



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т (В А Р И А Н Т 1 2)

по лабораторной работе № 9

Название: Программирование с использованием Qt

Дисциплина: Объектно-ориентированное программирование

Студент

ИУ6-23Б

(Группа)

13.05.2023

(Подпись, дата)

В.К. Залыгин

(И.О. Фамилия)

Преподаватель

А.М. Минитаева

(Подпись, дата)

(И.О. Фамилия)

Москва, 2023

Цель работы

Изучение основ работы с библиотекой QT на языке программирования C++.
Ознакомление с основными средствами работы для построения приложений с графическим интерфейсом.

Часть 1

Создание простого приложения На рисунке 5 показан внешний вид простого приложения, в котором предлагается ввести возраст с использованием одного из трех вариантов ввода: 1) непосредственного ввода числа, 2) посредством стрелок (элемент типа QSpinBox), последовательно увеличивающих или уменьшающих значение, 3) с помощью специального ползунка (слайдера – элемент типа QSlider).

Задание

Замените в программе схему выравнивания QHBoxLayout на QVBoxLayout и зафиксируйте результат.

Текст программы

```
1  #include "mainwindow.h"
2  #include <QApplication>
3  #include <QLocale>
4  #include <QTranslator>
5  #include <QDebug>
6  int main(int argc, char *argv[])
7  {
8      QApplication a(argc, argv);
9
10     QTranslator translator;
11     const QStringList uiLanguages = QLocale::system().uiLanguages();
12     for (const QString &locale : uiLanguages) {
13         const QString baseName = "t1_" + QLocale(locale).name();
14         if (translator.load(":/i18n/" + baseName)) {
15             a.installTranslator(&translator);
16             break;
17         }
18     }
19     MainWindow w;
20     w.show();
21     return a.exec();
22 }
```

Рисунок 1 - main.cpp

```

1  #include "mainwindow.h"
2  #include "./ui_mainwindow.h"
3
4  MainWindow::MainWindow(QWidget *parent)
5      : QMainWindow(parent)
6      , ui(new Ui::MainWindow)
7  {
8      ui->setupUi(this);
9  }
10
11  MainWindow::~MainWindow()
12  {
13      delete ui;
14  }
15
16
17  void MainWindow::on_pushButton_clicked()
18  {
19      qDebug(). <<"Программа завершена корректно";
20  }
21

