

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе
по курсу «Программирование»
на тему «Инженерный калькулятор»

Выполнила:

студентка группы 21ВВ4
Нагорная Д. А.

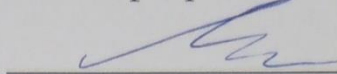
Приняли:

к.т.н., доцент Слепцов Н. В.
Патунин Д. В.

10.06.22
Патунин отл.

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ



М.А. Митрохин

« ____ » ____ 20__ г

ЗАДАНИЕ

на курсовое проектирование по курсу

Студенту Нагорной Дарье Александровне Группа 21ВВ4

Тема проекта Инженерный калькулятор

Исходные данные (технические требования) на проектирование

1. Графический интерфейс.
2. Максимальная разрядность числа - 32 бита.
3. Представление десятичного числа в других системах счисления (двоичной, восьмеричной и шестнадцатеричной).
4. Набор стандартных арифметических функций.
5. Набор тригонометрических функций.
6. Калькулятор должен содержать такие функции, как обратная величина числа, логарифмы десятичный, возведение числа в степень.
7. Результат должен быть не конечным.
8. Возможность корректировки ввода (сброс, сброс одной позиции).

Функциональные требования к программе:

Многофункциональность; использование сложных типов данных; режим работы видеосистемы - текстовый / графический; устройство ввода информации - клавиатура, мышь; пользовательский интерфейс должен быть построен на основе меню и панели инструментов; наличие заставки; операционная система MS Windows; язык программирования Си и Ассемблер; среда разработки ПО - Microsoft Visual Studio.

Объем работы по курсу

1. Расчетная часть

Разработка программы.

2. Графическая часть

Схема данных, схема ресурсов системы, схема работы системы, иерархическая структура программы, схема взаимодействия программы.

3. Экспериментальная часть

Отладка программы.

Срок выполнения проекта по разделам

В соответствии с графиком.

Дата выдачи задания "04" март 2022 г.

Дата защиты проекта "10" 06 22

Руководитель Горюхов Д. В.

Задание получил "4" марта 2022 г.

Студент Нагорная Варья Александровна

Содержание

Введение	5
1 Постановка задачи	6
2 Выбор решения	7
2.1 Определение необходимых модулей программы	8
2.2 Определение алгоритма программы	9
3 Описание разработки программы	12
4 Отладка и тестирование	14
5 Описание программы	15
6 Руководство пользователя	19
Заключение	22
Список используемых источников	23
Приложение А Листинг программы	24
Исходный файл «ordinary.cpp»	24
Заголовочный файл «ordinary.h»	25
Исходный файл «engineering.cpp»	46
Заголовочный файл «engineering.h»	47
Исходный файл «programmatic.cpp»	71
Заголовочный файл «programmatic.h»	71
Приложение В Результаты работы программы (снимки экрана)	93
Заголовочный файл «ordinary.h [Конструктор]»	93
Заголовочный файл «engineering.h[Конструктор]»	94
Заголовочный файл «programmatic.h[Конструктор]»	95

Введение

Визуальный язык программирования – это язык взаимодействия пользователя с системой программирования, реализуемой диалоговыми средствами графического интерфейса пользователя. Появление визуальных языков была следствием сложности программирования графического интерфейса программ, предназначенных для выполнения в операционных средах типа Windows. Чтобы облегчить программирование элементов графического интерфейса как раз и используются визуальные языки.

Графический интерфейс пользователя (Graphical User Interface, GUI) делает возможным использование графики на растровом экране. Графика дает лучшее восприятие элементов управления на экране, визуальную богатую среду для передачи информации. В GUI экран становится устройством ввода и показывает различные графические объекты в виде картинок и конструкций для ввода информации, таких как кнопки или полосы прокрутки. Используя клавиатуру и манипулятор (мышь, touchPad), пользователь может непосредственно оперировать объектами на экране. Взаимодействие между пользователем и программой становится более тесным.

Любая программа для Windows имеет окно — прямоугольную область на экране, в котором приложение отображает информацию и получает реакцию от пользователя. Окно идентифицируется заголовком. Пользователям не надо тратить слишком много времени на то, чтобы научиться пользоваться компьютером и составлять новые программы. Система Windows способствует этому, поскольку все программы для Windows выглядят и воспринимаются одинаково.

Универсальным техническим средством обработки любой информации является компьютер и разработанные для конкретных целей программные продукты. К таким продуктам обработки информации относятся различные информационные системы, позволяющие систематизировать и обрабатывать огромные потоки информации. Для персональных компьютеров было разработано множество программ, предназначенных для решения различных потребностей человека. Данная программа является информационно-вычислительной системой, позволяющей любому человеку считать на калькуляторе.

Калькулятор – это устройство для арифметических вычислений. Целью курсовой работы является углубление знаний и расширение навыков по разработке алгоритмов и их реализации на персональном компьютере на примере создания калькулятора на языке C, используя ассемблерную вставку, для выполнения математических вычислений.

1 Постановка задачи

Необходимо разработать программу «Инженерный калькулятор». Проанализировав существенные программные решения, были определены следующие функциональные требования к разрабатываемой программе.

Многомодульность программы. Необходимо поделить программу на логические модули. Это повысит расширяемость программы.

Использование сложных типов данных – классов и объектов. Эти составные типы данных являются основными концепциями объектно-ориентированного программирования (ООП).

Программа должна иметь простой, но в то же время понятный и наглядный графический интерфейс, который не должен перегружать ресурсы компьютера. Наличие графического интерфейса позволяет осуществлять взаимодействие человека с компьютером в форме диалога с использованием окон, и элементов управления.

Устройство ввода-вывода – клавиатура и мышь. Управлять калькулятором можно как с помощью мыши, так и посредством клавиатуры. Ввод чисел с клавиатуры удобнее осуществлять в том случае, когда необходимо произвести большое количество вычислений. Необходимо научиться анализировать и различать события, возникающие от работы с клавиатурой и мышью.

Пользовательский интерфейс должен быть построен на основе меню или панели инструментов.

Модификация и сопровождение (или соблюдение правильности вычислений). Это этап эксплуатации программы. Каким бы ни было изощренным тестирование программы, к сожалению, в больших программных комплексах чрезвычайно тяжело устранить абсолютно все ошибки. Устранение обнаруженных при эксплуатации ошибок, неудачных проектных решений, «узких мест» - одна из основных задач.

При разработке программы необходимо использовать среду MSWindows с системой программирования MSVisualStudio.

Язык программирования: С и Ассемблер. Язык программирования С имеет возможность применения классов при использовании обычного стиля программирования и поддерживает визуальное программирование с немедленным отображением графических объектов.

2 Выбор решения

2.1 Определение необходимых модулей программы

Технических требований на проектирование достаточно много, поэтому будет сложно сразу увидеть все кнопки с определенной функцией на одном оконном приложении. Для решения этой проблемы были реализованы три графических интерфейса: первый представляет собой обычный калькулятор со стандартными арифметическими функциями, а также содержит функцию возведения числа в квадрат, извлечения корня, обратной величины числа и подсчета процента числа; второй включает в себя возведение числа в любую степень, подсчет логарифмов, константы (число «Пи» и число Эйлера) и расчет основных тригонометрических функций; третий имеет функцию перевода целого знакового десятичного числа в двоичную, восьмеричную и шестнадцатеричную систему счисления.

Поэтому, разработанная программа содержит 6 модулей (три заголовочных файла и три исходных файла):

```
ordinary.cpp
ordinary.h
engineering.cpp
engineering.h
programmatic.cpp
programmatic.h
```

Графический интерфейс ordinary.h (обычный калькулятор) при запуске программы должен открываться первым, поэтому является ведущим модулем. Для этого в файле ordinary.cpp реализован класс, который позволяет сделать данный интерфейс видимым при запуске:

```
#include "ordinary.h"

using namespace System;
using namespace System::Windows::Forms;

[STAThread]
void Main(cli::array<String^>^ args)
{
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);

    CWCalculator::ordinary form;
    Application::Run(% form);
    //перезагрузка для запуска основной формой
}
```

На рисунке представлена взаимосвязь модулей программы:



Рисунок 1. Структура модулей программы.

Управление программой осуществляется с помощью клавиатуры и мыши. При запуске программы сразу выводится калькулятор, который уже готов к работе.

Одновременная работа двух или трех графических интерфейсов (нескольких режимов калькуляторов) не допускается, чтобы избежать медленной работы программы. Но данные калькулятора при переходе на другой графический интерфейс сохраняются.

2.2 Определение алгоритма программы

Для представления принципа работы программы ниже приведена ее функциональная схема (рис.2), которая предназначена только для стандартных арифметических операций, подсчета процента, возведения числа в степень и, вероятно, использование констант (число «Пи» и число Эйлера):



Рисунок 2. Функциональная схема программы.

Пункт «Ввод первого числа» представляет собой часть программы, в которой выбирается нужное число для подсчета. После выбора числа необходимо выбрать действие, которое необходимо выполнить (пункт «Выбор функции»). Далее вводим второе значение (пункт «Ввод второго числа»). Затем по введенным данным вычисляется результат (пункт «Вычисление значения функции»).

Далее пользователю предоставляется продолжить работу с результатом (пункт «Работа с результатом»). В следствии работы с ним пользователь может выйти из программы или сбросить результат и продолжить вычисления.

Для вычисления нестандартных функций будет использоваться другой метод (рис. 3).



Рисунок 3. Функциональная схема программы.

Данный алгоритм будет использован для тригонометрических функций, а также при таких функциях, как возведение числа в квадрат, извлечение корня, нахождение обратной величины числа и подсчет логарифмов.

Как уже было выявлено, в интерфейсе ordinary.h содержатся кнопки перехода из одной формы в другие - engineering.h (инженерный калькулятор) и programmatic.h (калькулятор с представлением числа в другие системы счисления).

Переключаться с использования одного варианта калькулятора на другой можно непосредственно в процессе работы, а данные при этом должны сохраняться. Для перемещения в начальную форму в других интерфейсах также имеется кнопка.

Ниже на рисунке 4 представлена схема навигации (перемещения) из одного интерфейса в другой.

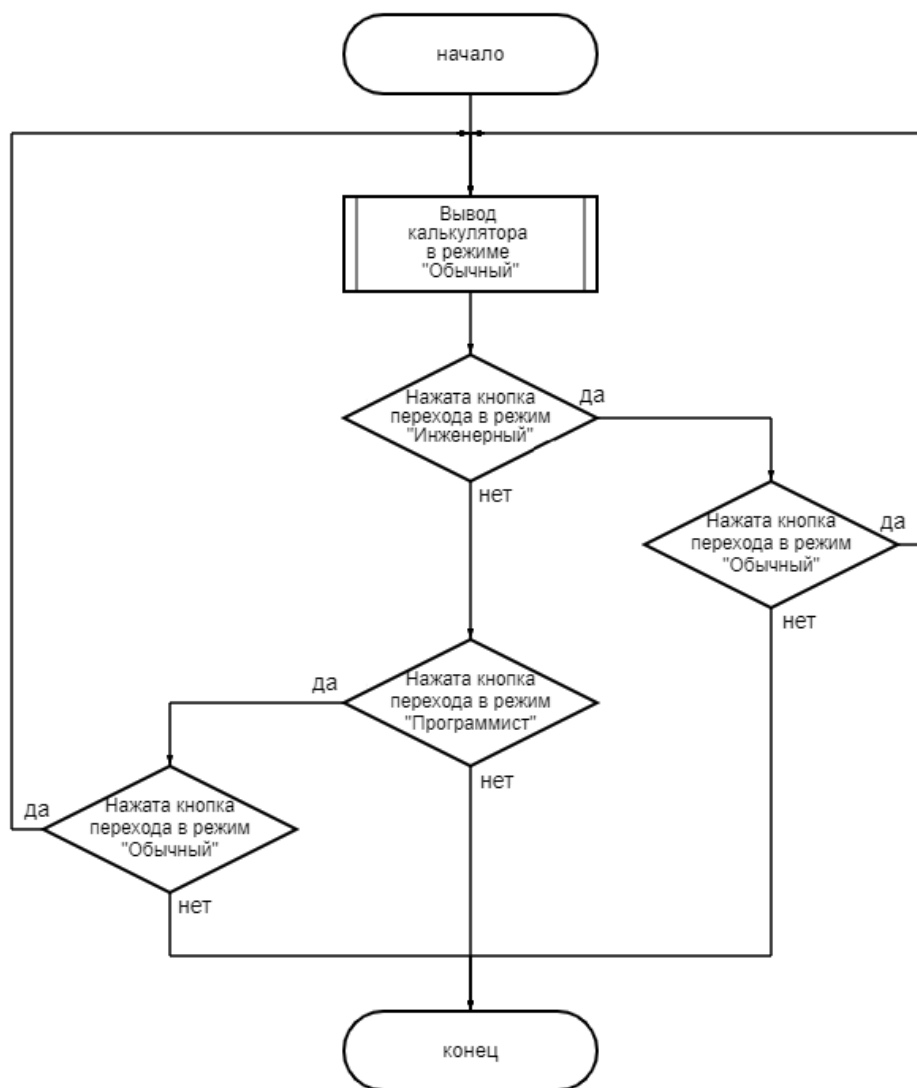


Рисунок 4. Схема навигации программы.

