|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 9 |

**Название:** Программирование с использованием библиотеки QT

**Дисциплина:** Объектно-ориентированное программирование

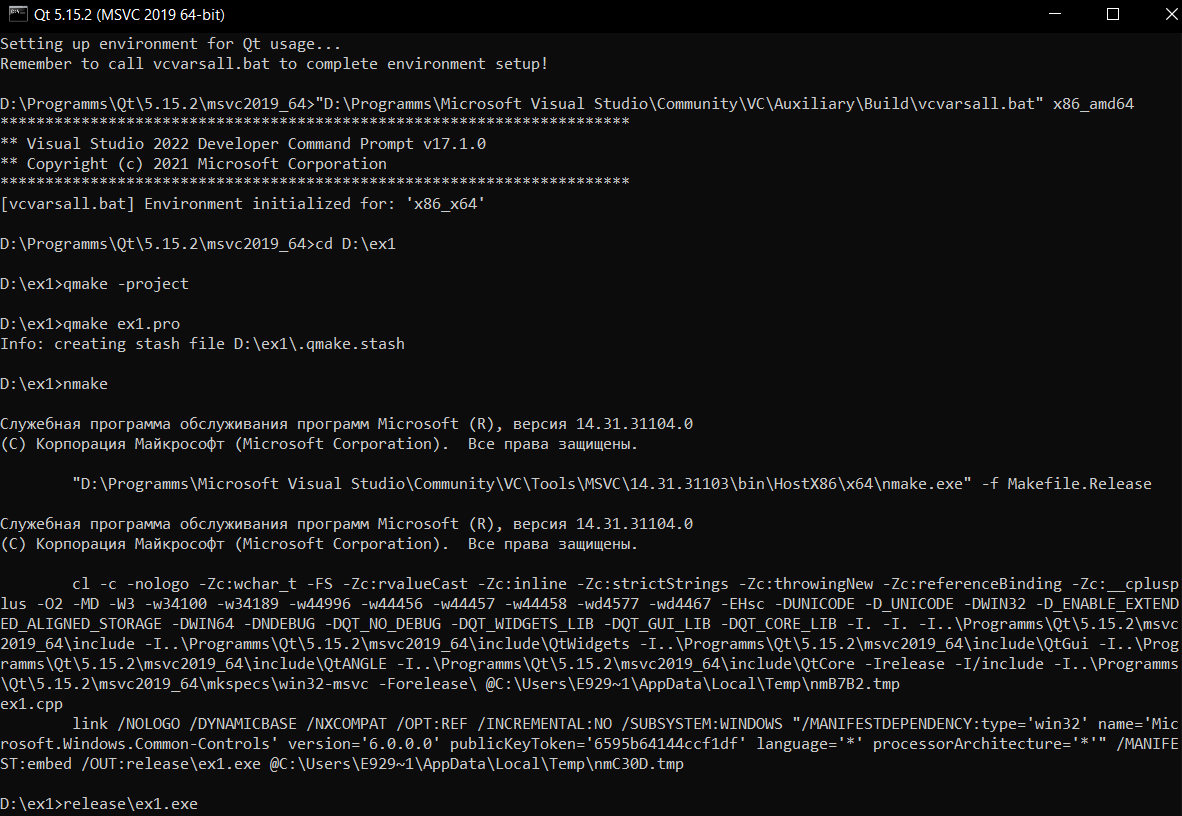
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-22Б |  |  | Д.С.Твердюк |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | А.А. Веселовский |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