```

Рисунок 2 - mainwindow.cpp

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5
6  QT_BEGIN_NAMESPACE
7  namespace Ui { class MainWindow; }
8  QT_END_NAMESPACE
9
10  class MainWindow : public QMainWindow
11  {
12      Q_OBJECT
13
14  public:
15      MainWindow(QWidget *parent = nullptr);
16      ~MainWindow();
17
18  private slots:
19      void on_pushButton_clicked();
20
21  private:
22      Ui::MainWindow *ui;
23  };
24  #endif // MAINWINDOW_H
25

```

Рисунок 3 - mainwindow.h

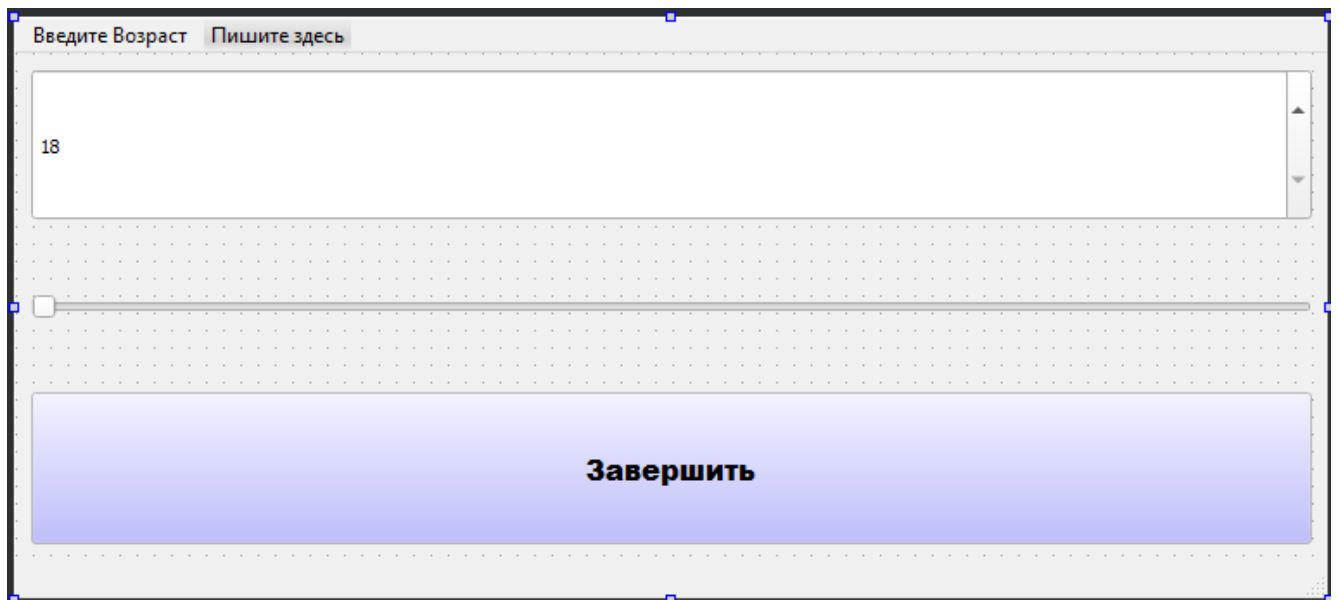


Рисунок 4 - интерфейс
Тестовые данные

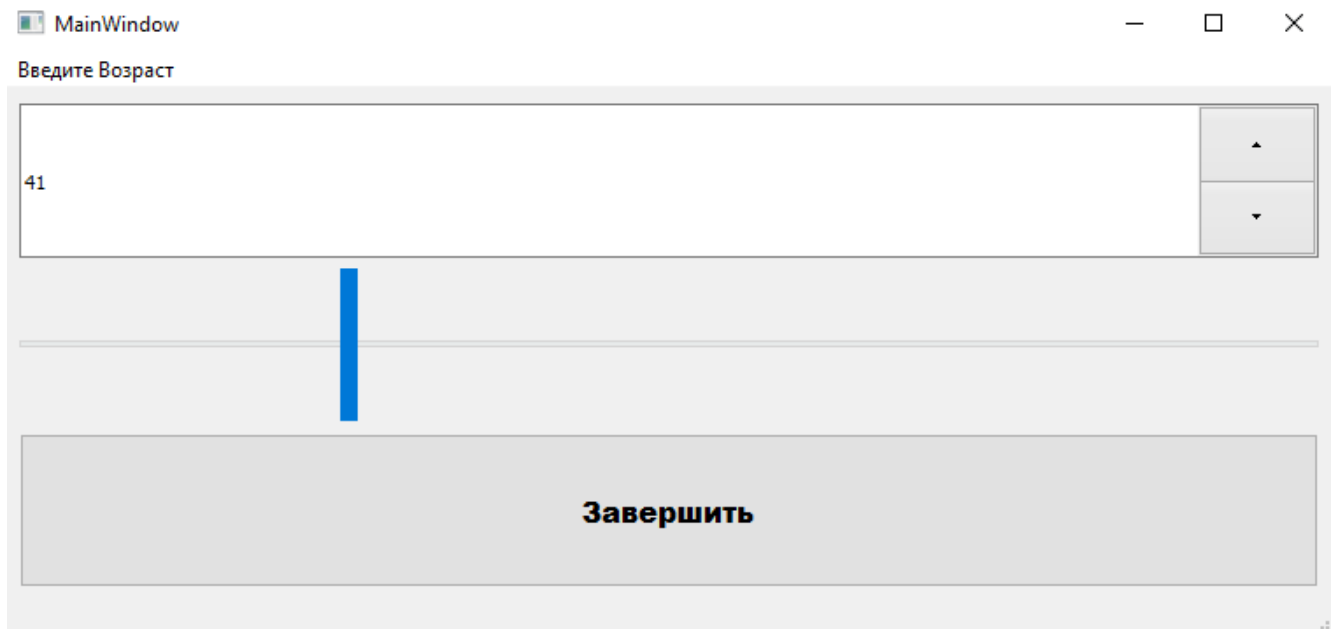


Рисунок 5 - пример работы

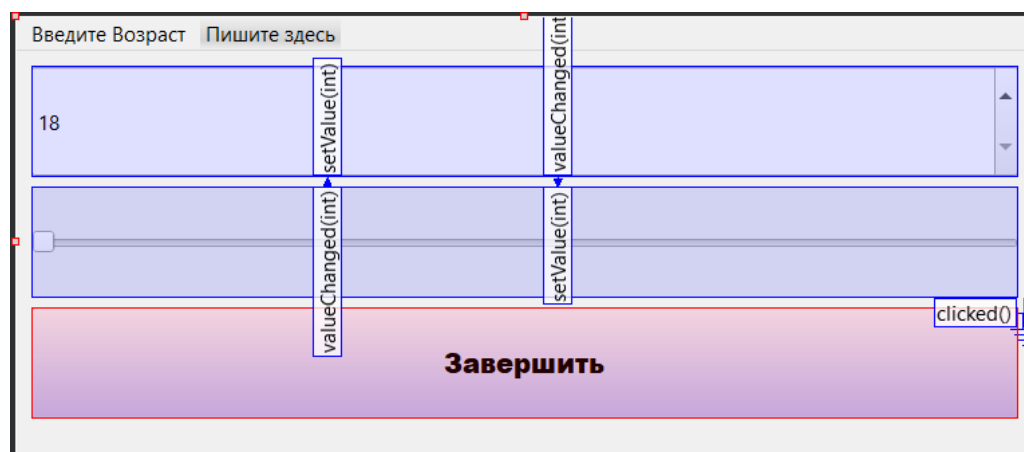


Рисунок 6 - слоты

Часть 2

Создание простого приложения в QtDesigner Рассмотрим разработку прототипов диалогов в QtDesigner. В предыдущем задании были показаны простейшие способы выравнивания виджетов. QtDesigner, помимо разнообразных виджетов, позволяет использовать так называемые менеджеры компоновки, работа с которыми и будет рассмотрена в задании

Задание

Измените тип разделителя с QSplitter(Qt::Horizontal); на QSplitter(Qt::Vertical); и зафиксируйте полученный результат.

Текст программы