3 Описание разработки программы

Для написания данной программы будет использован язык программирования Си. Этот язык является распространенным языком программирования. При разработке языка Си был принят компромисс между низким уровнем языка ассемблера и высоким уровнем других языков. В языке Си предусмотрено много операций, непосредственно исполняемых большинством микропроцессоров; в то же время он дает программисту возможность выражать свои алгоритмы и данные наиболее подходящими средствами. Си обеспечивает возможности структурирования данных. Он позволяет разрабатывать большие, но структурированные программы, предоставляя возможность отдельной разработки программы.

Целью создателей языка Си была разработка мобильного языка, который можно было бы использовать для разработки системного программного обеспечения. Программы, написанные на языке высокого уровня, мобильны, поскольку язык должен быть одним и тем же независимо от того, на каком компьютере и в какой операционной системе он используется. Дополнительное свойство языка Си, называемое условной компиляцией, позволяет программисту изолировать машинно-зависимые операторы и контролировать их компиляцию в другой среде. Это дополнительно повышает мобильность программного обеспечения, написанного на языке Си.

В качестве среды программирования был выбран программный продукт VisualStudio 2019.

Компонентный подход к созданию приложений (или визуальное программирование), реализованный в различных средах разработки приложений, произвел настоящую революцию в технологиях программирования. Суть компонентного подхода проста: приложение собирается из готовых элементов – компонентов, библиотека которых поставляется вместе со средой программирования. В качестве строительной площадки служит форма – окно Windows, которое во время разработки обладает рядом вспомогательных свойств, облегчающих размещение компонентов.

Интерфейс информационно-вычислительной системы будет представлен по большей части в виде панели инструментов, т.к. это наиболее удобный доступ к данным. Перемещение по пунктам меню будет осуществляться с помощью кнопок на графическом интерфейсе.

Для создания программы «Калькулятор» реализован алгоритм, позволяющий иметь возможность при выборе действия (операции) выводить на экран полученный результат вычислений. Для анализа программы необходимо выявить ее поведение (табл.1):

Описание	Ожидаемый результат
Запуск программы	Задание верных размеров графического интерфейса, который в процессе работы можно изменять.
Ввод числа с клавиатуры	Максимальная разрядность числа не превышает 32 бит. В случае переполнения кнопки цифр будут заблокированы.
Работа с арифметическими операциями	Выражение должно быть рассчитано не в одну стандартную арифметическую операцию.
Представление чисел в системах счисления	Перевод отрицательного числа в дополнительный код.
Разработка исключений	Деление на ноль – невозможно. Вывод большого длинного числа должен быть представлен в экспоненциальной форме. Изменение знака в случае неверного ввода другого.

Таблица 1. Анализ программы.

4 Отладка и тестирование

В качестве среды разработки была выбрана программа VisualStudio 2019. Программа предоставляет все средства, необходимые при разработке и отладке вырабатываемых модулей и программ. Для отладки использовались такие инструменты, как точка останова, выполнение кода по шагам, анализ содержимого локальных и глобальных переменных, анализ содержимого памяти.

Тестирование проводилось в рабочем порядке, в процессе разработки, после завершения написания программы. В ходе тестирования было выявлено и исправлено множество проблем, связанных с подсчетом результата при использовании операций и функций, освобождением выделенной памяти и многое другое.

5 Описание программы

Итак, программа «Калькулятор» может быть использована в трех вариантах: режим «Обычный», режим «Инженерный» и режим «Программист».

Обычный калькулятор предназначен для выполнения стандартных арифметических функций (сложение, вычитание, умножение, деление), включая операции с буферной памятью:

MS – занести в память содержимое индикатора;

MC – очистить память;

MR – вывести на экран содержимое памяти;

M+ – прибавить значение, отображаемое на индикаторе, к содержимому памяти;

M– – вычесть значение, отображаемое на индикаторе, от содержимого памяти.

Для стандартных арифметических операций была реализована функция void. Алгоритм работы представлен на рис. 5.

Причем, необходимо учесть блокировку кнопок MC и MR до того момента, как пользователь занесет значение в память с помощью кнопки MS. При очистке памяти также происходит блокировка этих же клавиш. Принцип работы представлен на рис. 6.

Клавиши имеют индуктивно понятные обозначения. Так, кнопка с плавающей запятой, вычисления квадратного корня, изменения знака на противоположный и подсчета процента имеют специальный символ, а кнопка $1/x$ используется для вычисления обратной величины.

Неверные введенные данные удаляются по одной цифре с помощью клавиши, которая содержит символ стрелки. Все содержимое индикатора можно удалить щелчком мыши на кнопку CE, а полностью очистить память калькулятора – щелчком мыши на кнопку C.

Для реализации данного калькулятора были выявлены некоторые исключения: число индикатора имеет разрядность, недопустимость деления числа на ноль (и при вычислении обратной функции числа). Число на индикаторе имеет разрядность 32 бит (в случае ввода числа, которая имеет разрядность выше, чем 32 бит, клавиши с цифрами будут заблокированы, а в случае вывода большого числа (результата) будет выведена экспоненциальная форма).

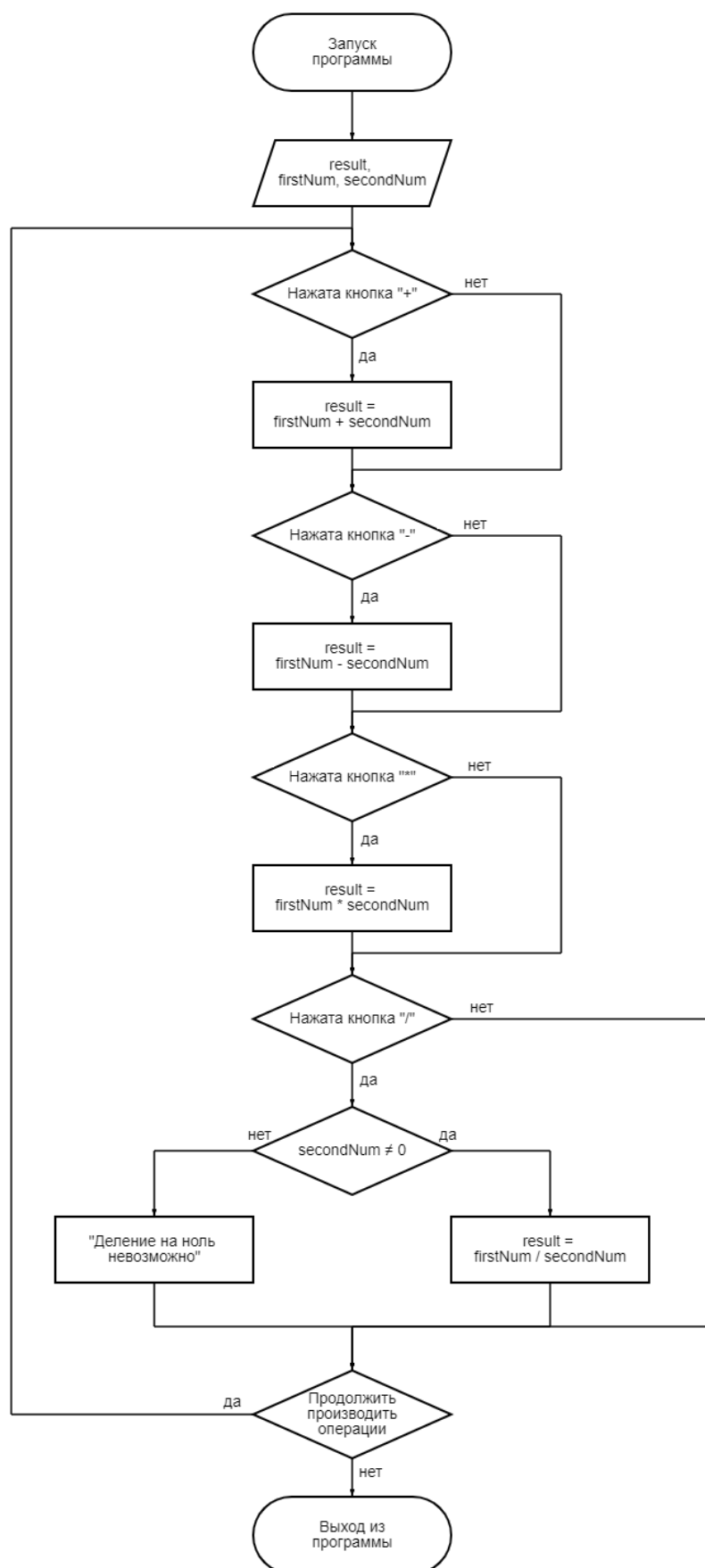


Рисунок 5. Принцип работы с арифметическими операциями.

Инженерный калькулятор соответствует обычному калькулятору, но имеет диапазон нестандартных функций: константы числа «Пи» и числа Эйлера, десятичный и натуральный логарифм, возведение числа в любую степень.

В ходе работы была использована ассемблерная вставка для констант:

```
void main()
{
    double pi = 0.1415926535;
    double e = 0.7182818284;

    int pii, ee;

    _asm {
        mov eax, 3
        mov pii, eax
    }

    _asm {
        mov eax, 2
        mov ee, eax
    }

    pi = pi + pii;
    e = e + ee;
}
```

Калькулятор в режиме «Программист» имеет набор стандартных арифметических функций, а также функцию перевода десятичного числа в разные системы счисления, а именно в двоичную, восьмеричную и шестнадцатеричную. Для этого заблокирована плавающая запятая, а также есть возможность представить число в дополнительном коде в случае отрицательного числа.

6 Руководство пользователя

При запуске программы откроется графический интерфейс, который представляет собой стандартный калькулятор, или же обычный (рис. 7).

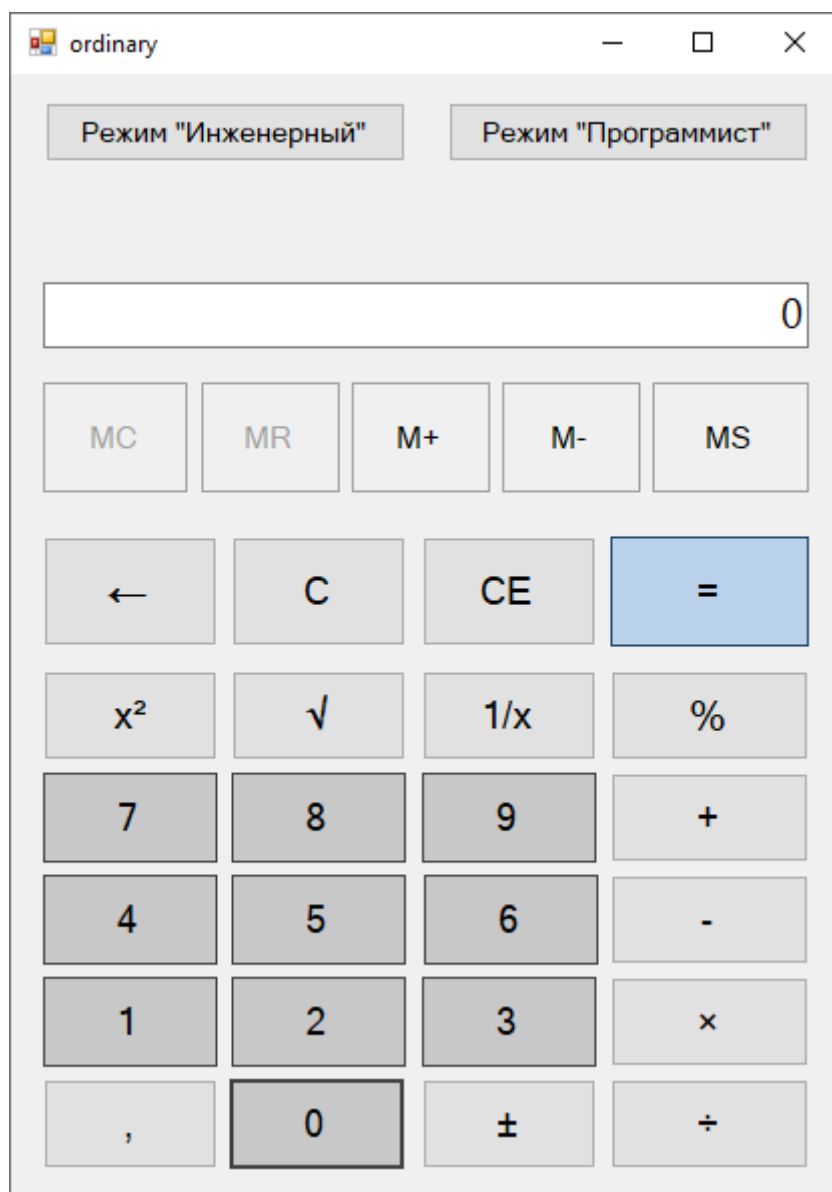


Рисунок 7. Калькулятор в режиме «Обычный».