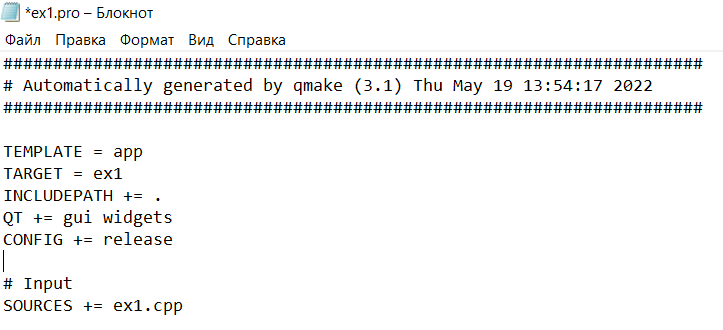
**Ход работы:**

**Задание 1.**

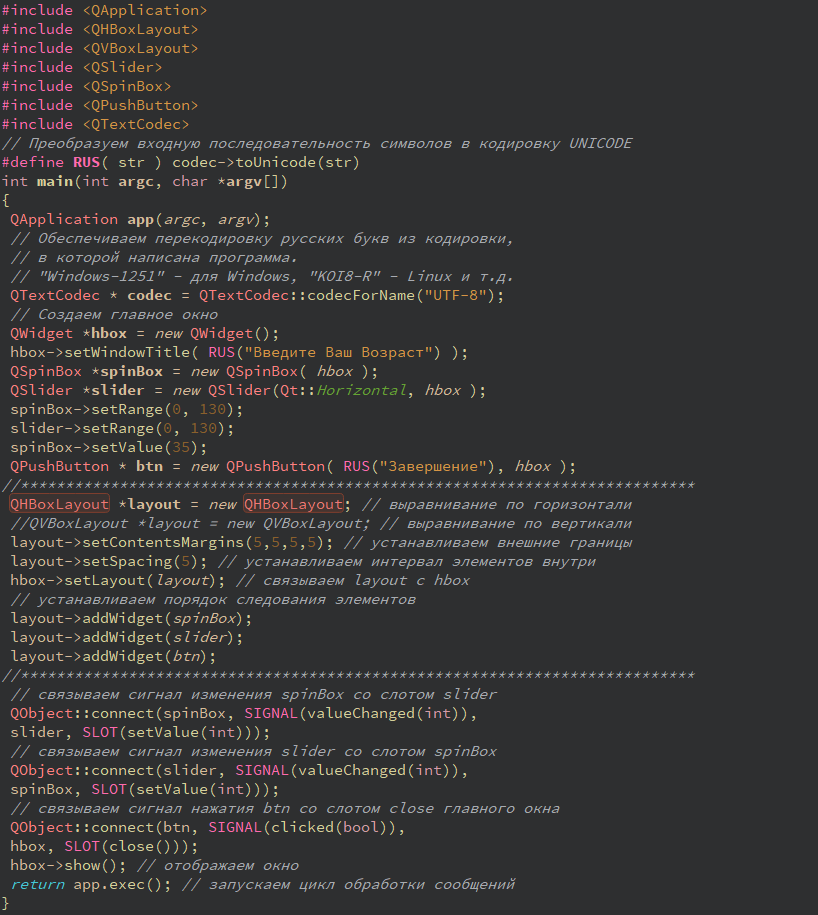
Сборка проекта:

****

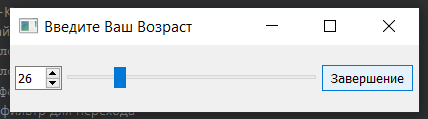
Файл ex1.pro:



Код основной программы:



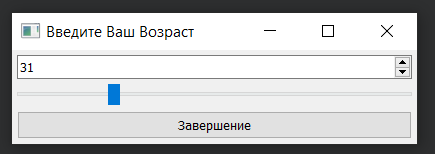
Результат работы написанной программы:



Замена схема выравнивания QHBoxLayout на QVBoxLayout:

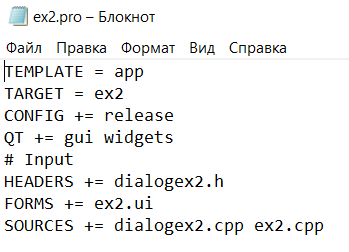


Результат работы программы после изменения:

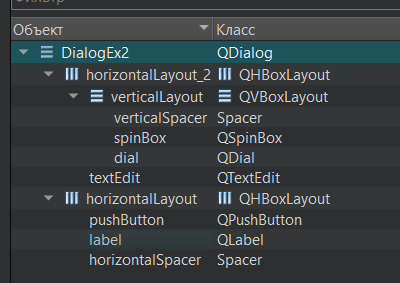
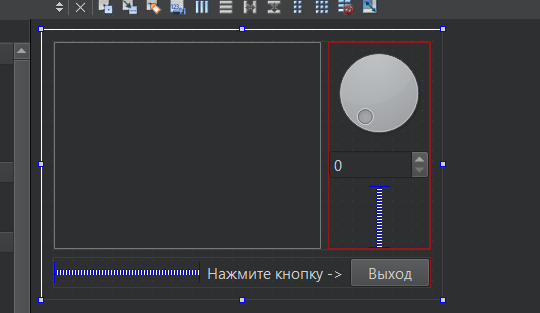