```
1  #include "mainwindow.h"
2
3  #include <QApplication>
4  #include <QLocale>
5  #include <QTranslator>
6
7  int main(int argc, char *argv[])
8  {
9      QApplication a(argc, argv);
10
11     QTranslator translator;
12     const QStringList uiLanguages = QLocale::system().uiLanguages();
13     for (const QString &locale : uiLanguages) {
14         const QString baseName = "t2_" + QLocale(locale).name();
15         if (translator.load(":/i18n/" + baseName)) {
16             a.installTranslator(&translator);
17             break;
18         }
19     }
20     MainWindow w;
21     w.show();
22     return a.exec();
23 }
```

Рисунок 7 - main.cpp

```
1  #include "mainwindow.h"
2  #include "../ui_mainwindow.h"
3  #include <QMessageBox>
4
5  MainWindow::MainWindow(QWidget *parent)
6      : QMainWindow(parent)
7      , ui(new Ui::MainWindow)
8  {
9      ui->setUi(this);
10 }
11
12 MainWindow::~MainWindow()
13 {
14     delete ui;
15 }
16
17
18 void MainWindow::on_pushButton_clicked() {
19     if(QMessageBox::question(this, "What?", "close?", QMessageBox::Yes | QMessageBox::No) == QMessageBox::Yes) {
20         QApplication::quit();
21     }
22     else show();
23 }
24
25
26
27 void MainWindow::on_pushButton_3_clicked()
28 {
29     if(QMessageBox::question(this, "What?", "close?", QMessageBox::Yes | QMessageBox::No) == QMessageBox::Yes) {
30         QApplication::quit();
31     }
32     else show();
33 }
34 }
```

Рисунок 8 - mainwindow.cpp

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5
6  QT_BEGIN_NAMESPACE
7  namespace Ui { class MainWindow; }
8  QT_END_NAMESPACE
9
10 class MainWindow : public QMainWindow
11 {
12     Q_OBJECT
13
14     public:
15
16     MainWindow(QWidget *parent = nullptr);
17     ~MainWindow();
18
19     private slots:
20     void on_pushButton_clicked();
21
22     void on_pushButton_3_clicked();
23
24     private:
25     Ui::MainWindow *ui;
26
27 };
28 #endif // MAINWINDOW_H