При нажатии на кнопки, содержащие цифры, наше текстовое поле, которое изначально содержало значение ноль, будет заполняться цифрами.

Если необходимо рассчитать какое-нибудь простое выражение, например, сложить одно число с другим, необходимо выбрать кнопку операции сложения, далее выбрать второе число и нажать кнопку знака «равно». Тогда в текстовом поле будет выведен полученный результат.

Для расчетов с отрицательным числом была реализована кнопка, которая позволяет изменить знак числа на противоположный.

Также в обычном калькуляторе есть возможность высчитать процент числа. Например, необходимо найти 12% от 24. Для этого мы набираем 24 в текстовое поле и умножаем на 12. Затем нажимаем кнопку процент. Результат в текстовом поле равен 2,88.

Для функций возведения числа в квадрат, извлечения корня и нахождения обратной величины числа необходимо ввести число с помощью кнопок (или же это можно делать на клавиатуре) и выбрать соответствующую функцию.

Сверху имеются кнопки для перехода в другой режим – Инженерный и Режим «Программист».

Режим «Инженерный» (рис. 8) содержит по большей части нестандартные функции. Руководство использования в данном калькуляторе должно быть понятно.

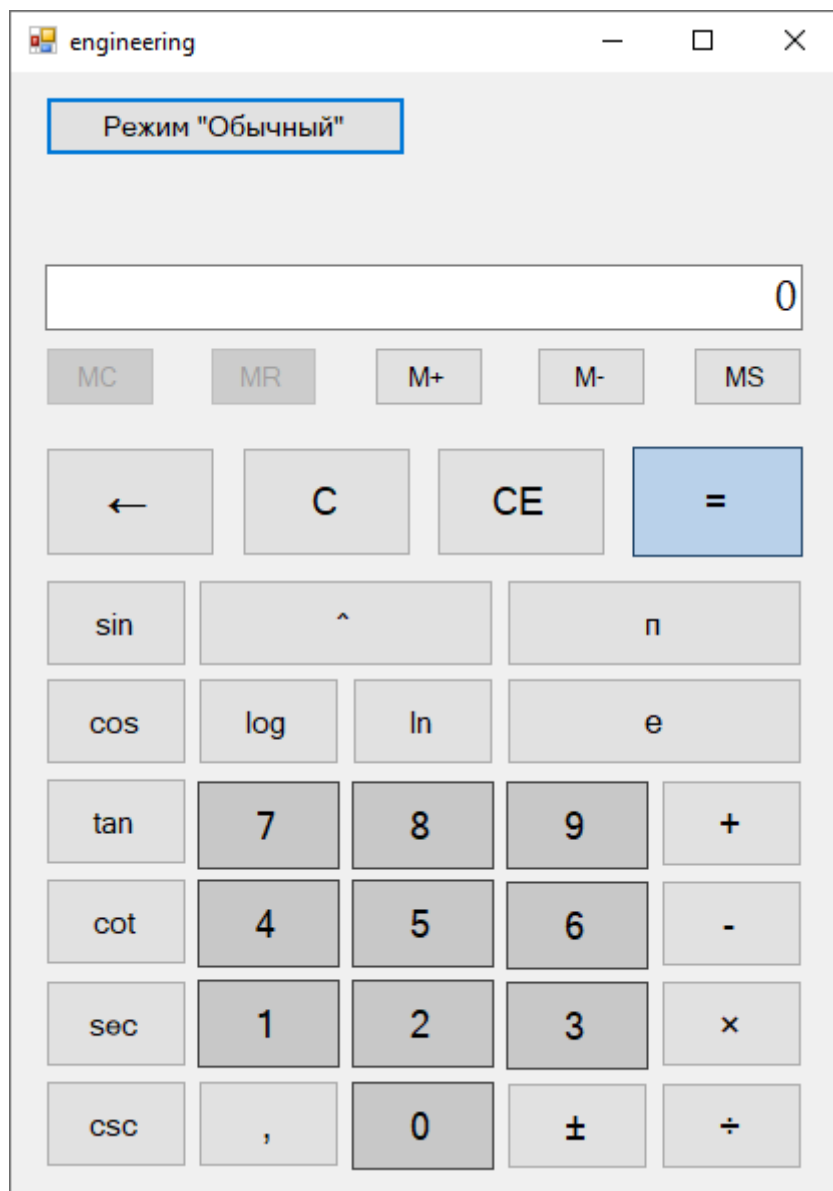


Рисунок 8. Калькулятор в режиме «Инженерный».

Калькулятор в режиме «Программист» (рис. 9) представляет введенное десятичное число в двоичной, восьмеричной и шестнадцатеричной системе счисления. Обработка результата ведется даже при работе с арифметическими операциями. Для этого калькулятора характерны две вещи – блокировка плавающей запятой и возможность перевода числа в дополнительный код, даже при отрицательном.

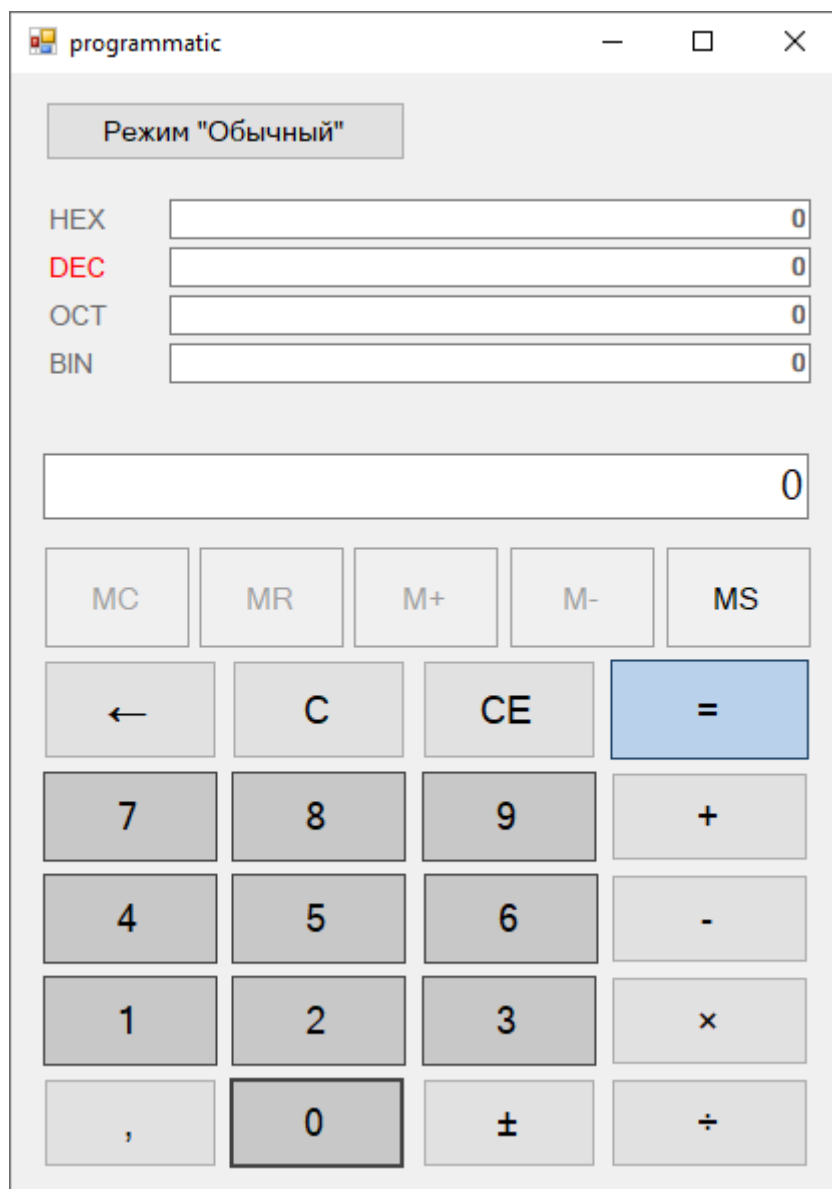


Рисунок 9. Калькулятор в режиме «Программист».

Заключение

При выполнении данной курсовой работы были изучены навыки разработки многомодульных программ. В результате выполнения данного проекта был разработан программный продукт "Калькулятор". При помощи этой программы можно высчитывать определенные арифметические операции. Программа не занимает много места, не требовательна к установленному программному обеспечению. Также было проведено исследование полученного программного продукта. В результате были выявлены следующие достоинства и недостатки полученного программного продукта:

Достоинства:

- Существует возможность сбрасывать результат;
- Программный продукт малотребователен к системным ресурсам компьютера.

Недостатки:

- Невозможность выполнения некоторых других математических операций.

В целом, поставленная в начале курсового проекта цель была достигнута. В программе выполняются все необходимые функциональные требования.

Список используемых источников

<https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/controls/windows-forms-controls-by-function?view=netframeworkdesktop-4.8>

<https://docs.microsoft.com/ru-ru/cpp/dotnet/graphics-operations-cpp-cli?view=msvc-160&viewFallbackFrom=vs-2017>

<https://docs.microsoft.com/ru-ru/cpp/cpp/c-cpp-language-and-standard-libraries?view=msvc-160&viewFallbackFrom=vs-2019>

Приложение А Листинг программы

Исходный файл «ordinary.cpp»

```
#include "ordinary.h"

using namespace System;
using namespace System::Windows::Forms;

[STAThread]
void Main(cli::array<String^>^ args)
{
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);

    CWCalculator::ordinary form;
    Application::Run(% form);
    //перегрузка для запуска основной формой
}
```

Заголовочный файл «ordinary.h»

```
#pragma once
#include "engineering.h"
#include "programmatic.h"

namespace CWCalculator {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    /// <summary>
    /// Сводка для ordinary
    /// </summary>
    public ref class ordinary : public
System::Windows::Forms::Form
    {
    public:
        ordinary(void)
        {
            InitializeComponent();
            //
            //TODO: добавьте код конструктора
            //
            MC->Enabled = false;
            MR->Enabled = false;
        }

    protected:
        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        ~ordinary()
        {
            if (components)
            {
                delete components;
            }
        }

    private: System::Windows::Forms::Button^ sqrt;
    protected:
    private: System::Windows::Forms::Button^ degree2;
    private: System::Windows::Forms::Button^ divideoneby1;
    private: System::Windows::Forms::Button^ percent;
    private: System::Windows::Forms::Button^ Mminus;
    private: System::Windows::Forms::Button^ Mplus;
    private: System::Windows::Forms::Button^ MC;
    private: System::Windows::Forms::Button^ MR;
    private: System::Windows::Forms::Button^ MS;
```

```

private: System::Windows::Forms::Label^ label1;
private: System::Windows::Forms::Button^ buttonprogrammatic;
private: System::Windows::Forms::Button^ buttonengineering;
private: System::Windows::Forms::TextBox^ textBox1;
private: System::Windows::Forms::Button^ change;
private: System::Windows::Forms::Button^ erase2;
private: System::Windows::Forms::Button^ erasel;
private: System::Windows::Forms::Button^ clear;
private: System::Windows::Forms::Button^ division;
private: System::Windows::Forms::Button^ multiplication;
private: System::Windows::Forms::Button^ minus;
private: System::Windows::Forms::Button^ plus;
private: System::Windows::Forms::Button^ equalsign;
private: System::Windows::Forms::Button^ point;
private: System::Windows::Forms::Button^ button9;
private: System::Windows::Forms::Button^ button8;
private: System::Windows::Forms::Button^ button7;
private: System::Windows::Forms::Button^ button6;
private: System::Windows::Forms::Button^ button5;
private: System::Windows::Forms::Button^ button4;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button1;
private: System::Windows::Forms::Button^ button0;

private:
    /// <summary>
    /// Обязательная переменная конструктора.
    /// </summary>
    System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Требуемый метод для поддержки конструктора — не
изменяйте
    /// содержимое этого метода с помощью редактора кода.
    /// </summary>
    void InitializeComponent(void)
    {
        this->sqrт = (gcnew
System::Windows::Forms::Button());
        this->degree2 = (gcnew
System::Windows::Forms::Button());
        this->divideoneby1 = (gcnew
System::Windows::Forms::Button());
        this->percent = (gcnew
System::Windows::Forms::Button());
        this->Mminus = (gcnew
System::Windows::Forms::Button());
        this->Mplus = (gcnew
System::Windows::Forms::Button());
        this->MC = (gcnew System::Windows::Forms::Button());
        this->MR = (gcnew System::Windows::Forms::Button());
    }

