setText(new_label);
14    } else if(button->text() == "%"){
15        all_numbers = (ui->label->text()).toDouble();
16        all_numbers = all_numbers * 0.01;
17        new_label = QString::number(all_numbers, 'g', 15);
18
19        ui->label->setText(new_label);
20    }
21 }
```

Функция кнопки точка

```
1 void MainWindow::on_pushButton_dot_clicked()
2 {
3     if(!(ui->label->text().contains('.')'))
4         ui->label->setText(ui->label->text() + ".");
5
6 }
```

Функции математических операций

```
1 void MainWindow::math_operations()
2 {
3     QPushButton *button = (QPushButton *)sender();
4
5     num_first = ui->label->text().toDouble();
6     ui->label->setText("");
7     button->setChecked(true);
8 }
```

Функция сброса калькулятора

```
1 void MainWindow::on_pushButton_AC_clicked()
2 {
3     ui->pushButton_plus->setChecked(false);
4     ui->pushButton_minus->setChecked(false);
5     ui->pushButton_div->setChecked(false);
6     ui->pushButton_mult->setChecked(false);
7
8     ui->label->setText("стёр, Хозяин");
9 }
```

Функция результата

```
1 void MainWindow::on_pushButton_result_clicked()
2 {
3     double label_number, num_second;
4     QString new_label;
5
6     num_second = ui->label->text().toDouble();
7
8     if(ui->pushButton_plus->isChecked()) {
9         label_number = num_first + num_second;
10        new_label = QString::number(label_number, 'g', 15);
11
12        ui->label->setText(new_label);
13        ui->pushButton_plus->setChecked(false);
14    } else if(ui->pushButton_minus->isChecked()) {
```



```

15     label_number = num_first - num_second;
16     new_label = QString::number(label_number, 'g', 15);
17
18     ui->label->setText(new_label);
19     ui->pushButton_minus->setChecked(false);
20
21 } else if(ui->pushButton_div->isChecked()) {
22     if (num_second == 0) {
23         ui->label->setText("раз такой умный сам делай");
24     } else {
25         label_number = num_first / num_second;
26         new_label = QString::number(label_number, 'g', 15);
27
28         ui->label->setText(new_label);
29     }
30     ui->pushButton_div->setChecked(false);
31
32 } else if(ui->pushButton_mult->isChecked()) {
33     label_number = num_first * num_second;
34     new_label = QString::number(label_number, 'g', 15);
35
36     ui->label->setText(new_label);
37     ui->pushButton_mult->setChecked(false);
38 }
39 }

```

Заключение

В итоге мы получили рабочую версию стандартного калькулятора