```

Рисунок 9 - mainwindow.h

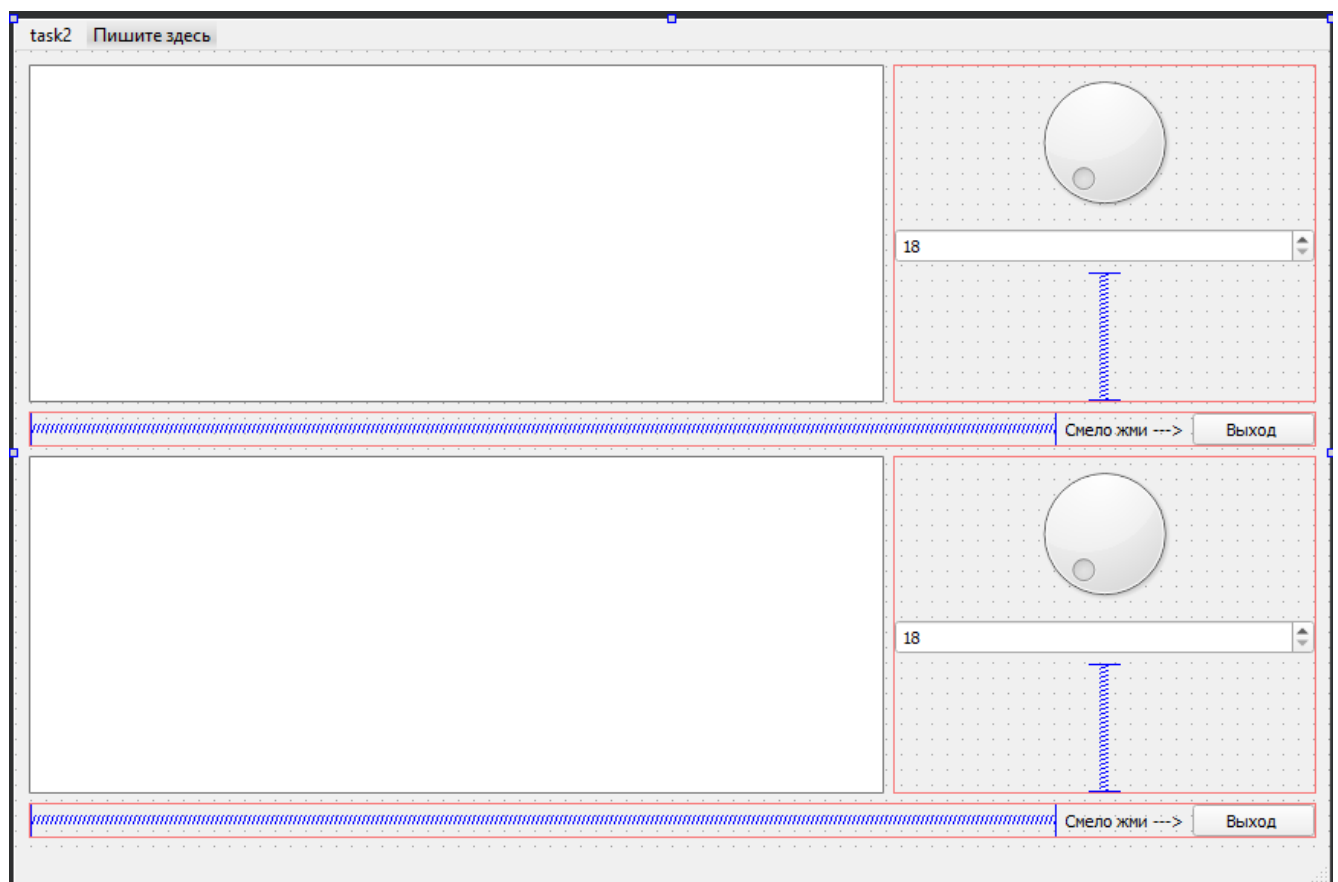


Рисунок 10 – интерфейс

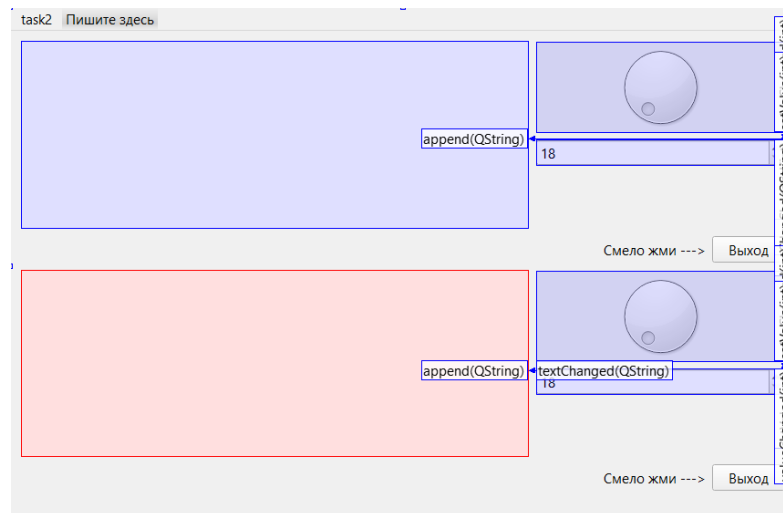


Рисунок 11 - слоты
Тестовые данные



Рисунок 12 - пример работы

Часть 3

Разработка калькулятора

Исходные данные: рассмотрим создание более сложного приложения, наглядно иллюстрирующего возможности библиотеки Qt в части компактного кода программы и динамического создания элементов управления. Разработаем калькулятор, внешний вид которого представлен на рисунке

Задание

Добавьте кнопки, выполняющие: бинарные операции $x \text{ } y$, $\log_y x$ (по аналогии с операциями $+$, $-$, $/$, $*$), а также унарные $\sin(x)$ и $\cos(x)$ (по аналогии с операцией $-/+$)

и разместите этот ряд кнопок вертикально, слева от цифровых кнопок с использованием нового объекта выравнивания (Layout).

Текст программы

```
1  #include "mainwindow.h"
2
3  #include <QApplication>
4  #include <QLocale>
5  #include <QTranslator>
6
7  int main(int argc, char *argv[])
8  {
9      QApplication a(argc, argv);
10
11      QTranslator translator;
12      const QStringList uiLanguages = QLocale::system().uiLanguages();
13      for (const QString &locale : uiLanguages) {
14          const QString baseName = "t3_" + QLocale(locale).name();
15          if (translator.load(":/i18n/" + baseName)) {
16              a.installTranslator(&translator);
17              break;
18          }
19      }
20      MainWindow w;
21      w.show();
22      return a.exec();
23  }
```

Рисунок 13 - main.cpp


```

1  #include "mainwindow.h"
2  #include "../ui_mainwindow.h"
3  #include <QtCore/qmath.h>
4  double num_first=0;
5
6  void backspace(QString& text) {
7      if (text.isEmpty()) {
8          // Если строка пустая, ничего не делаем
9          return;
10     }
11
12     // Проверяем, есть ли точка в числе
13     bool has_decimal_point = text.contains('.');
14
15     if (has_decimal_point) {
16         // Если после точки только одна цифра, удаляем ее вместе с точкой
17         int decimal_index = text.indexOf('.');
18         int num_digits_after_decimal = text.length() - decimal_index - 1;
19         if (num_digits_after_decimal == 1) {
20             text.chop(2);
21         } else {
22             // Удаляем только последнюю цифру после точки
23             text.chop(1);
24         }
25     } else {
26         // Удаляем последнюю цифру
27         text.chop(1);
28     }
29
30     // Если после удаления осталась только точка, удаляем ее
31     if (text.endsWith('.')) {
32         text.chop(1);
33     }
34 }
35 double calculateLog(double a, double x) {
36     return qLn(x) / qLn(a);
37 }
38 double calculatePower(double x, double a) {
39     return qPow(x, a);
40 }