```

```

        this->MS = (gcnew System::Windows::Forms::Button());
        this->label1 = (gcnew
System::Windows::Forms::Label());
        this->buttonprogrammatic = (gcnew
System::Windows::Forms::Button());
        this->buttonengineering = (gcnew
System::Windows::Forms::Button());
        this->textBox1 = (gcnew
System::Windows::Forms::TextBox());
        this->change = (gcnew
System::Windows::Forms::Button());
        this->erase2 = (gcnew
System::Windows::Forms::Button());
        this->eraset1 = (gcnew
System::Windows::Forms::Button());
        this->clear = (gcnew
System::Windows::Forms::Button());
        this->division = (gcnew
System::Windows::Forms::Button());
        this->multiplication = (gcnew
System::Windows::Forms::Button());
        this->minus = (gcnew
System::Windows::Forms::Button());
        this->plus = (gcnew
System::Windows::Forms::Button());
        this->equalsign = (gcnew
System::Windows::Forms::Button());
        this->point = (gcnew
System::Windows::Forms::Button());
        this->button9 = (gcnew
System::Windows::Forms::Button());
        this->button8 = (gcnew
System::Windows::Forms::Button());
        this->button7 = (gcnew
System::Windows::Forms::Button());
        this->button6 = (gcnew
System::Windows::Forms::Button());
        this->button5 = (gcnew
System::Windows::Forms::Button());
        this->button4 = (gcnew
System::Windows::Forms::Button());
        this->button3 = (gcnew
System::Windows::Forms::Button());
        this->button2 = (gcnew
System::Windows::Forms::Button());
        this->button1 = (gcnew
System::Windows::Forms::Button());
        this->button0 = (gcnew
System::Windows::Forms::Button());
        this->SuspendLayout();
        //
        // sqrt
        //

```

```

        this->sqrt->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14.25F,
System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->sqrt->Location = System::Drawing::Point(110,
298);

        this->sqrt->Name = L"sqrt";
        this->sqrt->Size = System::Drawing::Size(87, 45);
        this->sqrt->TabIndex = 96;
        this->sqrt->Text = L"√";
        this->sqrt->UseVisualStyleBackColor = true;
        this->sqrt->Click += gcnew
System::EventHandler(this, &ordinary::sqrt_Click);
        //
        // degree2
        //
        this->degree2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->degree2->Location = System::Drawing::Point(16,
298);

        this->degree2->Name = L"degree2";
        this->degree2->Size = System::Drawing::Size(87, 45);
        this->degree2->TabIndex = 95;
        this->degree2->Text = L"x²";
        this->degree2->UseVisualStyleBackColor = true;
        this->degree2->Click += gcnew
System::EventHandler(this, &ordinary::degree2_Click);
        //
        // divideoneby1
        //
        this->divideoneby1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->divideoneby1->Location =
System::Drawing::Point(205, 298);
        this->divideoneby1->Name = L"divideoneby1";
        this->divideoneby1->Size = System::Drawing::Size(87,
45);

        this->divideoneby1->TabIndex = 94;
        this->divideoneby1->Text = L"1/x";
        this->divideoneby1->UseVisualStyleBackColor = true;
        this->divideoneby1->Click += gcnew
System::EventHandler(this, &ordinary::divideoneby1_Click);
        //
        // percent
        //
        this->percent->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 15.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->percent->Location =
System::Drawing::Point(299, 298);

```

```

        this->percent->Name = L"percent";
        this->percent->Size = System::Drawing::Size(99, 45);
        this->percent->TabIndex = 93;
        this->percent->Text = L"%";
        this->percent->UseVisualStyleBackColor = true;
        this->percent->Click += gcnew
System::EventHandler(this, &ordinary::percent_Click);
        //
        // Mminus
        //
        this->Mminus->BackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->Cursor =
System::Windows::Forms::Cursors::Default;
        this->Mminus->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatAppearance->BorderSize = 0;
        this->Mminus->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->Mminus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->Mminus->Location = System::Drawing::Point(245,
154);

        this->Mminus->Name = L"Mminus";
        this->Mminus->Size = System::Drawing::Size(69, 55);
        this->Mminus->TabIndex = 92;
        this->Mminus->Text = L"M-";
        this->Mminus->UseVisualStyleBackColor = false;
        this->Mminus->Click += gcnew
System::EventHandler(this, &ordinary::Mminus_Click);
        //
        // Mplus
        //
        this->Mplus->BackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->Cursor =
System::Windows::Forms::Cursors::Default;
        this->Mplus->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatAppearance->BorderSize = 0;
        this->Mplus->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;

```

```

        this->Mplus->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->Mplus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->Mplus->Location = System::Drawing::Point(170,
154);

        this->Mplus->Name = L"Mplus";
        this->Mplus->Size = System::Drawing::Size(69, 55);
        this->Mplus->TabIndex = 91;
        this->Mplus->Text = L"M+";
        this->Mplus->UseVisualStyleBackColor = false;
        this->Mplus->Click += gcnew
System::EventHandler(this, &ordinary::Mplus_Click);
        //
        // MC
        //
        this->MC->BackColor =
System::Drawing::SystemColors::Control;
        this->MC->Cursor =
System::Windows::Forms::Cursors::Default;
        this->MC->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatAppearance->BorderSize = 0;
        this->MC->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->MC->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->MC->Location = System::Drawing::Point(16,
154);

        this->MC->Name = L"MC";
        this->MC->Size = System::Drawing::Size(72, 55);
        this->MC->TabIndex = 90;
        this->MC->Text = L"MC";
        this->MC->UseVisualStyleBackColor = false;
        this->MC->Click += gcnew System::EventHandler(this,
&ordinary::MC_Click);
        //
        // MR
        //

```

```

        this->MR->BackColor =
System::Drawing::SystemColors::Control;
        this->MR->Cursor =
System::Windows::Forms::Cursors::Default;
        this->MR->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->MR->FlatAppearance->BorderSize = 0;
        this->MR->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->MR->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->MR->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->MR->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->MR->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->MR->Location = System::Drawing::Point(95,
154);

        this->MR->Name = L"MR";
        this->MR->Size = System::Drawing::Size(69, 55);
        this->MR->TabIndex = 89;
        this->MR->Text = L"MR";
        this->MR->UseVisualStyleBackColor = false;
        this->MR->Click += gcnew System::EventHandler(this,
&ordinary::MR_Click);
        //
        // MS
        //
        this->MS->BackColor =
System::Drawing::SystemColors::Control;
        this->MS->Cursor =
System::Windows::Forms::Cursors::Default;
        this->MS->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatAppearance->BorderSize = 0;
        this->MS->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->MS->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));

```

```

        this->MS->Location = System::Drawing::Point(320,
154);
        this->MS->Name = L"MS";
        this->MS->Size = System::Drawing::Size(78, 55);
        this->MS->TabIndex = 88;
        this->MS->Text = L"MS";
        this->MS->UseVisualStyleBackColor = false;
        this->MS->Click += gcnew System::EventHandler(this,
&ordinary::MS_Click);
        //
        // label1
        //
        this->label1->BackColor =
System::Drawing::SystemColors::Control;
        this->label1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label1->ForeColor =
System::Drawing::Color::DarkGray;
        this->label1->Location = System::Drawing::Point(16,
77);
        this->label1->MaximumSize =
System::Drawing::Size(500, 90);
        this->label1->Name = L"label1";
        this->label1->Size = System::Drawing::Size(382, 24);
        this->label1->TabIndex = 87;
        this->label1->TextAlign =
System::Drawing::ContentAlignment::MiddleRight;
        //
        // buttonprogrammatic
        //
        this->buttonprogrammatic->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->buttonprogrammatic->Location =
System::Drawing::Point(218, 14);
        this->buttonprogrammatic->Name =
L"buttonprogrammatic";
        this->buttonprogrammatic->Size =
System::Drawing::Size(180, 30);
        this->buttonprogrammatic->TabIndex = 86;
        this->buttonprogrammatic->Text = L"Режим
\"Программист\"";
        this->buttonprogrammatic->UseVisualStyleBackColor =
true;
        this->buttonprogrammatic->Click += gcnew
System::EventHandler(this, &ordinary::buttonprogrammatic_Click);
        //
        // buttonengineering
        //
        this->buttonengineering->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->buttonengineering->Location =
System::Drawing::Point(17, 14);

```

```

        this->buttonengineering->Name =
L"buttonengineering";
        this->buttonengineering->Size =
System::Drawing::Size(180, 30);
        this->buttonengineering->TabIndex = 85;
        this->buttonengineering->Text = L"Режим
\"Инженерный\"";
        this->buttonengineering->UseVisualStyleBackColor =
true;
        this->buttonengineering->Click += gcnew
System::EventHandler(this, &ordinary::buttonengineering_Click);
        //
        // textBox1
        //
        this->textBox1->Font = (gcnew
System::Drawing::Font(L"Leelawadee", 15.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(0)));
        this->textBox1->Location =
System::Drawing::Point(16, 104);
        this->textBox1->Name = L"textBox1";
        this->textBox1->Size = System::Drawing::Size(382,
33);
        this->textBox1->TabIndex = 84;
        this->textBox1->Text = L"0";
        this->textBox1->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        this->textBox1->TextChanged += gcnew
System::EventHandler(this, &ordinary::textBox1_TextChanged);
        //
        // change
        //
        this->change->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->change->Location = System::Drawing::Point(205,
502);
        this->change->Name = L"change";
        this->change->Size = System::Drawing::Size(87, 45);
        this->change->TabIndex = 83;
        this->change->Text = L"±";
        this->change->UseVisualStyleBackColor = true;
        this->change->Click += gcnew
System::EventHandler(this, &ordinary::change_Click);
        //
        // erase2
        //
        this->erase2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->erase2->Location = System::Drawing::Point(205,
231);
        this->erase2->Name = L"erase2";
        this->erase2->Size = System::Drawing::Size(87, 55);

```

```

        this->erase2->TabIndex = 82;
        this->erase2->Text = L"CE";
        this->erase2->UseVisualStyleBackColor = true;
        this->erase2->Click += gcnew
System::EventHandler(this, &ordinary::erase2_Click);
        //
        // erasel
        //
        this->erasel->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->erasel->Location = System::Drawing::Point(110,
231);

        this->erasel->Name = L"erasel";
        this->erasel->Size = System::Drawing::Size(87, 55);
        this->erasel->TabIndex = 81;
        this->erasel->Text = L"C";
        this->erasel->UseVisualStyleBackColor = true;
        this->erasel->Click += gcnew
System::EventHandler(this, &ordinary::erasel_Click);
        //
        // clear
        //
        this->clear->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14,
System::Drawing::FontStyle::Bold));
        this->clear->Location = System::Drawing::Point(16,
231);

        this->clear->Name = L"clear";
        this->clear->Size = System::Drawing::Size(87, 55);
        this->clear->TabIndex = 80;
        this->clear->Text = L"←";
        this->clear->UseVisualStyleBackColor = true;
        this->clear->Click += gcnew
System::EventHandler(this, &ordinary::clear_Click);
        //
        // division
        //
        this->division->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->division->Location =
System::Drawing::Point(299, 502);
        this->division->Name = L"division";
        this->division->Size = System::Drawing::Size(99,
45);

        this->division->TabIndex = 79;
        this->division->Text = L"÷";
        this->division->UseVisualStyleBackColor = true;
        this->division->Click += gcnew
System::EventHandler(this, &ordinary::arithmetic);
        //
        // multiplication
        //

```

```

        this->multiplication->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->multiplication->Location =
System::Drawing::Point(299, 451);
        this->multiplication->Name = L"multiplication";
        this->multiplication->Size =
System::Drawing::Size(99, 45);
        this->multiplication->TabIndex = 78;
        this->multiplication->Text = L"×";
        this->multiplication->UseVisualStyleBackColor =
true;

        this->multiplication->Click += gcnew
System::EventHandler(this, &ordinary::arithmetic);
        //
        // minus
        //
        this->minus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->minus->Location = System::Drawing::Point(299,
400);

        this->minus->Name = L"minus";
        this->minus->Size = System::Drawing::Size(99, 45);
        this->minus->TabIndex = 77;
        this->minus->Text = L"-";
        this->minus->UseVisualStyleBackColor = true;
        this->minus->Click += gcnew
System::EventHandler(this, &ordinary::arithmetic);
        //
        // plus
        //
        this->plus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->plus->Location = System::Drawing::Point(299,
349);

        this->plus->Name = L"plus";
        this->plus->Size = System::Drawing::Size(99, 45);
        this->plus->TabIndex = 76;
        this->plus->Text = L"+";
        this->plus->UseVisualStyleBackColor = true;
        this->plus->Click += gcnew
System::EventHandler(this, &ordinary::arithmetic);
        //
        // equalsign
        //
        this->equalsign->BackColor =
System::Drawing::SystemColors::GradientActiveCaption;
        this->equalsign->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->equalsign->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->equalsign->Location =
System::Drawing::Point(299, 231);
        this->equalsign->Name = L"equalsign";

```

```

        this->equalsign->Size = System::Drawing::Size(99,
55);
        this->equalsign->TabIndex = 75;
        this->equalsign->Text = L"=";
        this->equalsign->UseVisualStyleBackColor = false;
        this->equalsign->Click += gcnew
System::EventHandler(this, &ordinary::equalsign_Click);
        //
        // point
        //
        this->point->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->point->Location = System::Drawing::Point(16,
502);
        this->point->Name = L"point";
        this->point->Size = System::Drawing::Size(87, 45);
        this->point->TabIndex = 74;
        this->point->Text = L",";
        this->point->UseVisualStyleBackColor = true;
        this->point->Click += gcnew
System::EventHandler(this, &ordinary::point_Click);
        //
        // button9
        //
        this->button9->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button9->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button9->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button9->Location =
System::Drawing::Point(205, 349);
        this->button9->Name = L"button9";
        this->button9->Size = System::Drawing::Size(87, 45);
        this->button9->TabIndex = 73;
        this->button9->Text = L"9";
        this->button9->UseVisualStyleBackColor = false;
        this->button9->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button8
        //
        this->button8->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button8->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button8->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button8->Location =
System::Drawing::Point(110, 349);
        this->button8->Name = L"button8";
        this->button8->Size = System::Drawing::Size(87, 45);
        this->button8->TabIndex = 72;