**Задание 2.**

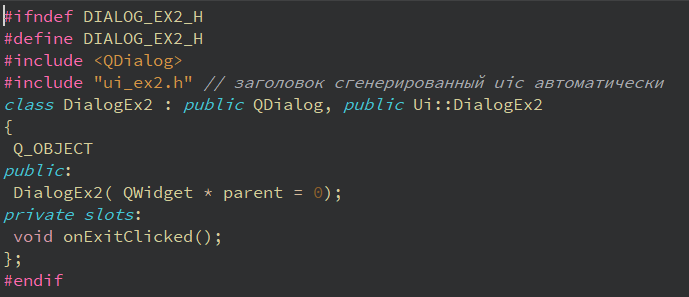
Файл ex2.pro:



Форма ex2.ui:



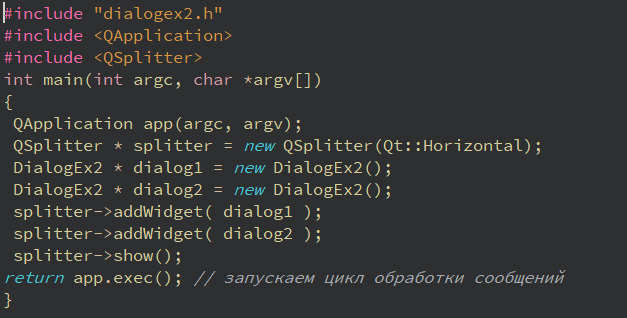
Заголовочный файл dialogex2.h:



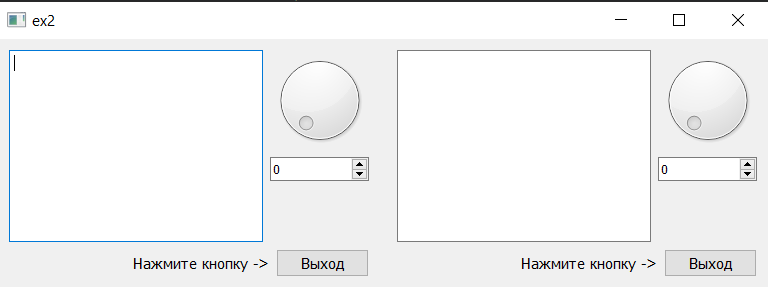
Файл реализации dialogex2.cpp:



Код основной программы:



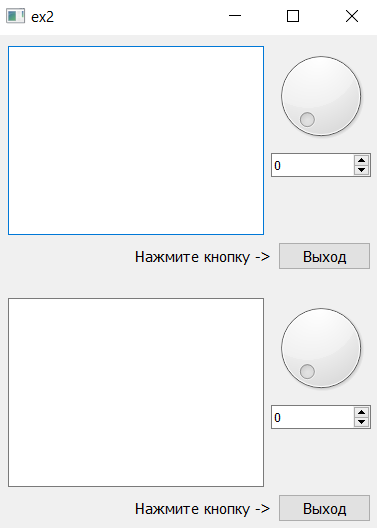
Результат работы написанной программы:



Изменение типа разделителя с QSplitter(Qt::Horizontal); на QSplitter(Qt::Vertical); :

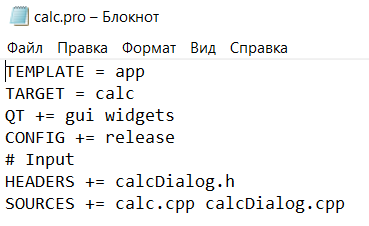


Результат работы программы после изменение типа разделителя:

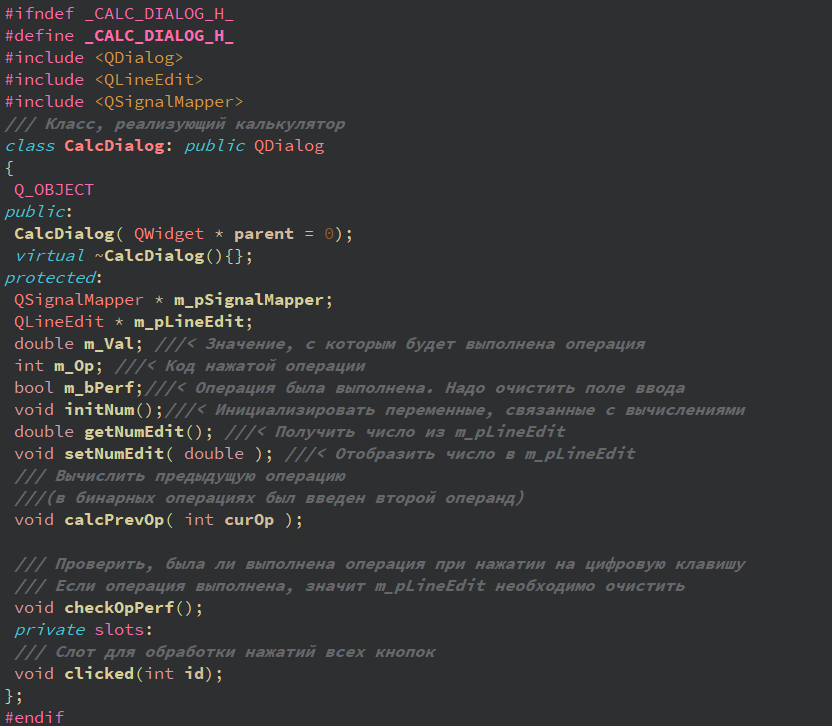


**Задание 3.**

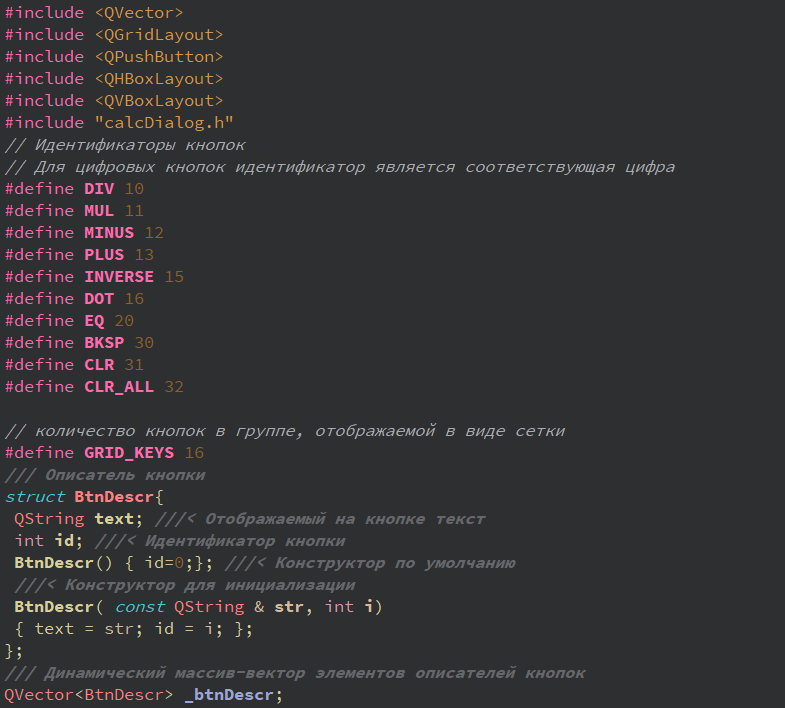
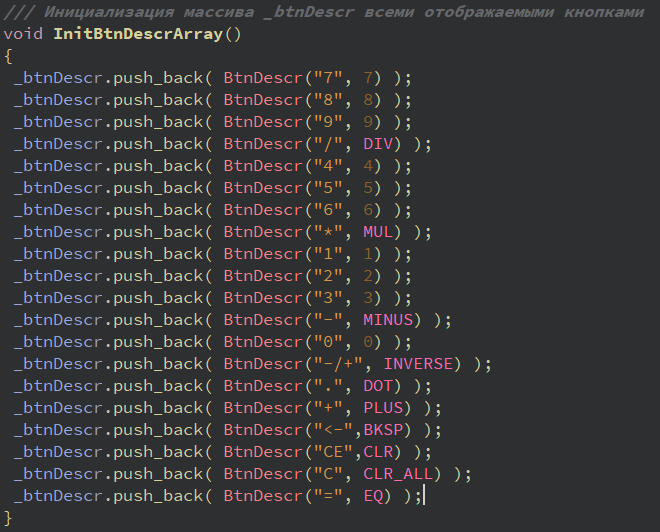
Файл calc.pro:

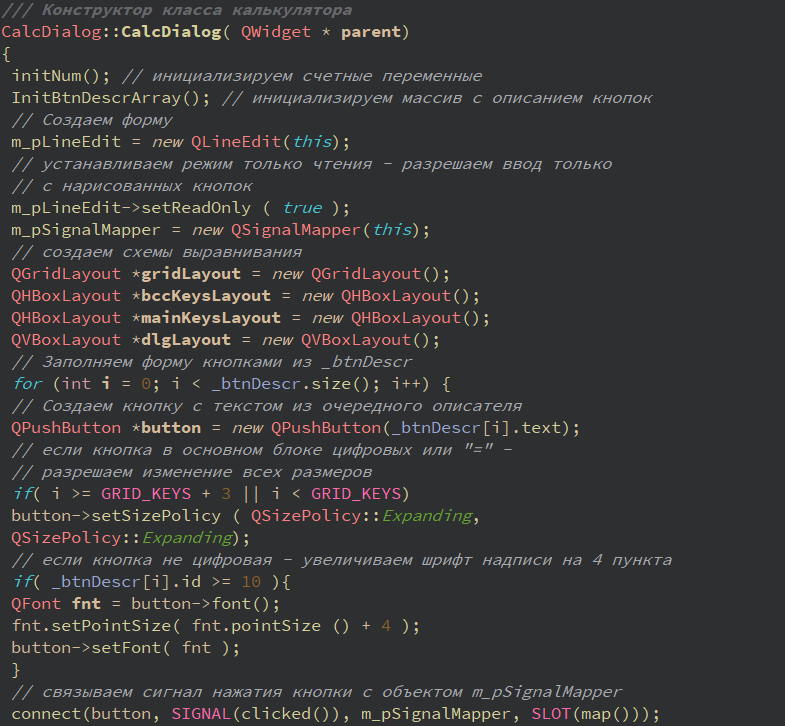
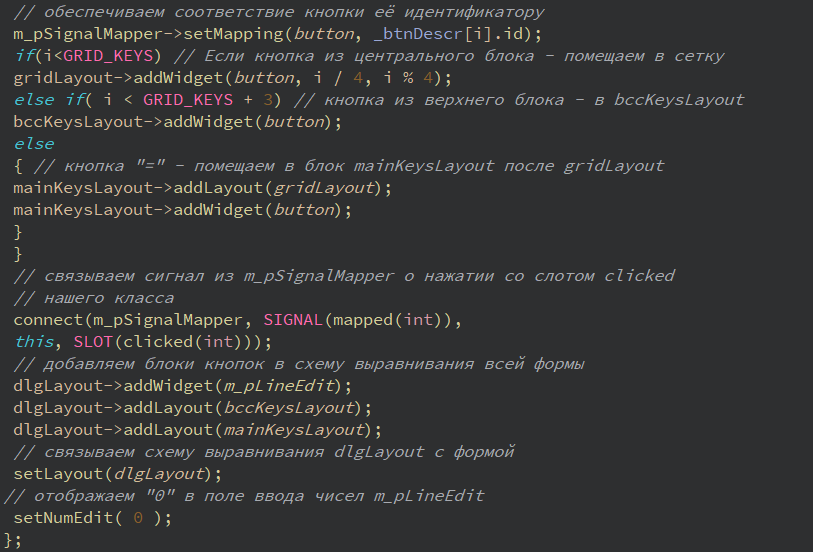