```

Рисунок 14 - mainwindow.cpp

```

42   MainWindow::MainWindow(QWidget *parent)
43       : QMainWindow(parent)
44       , ui(new Ui::MainWindow){
45       ui->setupUi(this);
46
47       connect(ui->nule_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
48       connect(ui->one_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
49       connect(ui->two_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
50       connect(ui->three_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
51       connect(ui->four_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
52       connect(ui->five_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
53       connect(ui->six_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
54       connect(ui->seven_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
55       connect(ui->eight_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
56       connect(ui->nine_but, SIGNAL(clicked()), this, SLOT(digitsNumbers()));
57
58       connect(ui->Inverse_but, SIGNAL(clicked()), this, SLOT(operations()));
59       connect(ui->bck_but, SIGNAL(clicked()), this, SLOT(operations()));
60       connect(ui->ClearEm_but, SIGNAL(clicked()), this, SLOT(operations()));
61
62       connect(ui->Plus_but, SIGNAL(clicked()), this, SLOT(math_operations()));
63       connect(ui->Minus_but, SIGNAL(clicked()), this, SLOT(math_operations()));
64       connect(ui->Sin_but, SIGNAL(clicked()), this, SLOT(operations()));
65       connect(ui->Cos_but, SIGNAL(clicked()), this, SLOT(operations()));
66       connect(ui->Log_but, SIGNAL(clicked()), this, SLOT(math_operations()));
67       connect(ui->Pow_but, SIGNAL(clicked()), this, SLOT(math_operations()));
68       connect(ui->Div_but, SIGNAL(clicked()), this, SLOT(math_operations()));
69       connect(ui->Mult_but, SIGNAL(clicked()), this, SLOT(math_operations()));
70
71       ui->Plus_but->setCheckable(true);
72       ui->Minus_but->setCheckable(true);
73       ui->Log_but->setCheckable(true);
74       ui->Pow_but->setCheckable(true);
75       ui->Mult_but->setCheckable(true);
76       ui->Div_but->setCheckable(true);
77
78   }
79
80   ~MainWindow()
81   {
82       delete ui;
83   }
84
85   void MainWindow::digitsNumbers(){
86       QPushButton *button = (QPushButton*)sender();
87
88       double allNums = 0;
89       QString num_text;
90
91       allNums = (ui->textEdit->toPlainText() + button->text()).toDouble();
92       num_text = QString::number(allNums, 'g', 15);
93       ui->textEdit->setText(num_text);
94   }
95
96   void MainWindow::on_dot_but_clicked(){
97       if(!ui->textEdit->toPlainText().contains('.'))
98           ui->textEdit->setText(ui->textEdit->toPlainText()+".");
99   }

```

Рисунок 15 - mainwindow.cpp

```

101 void MainWindow::operations(){
102     QPushButton *button = (QPushButton*)sender();
103
104     double allNums = 0;
105     QString num_str;
106
107     if(button->text() == "-/+"){
108         allNums = (ui->textEdit->toPlainText()).toDouble();
109         allNums *= -1;
110         num_str = QString::number(allNums, 'g', 15);
111         ui->textEdit->setText(num_str);
112
113     }else if (button->text() == "<-"){
114         allNums = (ui->textEdit->toPlainText()).toDouble();
115         num_str = QString::number(allNums, 'g', 15);
116         backspace(num_str);
117
118         ui->textEdit->setText(num_str);
119     }else if (button->text() == "CE"){
120         ui->textEdit->clear();
121     }else if (button->text() == "sin(x)"){
122         allNums = (ui->textEdit->toPlainText()).toDouble();
123         allNums = qSin(allNums);
124         num_str = QString::number(allNums, 'g', 15);
125         ui->textEdit->setText(num_str);
126     }else if (button->text() == "cos(x)"){
127         allNums = (ui->textEdit->toPlainText()).toDouble();
128         allNums = qCos(allNums);
129         num_str = QString::number(allNums, 'g', 15);
130         ui->textEdit->setText(num_str);
131     }
132 }
133
134 void MainWindow::math_operations(){
135     QPushButton *button = (QPushButton*)sender();
136
137     num_first = ui->textEdit->toPlainText().toDouble();
138     ui->textEdit->clear();
139     button->setChecked(true);
140 }
141
142 void MainWindow::on_ClearAll_but_clicked(){
143     num_first = 0;
144     ui->textEdit->clear();
145     ui->Plus_but->setChecked(false);
146     ui->Minus_but->setChecked(false);
147     ui->Log_but->setChecked(false);
148     ui->Pow_but->setChecked(false);
149     ui->Mult_but->setChecked(false);
150     ui->Div_but->setChecked(false);
151
152 }