```

```

        this->button8->Text = L"8";
        this->button8->UseVisualStyleBackColor = false;
        this->button8->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button7
        //
        this->button7->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button7->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button7->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button7->Location = System::Drawing::Point(16,
349);

        this->button7->Name = L"button7";
        this->button7->Size = System::Drawing::Size(87, 45);
        this->button7->TabIndex = 71;
        this->button7->Text = L"7";
        this->button7->UseVisualStyleBackColor = false;
        this->button7->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button6
        //
        this->button6->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button6->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button6->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button6->Location =
System::Drawing::Point(206, 400);
        this->button6->Name = L"button6";
        this->button6->Size = System::Drawing::Size(87, 45);
        this->button6->TabIndex = 70;
        this->button6->Text = L"6";
        this->button6->UseVisualStyleBackColor = false;
        this->button6->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button5
        //
        this->button5->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button5->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button5->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button5->Location =
System::Drawing::Point(110, 400);
        this->button5->Name = L"button5";
        this->button5->Size = System::Drawing::Size(87, 45);

```

```

        this->button5->TabIndex = 69;
        this->button5->Text = L"5";
        this->button5->UseVisualStyleBackColor = false;
        this->button5->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button4
        //
        this->button4->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button4->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button4->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button4->Location = System::Drawing::Point(16,
400);

        this->button4->Name = L"button4";
        this->button4->Size = System::Drawing::Size(87, 45);
        this->button4->TabIndex = 68;
        this->button4->Text = L"4";
        this->button4->UseVisualStyleBackColor = false;
        this->button4->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button3
        //
        this->button3->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button3->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button3->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button3->Location =
System::Drawing::Point(205, 451);
        this->button3->Name = L"button3";
        this->button3->Size = System::Drawing::Size(87, 45);
        this->button3->TabIndex = 67;
        this->button3->Text = L"3";
        this->button3->UseVisualStyleBackColor = false;
        this->button3->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button2
        //
        this->button2->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button2->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button2->Location =
System::Drawing::Point(110, 451);
        this->button2->Name = L"button2";

```

```

        this->button2->Size = System::Drawing::Size(87, 45);
        this->button2->TabIndex = 66;
        this->button2->Text = L"2";
        this->button2->UseVisualStyleBackColor = false;
        this->button2->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button1
        //
        this->button1->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button1->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button1->Location = System::Drawing::Point(16,
451);

        this->button1->Name = L"button1";
        this->button1->Size = System::Drawing::Size(87, 45);
        this->button1->TabIndex = 65;
        this->button1->Text = L"1";
        this->button1->UseVisualStyleBackColor = false;
        this->button1->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // button0
        //
        this->button0->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button0->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button0->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button0->Location =
System::Drawing::Point(109, 502);
        this->button0->Name = L"button0";
        this->button0->Size = System::Drawing::Size(87, 45);
        this->button0->TabIndex = 64;
        this->button0->Text = L"0";
        this->button0->UseVisualStyleBackColor = false;
        this->button0->Click += gcnew
System::EventHandler(this, &ordinary::numberbuttons);
        //
        // ordinary
        //
        this->AutoScaleDimensions =
System::Drawing::SizeF(6, 13);
        this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(414, 561);
        this->Controls->Add(this->sqr);
        this->Controls->Add(this->degree2);
        this->Controls->Add(this->divideoneby1);

```

```

        this->Controls->Add(this->percent);
        this->Controls->Add(this->Mminus);
        this->Controls->Add(this->Mplus);
        this->Controls->Add(this->MC);
        this->Controls->Add(this->MR);
        this->Controls->Add(this->MS);
        this->Controls->Add(this->label1);
        this->Controls->Add(this->buttonprogrammatic);
        this->Controls->Add(this->buttonengineering);
        this->Controls->Add(this->textBox1);
        this->Controls->Add(this->change);
        this->Controls->Add(this->erase2);
        this->Controls->Add(this->erasel);
        this->Controls->Add(this->clear);
        this->Controls->Add(this->division);
        this->Controls->Add(this->multiplication);
        this->Controls->Add(this->minus);
        this->Controls->Add(this->plus);
        this->Controls->Add(this->equalsign);
        this->Controls->Add(this->point);
        this->Controls->Add(this->button9);
        this->Controls->Add(this->button8);
        this->Controls->Add(this->button7);
        this->Controls->Add(this->button6);
        this->Controls->Add(this->button5);
        this->Controls->Add(this->button4);
        this->Controls->Add(this->button3);
        this->Controls->Add(this->button2);
        this->Controls->Add(this->button1);
        this->Controls->Add(this->button0);
        this->Name = L"ordinary";
        this->StartPosition =
System::Windows::Forms::FormStartPosition::CenterScreen;
        this->Text = L"ordinary";
        this->ResumeLayout(false);
        this->PerformLayout();

    }
#pragma endregion

    double firstNum = Double::NaN;
    double secondNum = Double::NaN;
    double result = Double::NaN;
    double memory = 0;
    String^ currentOper;
    bool isCalculated = false;
    bool flag = false;

    void calculate()
    {
        if (!Double::IsNaN(firstNum) &&
!Double::IsNaN(secondNum))
        {

```

```

        if (currentOper == L"+")
        {
            result = firstNum + secondNum;
        }
        else if (currentOper == L"-")
        {
            result = firstNum - secondNum;
        }
        else if (currentOper == L"x")
        {
            result = firstNum * secondNum;
        }
        else if (currentOper == L"÷")
        {
            if (secondNum != 0)
            {
                result = firstNum / secondNum;
            }
            else
            {
                MessageBox::Show("Деление на ноль невозможно", "Error",
                MessageBoxButtons::OK, MessageBoxIcon::Error);
                return;
            }
        }

        label1->Text += String::Format(" {0} =",
secondNum);
        textBox1->Text =
System::Convert::ToString(result);
        isCalculated = true;
    }
}

private: System::Void buttonengineering_Click(System::Object^
sender, System::EventArgs^ e)
{
    engineering^ Fo2 = gcnew engineering();
    Fo2->Show(); //показываем вторую форму
    ordinary::Hide();
    Fo2->Owner = this;
}

private: System::Void buttonprogrammatic_Click(System::Object^
sender, System::EventArgs^ e)
{
    programmatic^ Fo2 = gcnew programmatic();
    Fo2->Show(); //показываем третью форму
    ordinary::Hide();
    Fo2->Owner = this;
}

private: System::Void numberbuttons(System::Object^ sender,
System::EventArgs^ e)

```

```

{
    Button^ numbers = safe_cast<Button^>(sender);

    if (textBox1->Text == "0")
    {
        textBox1->Text = numbers->Text;
    }
    else if (textBox1->Text->Length <= 31)
    {
        textBox1->Text += numbers->Text;
    }
    isCalculated = false;
}

private: System::Void arithmetic(System::Object^ sender,
System::EventArgs^ e)
{
    Button^ operators = safe_cast<Button^>(sender);
    String^ outNumber;

    // second operation
    if (labell1->Text != "")
    {
        // second number
        if (!isCalculated)
        {
            secondNum = Double::Parse(textBox1->Text);
        }

        if (Double::IsNaN(result))
        {
            calculate();
        }
        else
        {
            firstNum = result;
            if (!isCalculated)
            {
                calculate();
            }
        }
        outNumber = System::Convert::ToString(result);
    }
    else // first operation
    {
        firstNum = Double::Parse(textBox1->Text);
        outNumber = System::Convert::ToString(firstNum);
    }

    currentOper = operators->Text;
    textBox1->Text = "";
    labell1->Text = outNumber + " " + currentOper;
}

```

```

private: System::Void equalsign_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!isCalculated)
    {
        //firstNum = Double::IsNaN(result) ? firstNum : result;
        secondNum = Double::Parse(textBox1->Text);
    }
    else
    {
        //firstNum = result;
        labell->Text = String::Format("{0} {1}", firstNum,
currentOper);
    }

    firstNum = Double::IsNaN(result) ? firstNum : result;
    //secondNum = isCalculated ? secondNum :
Double::Parse(CalculationBox->Text);

    calculate();
}
private: System::Void clear_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text->Length > 0)
    {
        textBox1->Text = textBox1->Text->Remove(textBox1->Text-
>Length - 1, 1);
    }
}
private: System::Void erasel_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = "0";
    labell->Text = "";
    result = Double::NaN;
    firstNum = Double::NaN;
    secondNum = Double::NaN;
    isCalculated = false;
    currentOper = "";
}
private: System::Void erase2_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = "";
}
private: System::Void point_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!textBox1->Text->Contains(","))
    {
        textBox1->Text += ",";
    }
}

```

```

}
private: System::Void change_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text->Contains("-"))
    {
        textBox1->Text = textBox1->Text->Remove(0, 1);
    }
    else
    {
        textBox1->Text = "-" + textBox1->Text;
    }
}
private: System::Void degree2_Click(System::Object^ sender,
System::EventArgs^ e)
{
    label1->Text = String::Format("sqr ({0})", textBox1->Text);
    textBox1->Text =
System::Convert::ToString(Math::Pow(Double::Parse(textBox1->Text),
2));
}
private: System::Void sqrt_Click(System::Object^ sender,
System::EventArgs^ e)
{
    label1->Text = String::Format("sqrt ({0})", textBox1->Text);
    textBox1->Text =
System::Convert::ToString(Math::Sqrt(Double::Parse(textBox1-
>Text)));
}
private: System::Void divideoneby1_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text != "0")
    {
        label1->Text = String::Format("1/({0})", textBox1->Text);
        textBox1->Text = System::Convert::ToString(1 /
Double::Parse(textBox1->Text));
    }
    else
    {
        MessageBox::Show("Деление на ноль невозможно", "Error",
MessageBoxButtons::OK, MessageBoxIcon::Error);
        return;
    }
}
private: System::Void percent_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!Double::IsNaN(firstNum) && label1->Text != "")
    {
        label1->Text = "";
        textBox1->Text = System::Convert::ToString(firstNum / 100
* Double::Parse(textBox1->Text));
    }
}

```

```

    }
}
private: System::Void MS_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory = Double::Parse(textBox1->Text);
    textBox1->Text = System::Convert::ToString(memory);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void MC_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory = 0;
    MC->Enabled = false;
    MR->Enabled = false;
}
private: System::Void MR_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = System::Convert::ToString(memory);
    isCalculated = true;
}
private: System::Void Mplus_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory += Double::Parse(textBox1->Text);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void Mminus_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory -= Double::Parse(textBox1->Text);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void textBox1_TextChanged(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text == "")
    {
        textBox1->Text = "0";
    }
}
};
}

```

Исходный файл «engineering.cpp»

```
#include "engineering.h"
#include "engineering.h"
#include <stdio.h>
void main()
{
    double pi = 0.1415926535;
    double e = 0.7182818284;

    int pii, ee;

    _asm {
        mov eax, 3
        mov pii, eax
    }

    _asm {
        mov eax, 2
        mov ee, eax
    }

    pi = pi + pii;
    e = e + ee;
}
```

Заголовочный файл «engineering.h»

```
#pragma once
#include <string>
#include <stdlib.h>
#include <iostream>

namespace CWCalculator {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    /// <summary>
    /// Сводка для engineering
    /// </summary>
    public ref class engineering : public
System::Windows::Forms::Form
    {
    public:
        engineering(void)
        {
            InitializeComponent();
            //
            //TODO: добавьте код конструктора
            //
            MC->Enabled = false;
            MR->Enabled = false;
        }

    protected:
        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        ~engineering()
        {
            if (components)
            {
                delete components;
            }
        }

    private: System::Windows::Forms::Button^ numberpi;
    protected:
    private: System::Windows::Forms::Button^ exhibitor;
    private: System::Windows::Forms::Button^ natlogarithm;
    private: System::Windows::Forms::Button^ logarithm;
    private: System::Windows::Forms::Button^ degreey;
    private: System::Windows::Forms::Button^ buttoncsc;
    private: System::Windows::Forms::Button^ buttonsec;
    private: System::Windows::Forms::Button^ buttoncot;
```

```

private: System::Windows::Forms::Button^ buttontan;
private: System::Windows::Forms::Button^ buttoncos;
private: System::Windows::Forms::Button^ buttonsin;
private: System::Windows::Forms::Button^ Mminus;
private: System::Windows::Forms::Button^ Mplus;
private: System::Windows::Forms::Button^ MC;
private: System::Windows::Forms::Button^ MR;
private: System::Windows::Forms::Button^ MS;
private: System::Windows::Forms::Label^ label1;
private: System::Windows::Forms::TextBox^ textBox1;
private: System::Windows::Forms::Button^ change;
private: System::Windows::Forms::Button^ erase2;
private: System::Windows::Forms::Button^ erasel;
private: System::Windows::Forms::Button^ clear;
private: System::Windows::Forms::Button^ division;
private: System::Windows::Forms::Button^ multiplication;
private: System::Windows::Forms::Button^ minus;
private: System::Windows::Forms::Button^ plus;
private: System::Windows::Forms::Button^ equalsign;
private: System::Windows::Forms::Button^ point;
private: System::Windows::Forms::Button^ button9;
private: System::Windows::Forms::Button^ button8;
private: System::Windows::Forms::Button^ button7;
private: System::Windows::Forms::Button^ button6;
private: System::Windows::Forms::Button^ button5;
private: System::Windows::Forms::Button^ button4;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button1;
private: System::Windows::Forms::Button^ button0;
private: System::Windows::Forms::Button^ buttonordinary1;

private:
    /// <summary>
    /// Обязательная переменная конструктора.
    /// </summary>
    System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Требуемый метод для поддержки конструктора — не
изменяйте
    /// содержимое этого метода с помощью редактора кода.
    /// </summary>
    void InitializeComponent(void)
    {
        this->numberpi = (gcnew
System::Windows::Forms::Button());
        this->exhibitor = (gcnew
System::Windows::Forms::Button());
        this->natlogarithm = (gcnew
System::Windows::Forms::Button());