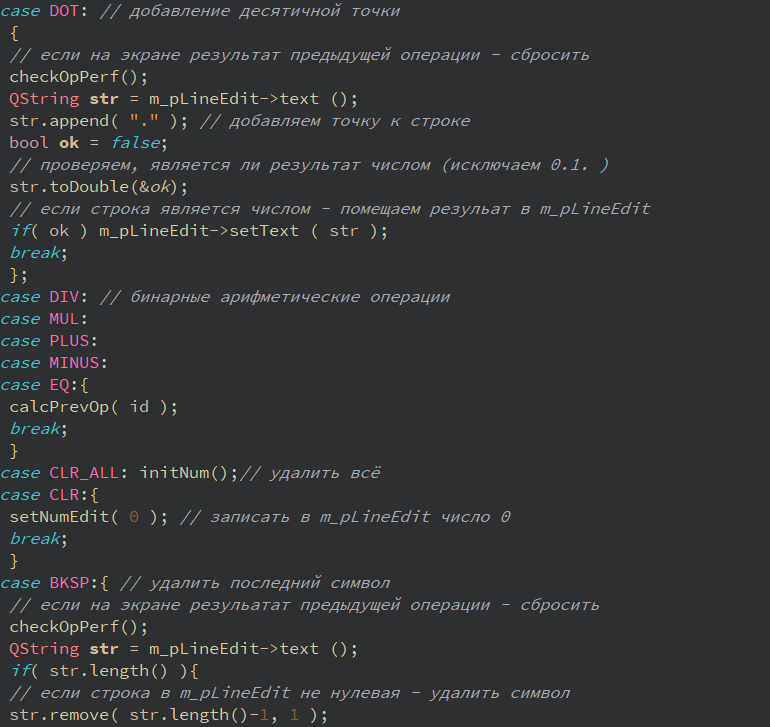
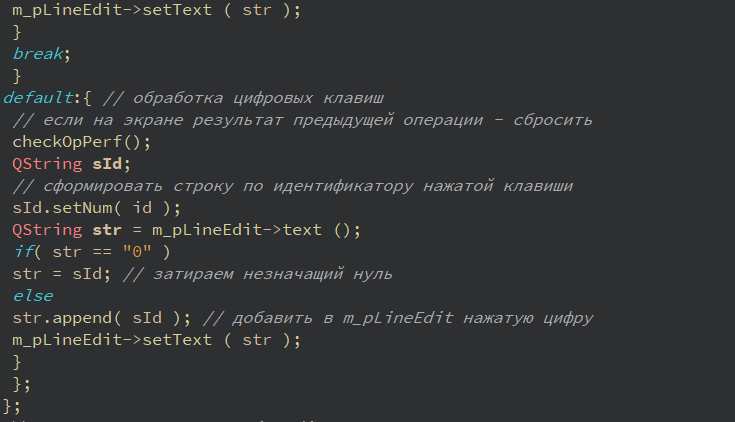
Заголовочный файл calcDialog.h:

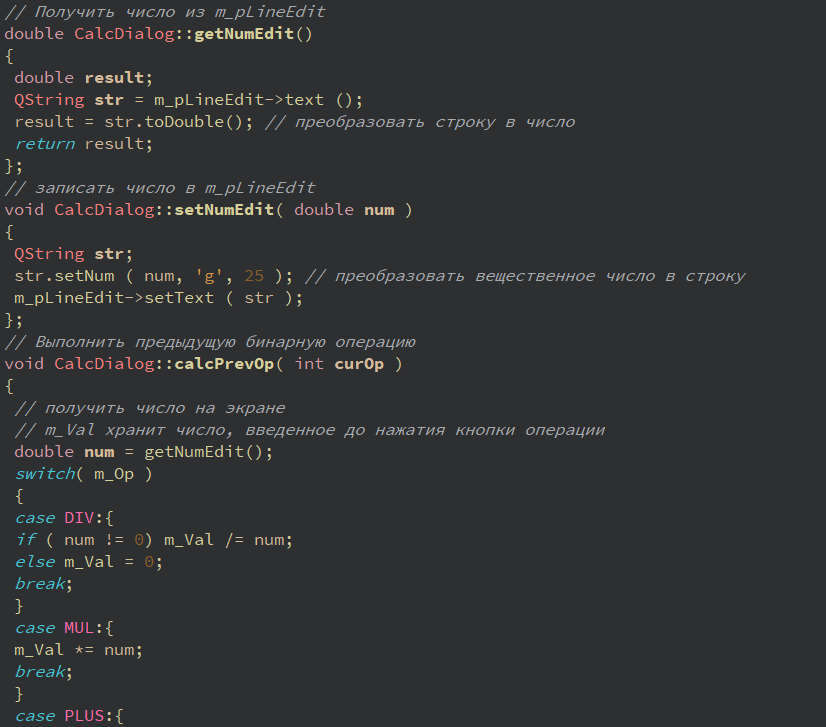
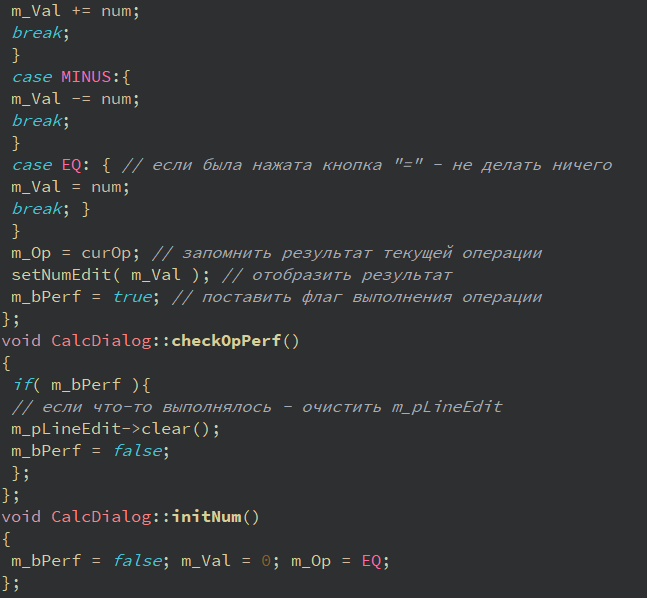


Файл реализации calcDialog.cpp:

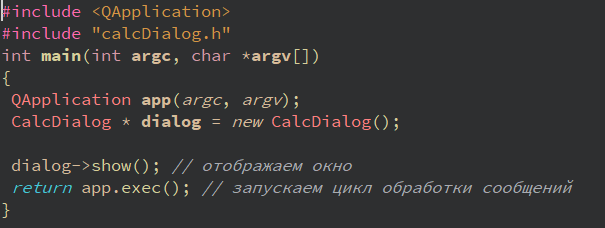




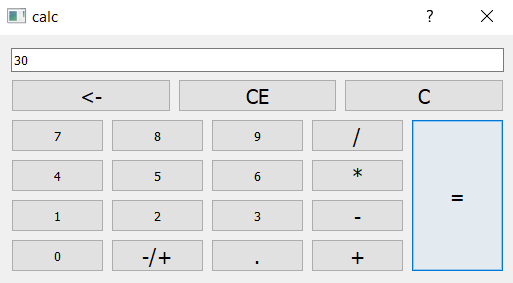




Код основной программы:



Результаты работы написанной программы:



Добавление новых операций, соответствующих кнопок и нового объекта выравнивания:

…

#define **DOT** 16

#define **EXP** 17

#define **LOG** 18

#define **SIN** 19

#define **COS** 20

#define **EQ** 21

…

void **InitBtnDescrArray**()

{

…

\_btnDescr.push\_back( BtnDescr("x^y", EXP) );

\_btnDescr.push\_back( BtnDescr("Log(y)x", LOG) );

\_btnDescr.push\_back( BtnDescr("sin x", SIN) );

\_btnDescr.push\_back( BtnDescr("cos x", COS) );

…

}

CalcDialog::**CalcDialog**( QWidget \* **parent**)

{

…

QVBoxLayout \***Maksonishe** = new QVBoxLayout();

…

if(i<GRID\_KEYS - 4) // Если кнопка из центрального блока - помещаем в сетку

gridLayout->addWidget(button, i / 4, i % 4);

else if (i < GRID\_KEYS)

Maksonishe->addWidget(button);

else if( i < GRID\_KEYS + 3) // кнопка из верхнего блока - в bccKeysLayout

bccKeysLayout->addWidget(button);

else {

// кнопка "=" - помещаем в блок mainKeysLayout после gridLayout

mainKeysLayout->addLayout(Maksonishe);

mainKeysLayout->addLayout(gridLayout);

mainKeysLayout->addWidget(button);