```

Рисунок 16 - mainwindow.cpp

```

154 void MainWindow::on_Total_but_clicked(){
155     double textNum, num_second = 0;
156     QString new_Num;
157
158     num_second = ui->textEdit->toPlainText().toDouble();
159
160     if (ui->Plus_but->isChecked()){
161         textNum = num_second + num_first;
162         new_Num = QString::number(textNum, 'g', 15);
163         ui->textEdit->setText(new_Num);
164         ui->Plus_but->setChecked(false);
165
166     }else if (ui->Minus_but->isChecked()){
167         textNum = num_first - num_second;
168         new_Num = QString::number(textNum, 'g', 15);
169         ui->textEdit->setText(new_Num);
170         ui->Minus_but->setChecked(false);
171
172     }else if (ui->Log_but->isChecked()){
173
174         if (num_second > 0 && num_second != 1){
175             textNum = calculateLog(num_second, num_first);
176             new_Num = QString::number(textNum, 'g', 15);
177             ui->textEdit->setText(new_Num);
178             ui->Log_but->setChecked(false);
179         }else
180             ui->textEdit->setText("ERROR");
181
182     }else if (ui->Pow_but->isChecked()){
183         textNum = calculatePower(num_second, num_first);
184         new_Num = QString::number(textNum, 'g', 15);
185         ui->textEdit->setText(new_Num);
186         ui->Pow_but->setChecked(false);
187
188     }else if (ui->Div_but->isChecked()){
189
190         if (num_second != 0){
191             textNum = num_first/num_second;
192             new_Num = QString::number(textNum, 'g', 15);
193             ui->textEdit->setText(new_Num);
194             ui->Div_but->setChecked(false);
195         }else
196             ui->textEdit->setText("ERROR");
197     }else if (ui->Mult_but->isChecked()){
198         textNum = num_second * num_first;
199         new_Num = QString::number(textNum, 'g', 15);
200         ui->textEdit->setText(new_Num);
201         ui->Mult_but->setChecked(false);
202     }
203 }

```

Рисунок 17- mainwindow.cpp

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5
6  QT_BEGIN_NAMESPACE
7  namespace Ui { class MainWindow; }
8  QT_END_NAMESPACE
9
10 class MainWindow : public QMainWindow
11 {
12     Q_OBJECT
13
14 public:
15     MainWindow(QWidget *parent = nullptr);
16     ~MainWindow();
17
18 private:
19     Ui::MainWindow *ui;
20 private slots:
21     void digitsNumbers();
22     void on_dot_but_clicked();
23     void operations();
24     void math_operations();
25
26     void on_ClearAll_but_clicked();
27     void on_Total_but_clicked();
28 };
29 #endif // MAINWINDOW_H

```

Рисунок 18 - mainwindow.h

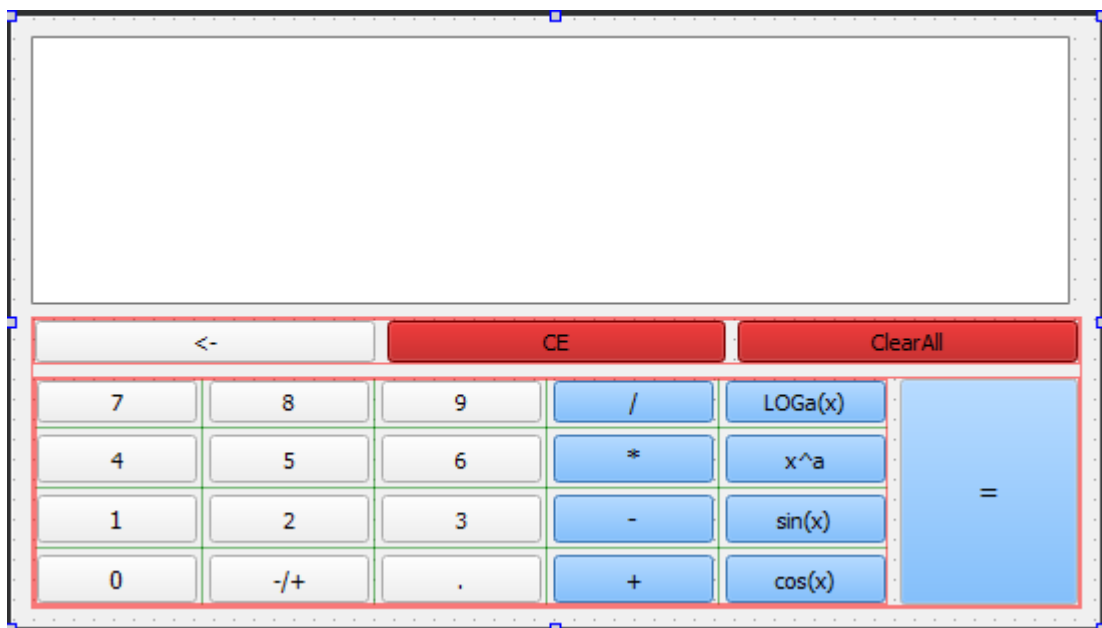


Рисунок 19 – интерфейс

Тестовые данные

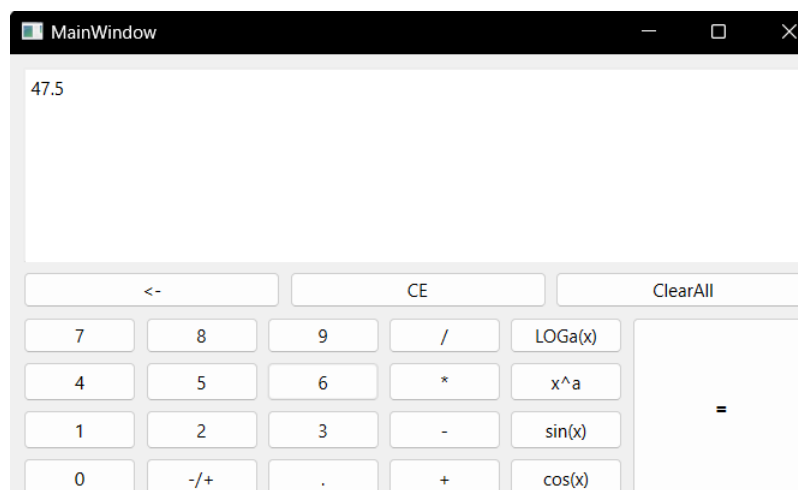


Рисунок 20 - пример работы программы

Часть 4

Некоторые средства для ввода и вывода текста Класс QString Для работы со строками в Qt используется класс QString. Основной особенностью этого класса является то, что внутреннее хранение и все операции над строками проводятся в кодировке UNICODE. Класс позволяет преобразовывать текст из различных кодировок строки в формат C и обратно. Реализуется операции склейки, добавления, сравнения, вырезания подстроки и пр.

Задание

Разработать приложение, имеющее строку ввода данных, кнопку запуска преобразования и текстовое поле, предназначенное только для отображения информации. При этом не использовать QtDesigner! Любой текст строки ввода должен отображаться в текстовом поле сразу после завершения ввода. В начале строки должна быть вставлена пометка «input:». При нажатии кнопки преобразования строка ввода должна быть преобразована либо в верхний регистр, либо в нижний противоположно тому, что производилось при предыдущем нажатии кнопки.

Текст программы