```

```

        this->logarithm = (gcnew
System::Windows::Forms::Button());
        this->degreey = (gcnew
System::Windows::Forms::Button());
        this->buttoncsc = (gcnew
System::Windows::Forms::Button());
        this->buttonsec = (gcnew
System::Windows::Forms::Button());
        this->buttoncot = (gcnew
System::Windows::Forms::Button());
        this->buttontan = (gcnew
System::Windows::Forms::Button());
        this->buttoncos = (gcnew
System::Windows::Forms::Button());
        this->buttonsin = (gcnew
System::Windows::Forms::Button());
        this->Mminus = (gcnew
System::Windows::Forms::Button());
        this->Mplus = (gcnew
System::Windows::Forms::Button());
        this->MC = (gcnew System::Windows::Forms::Button());
        this->MR = (gcnew System::Windows::Forms::Button());
        this->MS = (gcnew System::Windows::Forms::Button());
        this->labell1 = (gcnew
System::Windows::Forms::Label());
        this->textBox1 = (gcnew
System::Windows::Forms::TextBox());
        this->change = (gcnew
System::Windows::Forms::Button());
        this->erase2 = (gcnew
System::Windows::Forms::Button());
        this->erasel = (gcnew
System::Windows::Forms::Button());
        this->clear = (gcnew
System::Windows::Forms::Button());
        this->division = (gcnew
System::Windows::Forms::Button());
        this->multiplication = (gcnew
System::Windows::Forms::Button());
        this->minus = (gcnew
System::Windows::Forms::Button());
        this->plus = (gcnew
System::Windows::Forms::Button());
        this->>equalsign = (gcnew
System::Windows::Forms::Button());
        this->point = (gcnew
System::Windows::Forms::Button());
        this->button9 = (gcnew
System::Windows::Forms::Button());
        this->button8 = (gcnew
System::Windows::Forms::Button());
        this->button7 = (gcnew
System::Windows::Forms::Button());

```

```

        this->button6 = (gcnew
System::Windows::Forms::Button());
        this->button5 = (gcnew
System::Windows::Forms::Button());
        this->button4 = (gcnew
System::Windows::Forms::Button());
        this->button3 = (gcnew
System::Windows::Forms::Button());
        this->button2 = (gcnew
System::Windows::Forms::Button());
        this->button1 = (gcnew
System::Windows::Forms::Button());
        this->button0 = (gcnew
System::Windows::Forms::Button());
        this->buttonordinary1 = (gcnew
System::Windows::Forms::Button());
        this->SuspendLayout();
        //
        // numberpi
        //
        this->numberpi->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->numberpi->Location =
System::Drawing::Point(247, 253);
        this->numberpi->Name = L"numberpi";
        this->numberpi->Size = System::Drawing::Size(148,
44);

        this->numberpi->TabIndex = 254;
        this->numberpi->Text = L"π";
        this->numberpi->UseVisualStyleBackColor = true;
        this->numberpi->Click += gcnew
System::EventHandler(this, &engineering::numberpi_Click);
        //
        // exhibitor
        //
        this->exhibitor->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 12,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->exhibitor->Location =
System::Drawing::Point(247, 302);
        this->exhibitor->Name = L"exhibitor";
        this->exhibitor->Size = System::Drawing::Size(148,
44);

        this->exhibitor->TabIndex = 253;
        this->exhibitor->Text = L"e";
        this->exhibitor->UseVisualStyleBackColor = true;
        this->exhibitor->Click += gcnew
System::EventHandler(this, &engineering::exhibitor_Click);

```

```

//
// natlogarithm
//
this->natlogarithm->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));
this->natlogarithm->Location =
System::Drawing::Point(170, 302);
this->natlogarithm->Name = L"natlogarithm";
this->natlogarithm->Size = System::Drawing::Size(71,
44);

this->natlogarithm->TabIndex = 252;
this->natlogarithm->Text = L"ln";
this->natlogarithm->UseVisualStyleBackColor = true;
this->natlogarithm->Click += gcnew
System::EventHandler(this, &engineering::natlogarithm_Click);
//
// logarithm
//
this->logarithm->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));
this->logarithm->Location =
System::Drawing::Point(93, 302);
this->logarithm->Name = L"logarithm";
this->logarithm->Size = System::Drawing::Size(71,
44);

this->logarithm->TabIndex = 251;
this->logarithm->Text = L"log";
this->logarithm->UseVisualStyleBackColor = true;
this->logarithm->Click += gcnew
System::EventHandler(this, &engineering::logarithm_Click);
//
// degreey
//
this->degreey->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));
this->degreey->Location = System::Drawing::Point(93,
253);

this->degreey->Name = L"degreey";
this->degreey->Size = System::Drawing::Size(148,
44);

this->degreey->TabIndex = 250;
this->degreey->Text = L"^";
this->degreey->UseVisualStyleBackColor = true;

```

```

        this->degreey->Click += gcnew
System::EventHandler(this, &engineering::degreey_Click);
        //
        // buttoncsc
        //
        this->buttoncsc->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->buttoncsc->Location =
System::Drawing::Point(17, 503);
        this->buttoncsc->Name = L"buttoncsc";
        this->buttoncsc->Size = System::Drawing::Size(71,
44);

        this->buttoncsc->TabIndex = 249;
        this->buttoncsc->Text = L"csc";
        this->buttoncsc->UseVisualStyleBackColor = true;
        this->buttoncsc->Click += gcnew
System::EventHandler(this, &engineering::buttoncsc_Click);
        //
        // buttonsec
        //
        this->buttonsec->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->buttonsec->Location =
System::Drawing::Point(17, 453);
        this->buttonsec->Name = L"buttonsec";
        this->buttonsec->Size = System::Drawing::Size(71,
44);

        this->buttonsec->TabIndex = 248;
        this->buttonsec->Text = L"sec";
        this->buttonsec->UseVisualStyleBackColor = true;
        this->buttonsec->Click += gcnew
System::EventHandler(this, &engineering::buttonsec_Click);
        //
        // buttoncot
        //
        this->buttoncot->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->buttoncot->Location =
System::Drawing::Point(17, 402);
        this->buttoncot->Name = L"buttoncot";
        this->buttoncot->Size = System::Drawing::Size(71,
44);

        this->buttoncot->TabIndex = 247;
        this->buttoncot->Text = L"cot";

```

```

        this->buttoncot->UseVisualStyleBackColor = true;
        this->buttoncot->Click += gcnew
System::EventHandler(this, &engineering::buttoncot_Click);
        //
        // buttontan
        //
        this->buttontan->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->buttontan->Location =
System::Drawing::Point(17, 352);
        this->buttontan->Name = L"buttontan";
        this->buttontan->Size = System::Drawing::Size(71,
44);

        this->buttontan->TabIndex = 246;
        this->buttontan->Text = L"tan";
        this->buttontan->UseVisualStyleBackColor = true;
        this->buttontan->Click += gcnew
System::EventHandler(this, &engineering::buttontan_Click);
        //
        // buttoncos
        //
        this->buttoncos->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->buttoncos->Location =
System::Drawing::Point(17, 302);
        this->buttoncos->Name = L"buttoncos";
        this->buttoncos->Size = System::Drawing::Size(71,
44);

        this->buttoncos->TabIndex = 245;
        this->buttoncos->Text = L"cos";
        this->buttoncos->UseVisualStyleBackColor = true;
        this->buttoncos->Click += gcnew
System::EventHandler(this, &engineering::buttoncos_Click);
        //
        // buttonsin
        //
        this->buttonsin->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->buttonsin->Location =
System::Drawing::Point(17, 253);
        this->buttonsin->Name = L"buttonsin";
        this->buttonsin->Size = System::Drawing::Size(71,
44);

        this->buttonsin->TabIndex = 244;

```

```

        this->buttonsin->Text = L"sin";
        this->buttonsin->UseVisualStyleBackColor = true;
        this->buttonsin->Click += gcnew
System::EventHandler(this, &engineering::buttonsin_Click);
        //
        // Mminus
        //
        this->Mminus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 9.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->Mminus->Location = System::Drawing::Point(262,
137);

        this->Mminus->Name = L"Mminus";
        this->Mminus->Size = System::Drawing::Size(55, 30);
        this->Mminus->TabIndex = 243;
        this->Mminus->Text = L"M-";
        this->Mminus->UseVisualStyleBackColor = true;
        this->Mminus->Click += gcnew
System::EventHandler(this, &engineering::Mminus_Click);
        //
        // Mplus
        //
        this->Mplus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 9.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->Mplus->Location = System::Drawing::Point(181,
137);

        this->Mplus->Name = L"Mplus";
        this->Mplus->Size = System::Drawing::Size(55, 30);
        this->Mplus->TabIndex = 242;
        this->Mplus->Text = L"M+";
        this->Mplus->UseVisualStyleBackColor = true;
        this->Mplus->Click += gcnew
System::EventHandler(this, &engineering::Mplus_Click);
        //
        // MC
        //
        this->MC->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 9.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->MC->Location = System::Drawing::Point(17,
137);

        this->MC->Name = L"MC";
        this->MC->Size = System::Drawing::Size(55, 30);
        this->MC->TabIndex = 241;
        this->MC->Text = L"MC";
        this->MC->UseVisualStyleBackColor = true;

```

```

        this->MC->Click += gcnew System::EventHandler(this,
&engineering::MC_Click);
        //
        // MR
        //
        this->MR->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 9.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->MR->Location = System::Drawing::Point(99,
137);

        this->MR->Name = L"MR";
        this->MR->Size = System::Drawing::Size(54, 30);
        this->MR->TabIndex = 240;
        this->MR->Text = L"MR";
        this->MR->UseVisualStyleBackColor = true;
        this->MR->Click += gcnew System::EventHandler(this,
&engineering::MR_Click);
        //
        // MS
        //
        this->MS->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 9.75F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->MS->Location = System::Drawing::Point(340,
137);

        this->MS->Name = L"MS";
        this->MS->Size = System::Drawing::Size(55, 30);
        this->MS->TabIndex = 239;
        this->MS->Text = L"MS";
        this->MS->UseVisualStyleBackColor = true;
        this->MS->Click += gcnew System::EventHandler(this,
&engineering::MS_Click);
        //
        // labell
        //
        this->labell->BackColor =
System::Drawing::SystemColors::Control;
        this->labell->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->labell->ForeColor =
System::Drawing::Color::DarkGray;
        this->labell->Location = System::Drawing::Point(16,
69);

        this->labell->MaximumSize =
System::Drawing::Size(500, 90);
        this->labell->Name = L"labell";
        this->labell->Size = System::Drawing::Size(382, 24);
        this->labell->TabIndex = 238;

```

```

        this->label1->TextAlign =
System::Drawing::ContentAlignment::MiddleRight;
        //
        // textBox1
        //
        this->textBox1->Font = (gcnew
System::Drawing::Font(L"Leelawadee", 15.75F));
        this->textBox1->Location =
System::Drawing::Point(17, 96);
        this->textBox1->Name = L"textBox1";
        this->textBox1->Size = System::Drawing::Size(378,
33);

        this->textBox1->TabIndex = 237;
        this->textBox1->Text = L"0";
        this->textBox1->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        //
        // change
        //
        this->change->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->change->Location = System::Drawing::Point(247,
504);

        this->change->Name = L"change";
        this->change->Size = System::Drawing::Size(71, 44);
        this->change->TabIndex = 236;
        this->change->Text = L"+";
        this->change->UseVisualStyleBackColor = true;
        this->change->Click += gcnew
System::EventHandler(this, &engineering::change_Click);
        //
        // erase2
        //
        this->erase2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->erase2->Location = System::Drawing::Point(212,
187);

        this->erase2->Name = L"erase2";
        this->erase2->Size = System::Drawing::Size(85, 55);
        this->erase2->TabIndex = 235;
        this->erase2->Text = L"CE";
        this->erase2->UseVisualStyleBackColor = true;
        this->erase2->Click += gcnew
System::EventHandler(this, &engineering::erase2_Click);
        //
        // erase1
        //
        this->erase1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->erase1->Location = System::Drawing::Point(115,
187);

        this->erase1->Name = L"erase1";
        this->erase1->Size = System::Drawing::Size(85, 55);