}

}

…

void CalcDialog::**clicked**(int **id**)

{

…

case SIN:

{

setNumEdit( sin(getNumEdit())); break;

};

case COS:

{

setNumEdit( cos(getNumEdit())); break;

};

…

case EXP:

case LOG:

case EQ:{

calcPrevOp( id );

break;

}

…

}

void CalcDialog::**calcPrevOp**( int **curOp** )

{

…

case EXP:{

m\_Val = pow(m\_Val, num);

break;

}

case LOG:{

m\_Val = log10(m\_Val)/log10(num);

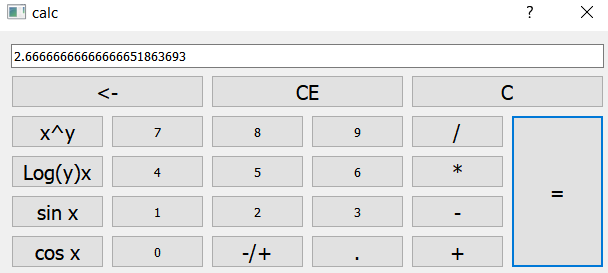
break;

}

…

}

Работа программы после изменения:



Тестирование написанной программы:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Введенные данные** | **Ожидаемый результат** | **Полученный результат** |
| *1* | 15/3 | 5 | 3 |
| *2* | 43 | 64 | 64 |
| *3* | log101000 | 3 | 3 |

**1**



**2**



3

**Задание 4.**

Файл task4.pro:

TEMPLATE = app

TARGET = Task4

QT += gui widgets

CONFIG += release

# Input

HEADERS += task4Win.h

SOURCES += task4.cpp task4Win.cpp

Файл task4Win.h:

#ifndef \_task\_4\_h\_

#define **\_task\_4\_h\_**

#include <QDialog>

#include <QLineEdit>

#include <QPushButton>

#include <QTextEdit>

enum **reg** {**UP**, **DOWN**};

class **MainWindow** : public QDialog

{

Q\_OBJECT

public:

**MainWindow**(QWidget \* **parrent** = 0);

virtual ~**MainWindow**() {};

protected:

QLineEdit \* **inputLine**;

QPushButton \***convertBtn**;

QTextEdit \***output**;

reg **regtype**;

private slots:

void **write**();

void **convert**();

};

#endif

Файл task4Win.cpp:

#include <QVBoxLayout>

#include "task4Win.h"

MainWindow::**MainWindow**(QWidget \* **parrent**){

QVBoxLayout \***mainLayout** = new QVBoxLayout();

inputLine = new QLineEdit();

mainLayout->addWidget(inputLine);

convertBtn = new QPushButton("Convert");

convertBtn->setFocusPolicy(Qt::NoFocus);

mainLayout->addWidget(convertBtn);

output = new QTextEdit();

output->setReadOnly(true);

mainLayout->addWidget(output);

setLayout(mainLayout);

connect(inputLine, SIGNAL(editingFinished()), this, SLOT(write()));

connect(convertBtn, SIGNAL(clicked()), this, SLOT(convert()));

regtype = UP;

}

void MainWindow::**write**() {

output->append("Input: " + inputLine->text());

}

void MainWindow::**convert**() {

switch (regtype)

{

case UP:

{

output->append("ALL UPPER: " + inputLine->text().toUpper());

regtype = DOWN;

break;

}

case DOWN:

{

output->append("all lower: " + inputLine->text().toLower());

regtype = UP;

}

}

}

Код основной программы:

#include <QApplication>

#include "task4Win.h"

int **main**(int **argc**, char \***argv**[]){

QApplication **app**(argc, argv);

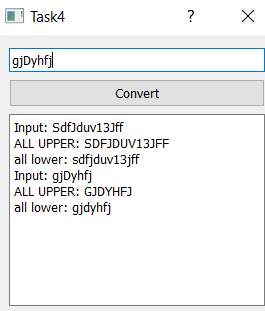
MainWindow \***mw** = new MainWindow();

mw->show();

return app.exec();

}

Результат работы написанной программы:



**Вывод:** программы работаю корректно на заданных исходных данных. Я научился создавать приложения с помощью QTCreator.