```
1  #include "mainwindow.h"
2
3  #include <QApplication>
4  #include <QLocale>
5  #include <QTranslator>
6
7  int main(int argc, char *argv[])
8  {
9      QApplication a(argc, argv);
10
11     QTranslator translator;
12     const QStringList uiLanguages = QLocale::system().uiLanguages();
13     for (const QString &locale : uiLanguages) {
14         const QString baseName = "t4_" + QLocale(locale).name();
15         if (translator.load(":/i18n/" + baseName)) {
16             a.installTranslator(&translator);
17             break;
18         }
19     }
20     MainWindow w;
21     w.show();
22     return a.exec();
23 }
```

Рисунок 21 - main.cpp

```

1  #include "mainwindow.h"
2  #include "./ui_mainwindow.h"
3  short num = 0;
4  QString this_word = "";
5
6  MainWindow::MainWindow(QWidget *parent)
7      : QMainWindow(parent)
8      , ui(new Ui::MainWindow)
9  {
10     ui->setupUi(this);
11 }
12
13 MainWindow::~MainWindow()
14 {
15     delete ui;
16 }
17
18
19 void MainWindow::on_pushButton_clicked(){
20     QString str;
21     str = ui->input->toPlainText();
22     if (str != this_word){
23         this_word = str;
24         num = 0;
25     }
26     if (num == 0){
27         ui->output->setText("Исходный: " + str);
28     }
29     num++;
30     if (num == 1){
31         ui->output->append("Строчные: " + str.toLowerCase());
32     }else if (num == 2)
33         ui->output->append("Заглавные: " + str.toUpperCase());
34 }
35

```

Рисунок 22 - mainwindow.cpp

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5
6  QT_BEGIN_NAMESPACE
7  namespace Ui { class MainWindow; }
8  QT_END_NAMESPACE
9
10 class MainWindow : public QMainWindow
11 {
12     Q_OBJECT
13
14     public:
15         MainWindow(QWidget *parent = nullptr);
16         ~MainWindow();
17
18     private slots:
19         void on_pushButton_clicked();
20
21     private:
22         Ui::MainWindow *ui;
23 };
24 #endif // MAINWINDOW_H