```

```

        this->erases->TabIndex = 234;
        this->erases->Text = L"C";
        this->erases->UseVisualStyleBackColor = true;
        this->erases->Click += gcnew
System::EventHandler(this, &engineering::erases_Click);
        //
        // clear
        //
        this->clear->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14,
System::Drawing::FontStyle::Bold));
        this->clear->Location = System::Drawing::Point(17,
187);

        this->clear->Name = L"clear";
        this->clear->Size = System::Drawing::Size(85, 55);
        this->clear->TabIndex = 233;
        this->clear->Text = L"←";
        this->clear->UseVisualStyleBackColor = true;
        this->clear->Click += gcnew
System::EventHandler(this, &engineering::clear_Click);
        //
        // division
        //
        this->division->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->division->Location =
System::Drawing::Point(324, 504);
        this->division->Name = L"division";
        this->division->Size = System::Drawing::Size(71,
44);

        this->division->TabIndex = 232;
        this->division->Text = L"÷";
        this->division->UseVisualStyleBackColor = true;
        this->division->Click += gcnew
System::EventHandler(this, &engineering::arithmetic);
        //
        // multiplication
        //
        this->multiplication->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->multiplication->Location =
System::Drawing::Point(324, 453);
        this->multiplication->Name = L"multiplication";
        this->multiplication->Size =
System::Drawing::Size(71, 44);
        this->multiplication->TabIndex = 231;
        this->multiplication->Text = L"×";
        this->multiplication->UseVisualStyleBackColor =
true;

        this->multiplication->Click += gcnew
System::EventHandler(this, &engineering::arithmetic);
        //
        // minus

```

```

        //
        this->minus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->minus->Location = System::Drawing::Point(324,
403);

        this->minus->Name = L"minus";
        this->minus->Size = System::Drawing::Size(71, 44);
        this->minus->TabIndex = 230;
        this->minus->Text = L"-";
        this->minus->UseVisualStyleBackColor = true;
        this->minus->Click += gcnew
System::EventHandler(this, &engineering::arithmetic);
        //
        // plus
        //
        this->plus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->plus->Location = System::Drawing::Point(324,
353);

        this->plus->Name = L"plus";
        this->plus->Size = System::Drawing::Size(71, 44);
        this->plus->TabIndex = 229;
        this->plus->Text = L"+";
        this->plus->UseVisualStyleBackColor = true;
        this->plus->Click += gcnew
System::EventHandler(this, &engineering::arithmetic);
        //
        // equalsign
        //
        this->equalsign->BackColor =
System::Drawing::SystemColors::GradientActiveCaption;
        this->equalsign->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->equalsign->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->equalsign->Location =
System::Drawing::Point(310, 187);
        this->equalsign->Name = L"equalsign";
        this->equalsign->Size = System::Drawing::Size(85,
55);

        this->equalsign->TabIndex = 228;
        this->equalsign->Text = L="";
        this->equalsign->UseVisualStyleBackColor = false;
        this->equalsign->Click += gcnew
System::EventHandler(this, &engineering::equalsign_Click);
        //
        // point
        //
        this->point->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->point->Location = System::Drawing::Point(93,
503);

        this->point->Name = L"point";

```

```

        this->point->Size = System::Drawing::Size(71, 44);
        this->point->TabIndex = 227;
        this->point->Text = L",";
        this->point->UseVisualStyleBackColor = true;
        this->point->Click += gcnew
System::EventHandler(this, &engineering::point_Click);
        //
        // button9
        //
        this->button9->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button9->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button9->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button9->Location =
System::Drawing::Point(247, 354);
        this->button9->Name = L"button9";
        this->button9->Size = System::Drawing::Size(71, 44);
        this->button9->TabIndex = 226;
        this->button9->Text = L"9";
        this->button9->UseVisualStyleBackColor = false;
        this->button9->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button8
        //
        this->button8->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button8->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button8->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button8->Location =
System::Drawing::Point(170, 354);
        this->button8->Name = L"button8";
        this->button8->Size = System::Drawing::Size(71, 44);
        this->button8->TabIndex = 225;
        this->button8->Text = L"8";
        this->button8->UseVisualStyleBackColor = false;
        this->button8->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button7
        //
        this->button7->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button7->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button7->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button7->Location = System::Drawing::Point(93,
354);

```

```

        this->button7->Name = L"button7";
        this->button7->Size = System::Drawing::Size(71, 44);
        this->button7->TabIndex = 224;
        this->button7->Text = L"7";
        this->button7->UseVisualStyleBackColor = false;
        this->button7->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button6
        //
        this->button6->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button6->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button6->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button6->Location =
System::Drawing::Point(247, 404);
        this->button6->Name = L"button6";
        this->button6->Size = System::Drawing::Size(71, 44);
        this->button6->TabIndex = 223;
        this->button6->Text = L"6";
        this->button6->UseVisualStyleBackColor = false;
        this->button6->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button5
        //
        this->button5->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button5->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button5->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button5->Location =
System::Drawing::Point(170, 403);
        this->button5->Name = L"button5";
        this->button5->Size = System::Drawing::Size(71, 44);
        this->button5->TabIndex = 222;
        this->button5->Text = L"5";
        this->button5->UseVisualStyleBackColor = false;
        this->button5->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button4
        //
        this->button4->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button4->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button4->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));

```

```

        this->button4->Location = System::Drawing::Point(93,
403);
        this->button4->Name = L"button4";
        this->button4->Size = System::Drawing::Size(71, 44);
        this->button4->TabIndex = 221;
        this->button4->Text = L"4";
        this->button4->UseVisualStyleBackColor = false;
        this->button4->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button3
        //
        this->button3->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button3->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button3->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button3->Location =
System::Drawing::Point(247, 454);
        this->button3->Name = L"button3";
        this->button3->Size = System::Drawing::Size(71, 44);
        this->button3->TabIndex = 220;
        this->button3->Text = L"3";
        this->button3->UseVisualStyleBackColor = false;
        this->button3->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button2
        //
        this->button2->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button2->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button2->Location =
System::Drawing::Point(170, 453);
        this->button2->Name = L"button2";
        this->button2->Size = System::Drawing::Size(71, 44);
        this->button2->TabIndex = 219;
        this->button2->Text = L"2";
        this->button2->UseVisualStyleBackColor = false;
        this->button2->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button1
        //
        this->button1->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button1->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;

```

```

        this->button1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button1->Location = System::Drawing::Point(93,
453);

        this->button1->Name = L"button1";
        this->button1->Size = System::Drawing::Size(71, 44);
        this->button1->TabIndex = 218;
        this->button1->Text = L"1";
        this->button1->UseVisualStyleBackColor = false;
        this->button1->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // button0
        //
        this->button0->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button0->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button0->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button0->Location =
System::Drawing::Point(170, 504);
        this->button0->Name = L"button0";
        this->button0->Size = System::Drawing::Size(71, 44);
        this->button0->TabIndex = 217;
        this->button0->Text = L"0";
        this->button0->UseVisualStyleBackColor = false;
        this->button0->Click += gcnew
System::EventHandler(this, &engineering::numberbuttons);
        //
        // buttonordinary1
        //
        this->buttonordinary1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->buttonordinary1->Location =
System::Drawing::Point(17, 12);
        this->buttonordinary1->Name = L"buttonordinary1";
        this->buttonordinary1->Size =
System::Drawing::Size(180, 30);
        this->buttonordinary1->TabIndex = 216;
        this->buttonordinary1->Text = L"Режим \"Обычный\"";
        this->buttonordinary1->UseVisualStyleBackColor =
true;

        this->buttonordinary1->Click += gcnew
System::EventHandler(this, &engineering::buttonordinary1_Click);
        //
        // engineering
        //
        this->AutoScaleDimensions =
System::Drawing::SizeF(6, 13);
        this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(414, 561);

```

```

        this->Controls->Add(this->numberpi);
        this->Controls->Add(this->exhibitor);
        this->Controls->Add(this->natlogarithm);
        this->Controls->Add(this->logarithm);
        this->Controls->Add(this->degreey);
        this->Controls->Add(this->buttoncsc);
        this->Controls->Add(this->buttonsec);
        this->Controls->Add(this->buttoncot);
        this->Controls->Add(this->buttontan);
        this->Controls->Add(this->buttoncos);
        this->Controls->Add(this->buttonsin);
        this->Controls->Add(this->Mminus);
        this->Controls->Add(this->Mplus);
        this->Controls->Add(this->MC);
        this->Controls->Add(this->MR);
        this->Controls->Add(this->MS);
        this->Controls->Add(this->label1);
        this->Controls->Add(this->textBox1);
        this->Controls->Add(this->change);
        this->Controls->Add(this->erase2);
        this->Controls->Add(this->erasel);
        this->Controls->Add(this->clear);
        this->Controls->Add(this->division);
        this->Controls->Add(this->multiplication);
        this->Controls->Add(this->minus);
        this->Controls->Add(this->plus);
        this->Controls->Add(this->equalsign);
        this->Controls->Add(this->point);
        this->Controls->Add(this->button9);
        this->Controls->Add(this->button8);
        this->Controls->Add(this->button7);
        this->Controls->Add(this->button6);
        this->Controls->Add(this->button5);
        this->Controls->Add(this->button4);
        this->Controls->Add(this->button3);
        this->Controls->Add(this->button2);
        this->Controls->Add(this->button1);
        this->Controls->Add(this->button0);
        this->Controls->Add(this->buttonordinary1);
        this->Name = L"engineering";
        this->StartPosition =
System::Windows::Forms::FormStartPosition::CenterScreen;
        this->Text = L"engineering";
        this->ResumeLayout(false);
        this->PerformLayout();

    }
#pragma endregion

    double firstNum = Double::NaN;
    double secondNum = Double::NaN;
    double result = Double::NaN;
    double memory = 0;

```

```

String^ currentOper;
bool isCalculated = false;
bool flag = false;
double a;
double pi;
double e;

void calculate()
{
    if (!Double::IsNaN(firstNum) &&
!Double::IsNaN(secondNum))
    {
        if (currentOper == L"+")
        {
            result = firstNum + secondNum;
        }
        else if (currentOper == L"-")
        {
            result = firstNum - secondNum;
        }
        else if (currentOper == L"x")
        {
            result = firstNum * secondNum;
        }
        else if (currentOper == L"÷")
        {
            if (secondNum != 0)
            {
                result = firstNum / secondNum;
            }
            else
            {
                MessageBox::Show("Деление на ноль невозможно", "Error",
MessageBoxButtons::OK, MessageBoxIcon::Error);
                return;
            }
        }

        else if (currentOper == L"^")
        {
            result = Math::Pow(firstNum, secondNum);
        }

        label1->Text += String::Format(" {0} =",
secondNum);

        textBox1->Text =
System::Convert::ToString(result);
        isCalculated = true;
    }
}

```

```

        private: System::Void buttonordinary1_Click(System::Object^
sender, System::EventArgs^ e)
        {
            Owner->Show();
            this->Close();
        }

private: System::Void arithmetic(System::Object^ sender,
System::EventArgs^ e)
{
    Button^ operators = safe_cast<Button^>(sender);
    String^ outNumber;

    // second operation
    if (label1->Text != "")
    {
        // second number
        if (!isCalculated)
        {
            secondNum = Double::Parse(textBox1->Text);
        }

        if (Double::IsNaN(result))
        {
            calculate();
        }
        else
        {
            firstNum = result;
            if (!isCalculated)
            {
                calculate();
            }
        }
        outNumber = System::Convert::ToString(result);
    }
    else // first operation
    {
        firstNum = Double::Parse(textBox1->Text);
        outNumber = System::Convert::ToString(firstNum);
    }

    currentOper = operators->Text;
    textBox1->Text = "";
    label1->Text = outNumber + " " + currentOper;
}

private: System::Void equalsign_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!isCalculated)
    {
        //firstNum = Double::IsNaN(result) ? firstNum : result;
        secondNum = Double::Parse(textBox1->Text);
    }
}

```

```

    }
    else
    {
        //firstNum = result;
        labell1->Text = String::Format("{0} {1}", firstNum,
currentOper);
    }

    firstNum = Double::IsNaN(result) ? firstNum : result;
    //secondNum = isCalculated ? secondNum :
Double::Parse(CalculationBox->Text);

    calculate();
}

private: System::Void clear_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text->Length > 0)
    {
        textBox1->Text = textBox1->Text->Remove(textBox1->Text-
>Length - 1, 1);
    }
}

private: System::Void erasel_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = "0";
    labell1->Text = "";
    result = Double::NaN;
    firstNum = Double::NaN;
    secondNum = Double::NaN;
    isCalculated = false;
    currentOper = "";
}

private: System::Void erase2_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = "";
}

private: System::Void point_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!textBox1->Text->Contains(","))
    {
        textBox1->Text += ",";
    }
}

private: System::Void change_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text->Contains("-"))
    {