```

Рисунок 23 - mainwindow.h

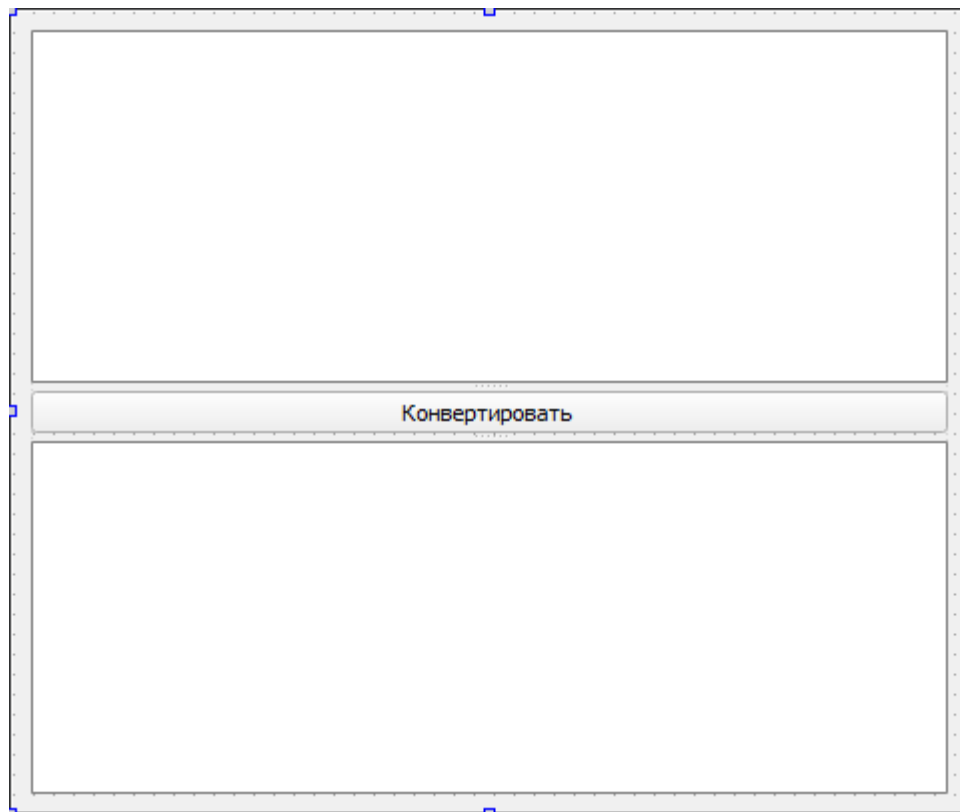


Рисунок 24 – интерфейс

Тестовые данные

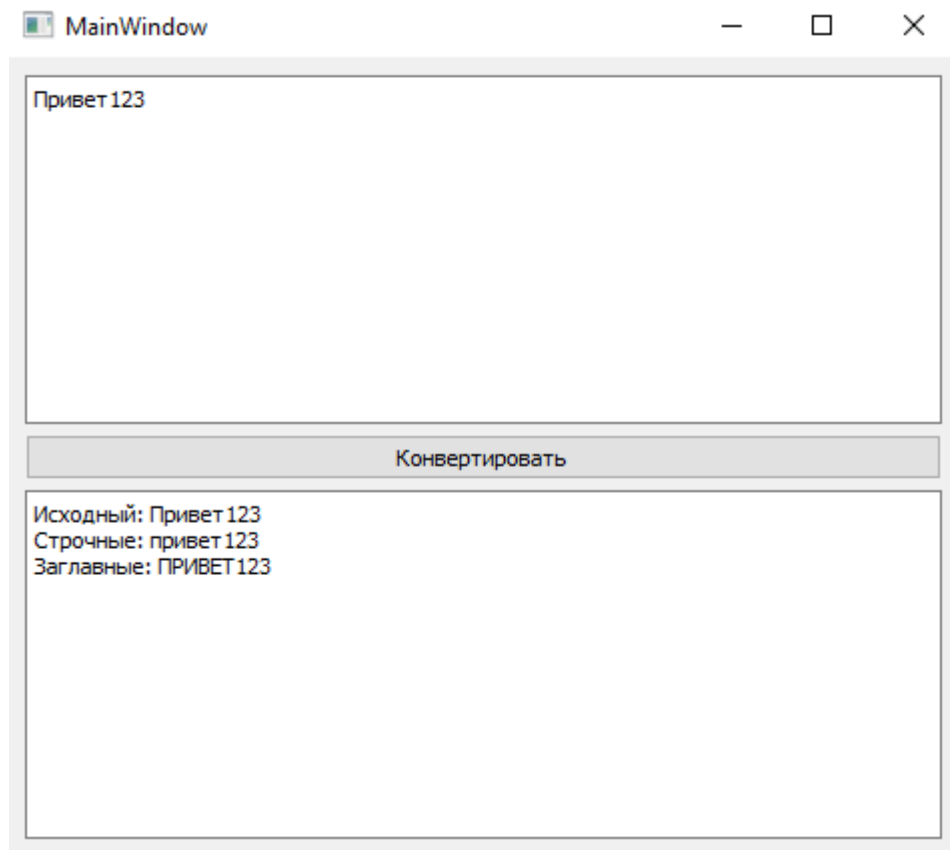


Рисунок 25 - пример работы

Вывод

Были изучены основы работы с библиотекой QT на языке программирования C++. Получены знания о работе с основными средствами построения приложений с графическим интерфейсом.