```

```

        textBox1->Text = textBox1->Text->Remove(0, 1);
    }
    else
    {
        textBox1->Text = "-" + textBox1->Text;
    }
}
private: System::Void MS_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory = Double::Parse(textBox1->Text);
    textBox1->Text = System::Convert::ToString(memory);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void MC_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory = 0;
    MC->Enabled = false;
    MR->Enabled = false;
}
private: System::Void MR_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = System::Convert::ToString(memory);
    isCalculated = true;
}
private: System::Void Mplus_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory += Double::Parse(textBox1->Text);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void Mminus_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory -= Double::Parse(textBox1->Text);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void numberpi_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text += "3,1415926535";
}
private: System::Void exhibitor_Click(System::Object^ sender,
System::EventArgs^ e)
{

```

```

        textBox1->Text += "2,7182818284";
    }
private: System::Void degreey_Click(System::Object^ sender,
System::EventArgs^ e)
{
    Button^ operators = safe_cast<Button^>(sender);
    String^ outNumber;

    // second operation
    if (label1->Text != "")
    {
        // second number
        if (!isCalculated)
        {
            secondNum = Double::Parse(textBox1->Text);
        }

        if (Double::IsNaN(result))
        {
            calculate();
        }
        else
        {
            firstNum = result;
            if (!isCalculated)
            {
                calculate();
            }
        }
        outNumber = System::Convert::ToString(result);
    }
    else // first operation
    {
        firstNum = Double::Parse(textBox1->Text);
        outNumber = System::Convert::ToString(firstNum);
    }

    currentOper = operators->Text;
    textBox1->Text = "";
    label1->Text = outNumber + " " + currentOper;
}

private: System::Void logarithm_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    label1->Text = System::Convert::ToString("log " + "(" +
(textBox1->Text) + ")");
    a = Math::Log10(a);
    textBox1->Text = System::Convert::ToString(a);
}

private: System::Void natlogarithm_Click(System::Object^ sender,
System::EventArgs^ e)

```

```

{
    a = Double::Parse(textBox1->Text);
    labell->Text = System::Convert::ToString("ln " + "(" +
(textBox1->Text) + ")");
    a = Math::Log(a);
    textBox1->Text = System::Convert::ToString(a);
}
private: System::Void buttonsln_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    labell->Text = System::Convert::ToString("sind " + "(" +
(textBox1->Text) + ")");
    a = Math::Sin(a);
    textBox1->Text = System::Convert::ToString(a);
}
private: System::Void buttoncos_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    labell->Text = System::Convert::ToString("cosd " + "(" +
(textBox1->Text) + ")");
    a = Math::Cos(a);
    textBox1->Text = System::Convert::ToString(a);
}
private: System::Void buttontan_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    labell->Text = System::Convert::ToString("tand " + "(" +
(textBox1->Text) + ")");
    a = Math::Tan(a);
    textBox1->Text = System::Convert::ToString(a);
}
private: System::Void buttoncot_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    labell->Text = System::Convert::ToString("cotd " + "(" +
(textBox1->Text) + ")");
    a = (Math::Cos(a)) / (Math::Sin(a));
    textBox1->Text = System::Convert::ToString(a);
}
private: System::Void buttonsec_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    labell->Text = System::Convert::ToString("cotd " + "(" +
(textBox1->Text) + ")");
    a = (1) / (Math::Cos(a));
    textBox1->Text = System::Convert::ToString(a);
}

```

```

private: System::Void buttoncsc_Click(System::Object^ sender,
System::EventArgs^ e)
{
    a = Double::Parse(textBox1->Text);
    labell1->Text = System::Convert::ToString("cotd " + "(" +
(textBox1->Text) + ")");
    a = (1) / (Math::Sin(a));
    textBox1->Text = System::Convert::ToString(a);
}
private: System::Void numberbuttons(System::Object^ sender,
System::EventArgs^ e)
{
    Button^ numbers = safe_cast<Button^>(sender);

    if (textBox1->Text == "0")
    {
        textBox1->Text = numbers->Text;
    }
    else if (textBox1->Text->Length <= 31)
    {
        textBox1->Text += numbers->Text;
    }
    isCalculated = false;
}
};
}

```

Исходный файл «programmatic.cpp»

```
#include "programmatic.h"
```

Заголовочный файл «programmatic.h»

```
#pragma once
```

```
namespace CWCalculator {
```

```
    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;
```

```
    /// <summary>
    /// Сводка для programmatic
    /// </summary>
```

```
    public ref class programmatic : public
System::Windows::Forms::Form
```

```
    {
    public:
        programmatic(void)
        {
            InitializeComponent();
            //
            //TODO: добавьте код конструктора
            //
            MC->Enabled = false;
            MR->Enabled = false;
            Mplus->Enabled = false;
            Mminus->Enabled = false;
        }
    }
```

```
    protected:
        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        ~programmatic()
        {
            if (components)
            {
                delete components;
            }
        }
    }
```

```
private: System::Windows::Forms::TextBox^ textBox5;
```

```
protected:
```

```
private: System::Windows::Forms::TextBox^ textBox4;
```

```
private: System::Windows::Forms::TextBox^ textBox3;
```

```
private: System::Windows::Forms::TextBox^ textBox2;
```

```

private: System::Windows::Forms::Label^ label1;
private: System::Windows::Forms::Button^ buttonordinary1;
private: System::Windows::Forms::Label^ label5;
private: System::Windows::Forms::Label^ label4;
private: System::Windows::Forms::Label^ label3;
private: System::Windows::Forms::Button^ Mminus;
private: System::Windows::Forms::Button^ Mplus;
private: System::Windows::Forms::Button^ MC;
private: System::Windows::Forms::Button^ MR;
private: System::Windows::Forms::Button^ MS;
private: System::Windows::Forms::Label^ label2;
private: System::Windows::Forms::TextBox^ textBox1;
private: System::Windows::Forms::Button^ change;
private: System::Windows::Forms::Button^ erase2;
private: System::Windows::Forms::Button^ erasel;
private: System::Windows::Forms::Button^ clear;
private: System::Windows::Forms::Button^ division;
private: System::Windows::Forms::Button^ multiplication;
private: System::Windows::Forms::Button^ minus;
private: System::Windows::Forms::Button^ plus;
private: System::Windows::Forms::Button^ equalsign;
private: System::Windows::Forms::Button^ point;
private: System::Windows::Forms::Button^ button9;
private: System::Windows::Forms::Button^ button8;
private: System::Windows::Forms::Button^ button7;
private: System::Windows::Forms::Button^ button6;
private: System::Windows::Forms::Button^ button5;
private: System::Windows::Forms::Button^ button4;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button1;
private: System::Windows::Forms::Button^ button0;

private:
    /// <summary>
    /// Обязательная переменная конструктора.
    /// </summary>
    System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Требуемый метод для поддержки конструктора — не
изменяйте
    /// содержимое этого метода с помощью редактора кода.
    /// </summary>
    void InitializeComponent(void)
    {
        this->textBox5 = (gcnew
System::Windows::Forms::TextBox());
        this->textBox4 = (gcnew
System::Windows::Forms::TextBox());
        this->textBox3 = (gcnew
System::Windows::Forms::TextBox());

```

```

        this->textBox2 = (gcnew
System::Windows::Forms::TextBox());
        this->label1 = (gcnew
System::Windows::Forms::Label());
        this->buttonordinary1 = (gcnew
System::Windows::Forms::Button());
        this->label5 = (gcnew
System::Windows::Forms::Label());
        this->label4 = (gcnew
System::Windows::Forms::Label());
        this->label3 = (gcnew
System::Windows::Forms::Label());
        this->Mminus = (gcnew
System::Windows::Forms::Button());
        this->Mplus = (gcnew
System::Windows::Forms::Button());
        this->MC = (gcnew System::Windows::Forms::Button());
        this->MR = (gcnew System::Windows::Forms::Button());
        this->MS = (gcnew System::Windows::Forms::Button());
        this->label2 = (gcnew
System::Windows::Forms::Label());
        this->textBox1 = (gcnew
System::Windows::Forms::TextBox());
        this->change = (gcnew
System::Windows::Forms::Button());
        this->erase2 = (gcnew
System::Windows::Forms::Button());
        this->erasel = (gcnew
System::Windows::Forms::Button());
        this->clear = (gcnew
System::Windows::Forms::Button());
        this->division = (gcnew
System::Windows::Forms::Button());
        this->multiplication = (gcnew
System::Windows::Forms::Button());
        this->minus = (gcnew
System::Windows::Forms::Button());
        this->plus = (gcnew
System::Windows::Forms::Button());
        this->>equalsign = (gcnew
System::Windows::Forms::Button());
        this->point = (gcnew
System::Windows::Forms::Button());
        this->button9 = (gcnew
System::Windows::Forms::Button());
        this->button8 = (gcnew
System::Windows::Forms::Button());
        this->button7 = (gcnew
System::Windows::Forms::Button());
        this->button6 = (gcnew
System::Windows::Forms::Button());
        this->button5 = (gcnew
System::Windows::Forms::Button());

```

```

        this->button4 = (gcnew
System::Windows::Forms::Button());
        this->button3 = (gcnew
System::Windows::Forms::Button());
        this->button2 = (gcnew
System::Windows::Forms::Button());
        this->button1 = (gcnew
System::Windows::Forms::Button());
        this->button0 = (gcnew
System::Windows::Forms::Button());
        this->SuspendLayout();
        //
        // textBox5
        //
        this->textBox5->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 8.25F,
System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->textBox5->ForeColor =
System::Drawing::SystemColors::WindowFrame;
        this->textBox5->Location =
System::Drawing::Point(79, 135);
        this->textBox5->Name = L"textBox5";
        this->textBox5->Size = System::Drawing::Size(320,
20);

        this->textBox5->TabIndex = 137;
        this->textBox5->Text = L"0";
        this->textBox5->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        //
        // textBox4
        //
        this->textBox4->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 8.25F,
System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->textBox4->ForeColor =
System::Drawing::SystemColors::WindowFrame;
        this->textBox4->Location =
System::Drawing::Point(79, 111);
        this->textBox4->Name = L"textBox4";
        this->textBox4->Size = System::Drawing::Size(320,
20);

        this->textBox4->TabIndex = 136;
        this->textBox4->Text = L"0";
        this->textBox4->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        //
        // textBox3
        //

```

```

        this->textBox3->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 8.25F,
System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->textBox3->ForeColor =
System::Drawing::SystemColors::WindowFrame;
        this->textBox3->Location =
System::Drawing::Point(79, 87);
        this->textBox3->Name = L"textBox3";
        this->textBox3->Size = System::Drawing::Size(320,
20);

        this->textBox3->TabIndex = 135;
        this->textBox3->Text = L"0";
        this->textBox3->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        //
        // textBox2
        //
        this->textBox2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 8.25F,
System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->textBox2->ForeColor =
System::Drawing::SystemColors::WindowFrame;
        this->textBox2->Location =
System::Drawing::Point(79, 63);
        this->textBox2->Name = L"textBox2";
        this->textBox2->Size = System::Drawing::Size(320,
20);

        this->textBox2->TabIndex = 134;
        this->textBox2->Text = L"0";
        this->textBox2->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        //
        // label1
        //
        this->label1->BackColor =
System::Drawing::SystemColors::Control;
        this->label1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label1->ForeColor =
System::Drawing::Color::DarkGray;
        this->label1->Location = System::Drawing::Point(17,
163);

        this->label1->MaximumSize =
System::Drawing::Size(500, 90);
        this->label1->Name = L"label1";
        this->label1->Size = System::Drawing::Size(382, 24);
        this->label1->TabIndex = 133;
        this->label1->TextAlign =
System::Drawing::ContentAlignment::MiddleRight;

```

```

        //
        // buttonordinary1
        //
        this->buttonordinary1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->buttonordinary1->Location =
System::Drawing::Point(17, 14);
        this->buttonordinary1->Name = L"buttonordinary1";
        this->buttonordinary1->Size =
System::Drawing::Size(180, 30);
        this->buttonordinary1->TabIndex = 132;
        this->buttonordinary1->Text = L"Режим \"ОБЫЧНЫЙ\"";
        this->buttonordinary1->UseVisualStyleBackColor =
true;
        this->buttonordinary1->Click += gcnew
System::EventHandler(this, &programmatic::buttonordinary1_Click);
        //
        // label5
        //
        this->label5->BackColor =
System::Drawing::SystemColors::Control;
        this->label5->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label5->ForeColor =
System::Drawing::Color::DimGray;
        this->label5->Location = System::Drawing::Point(16,
132);
        this->label5->MaximumSize =
System::Drawing::Size(500, 90);
        this->label5->Name = L"label5";
        this->label5->Size = System::Drawing::Size(73, 24);
        this->label5->TabIndex = 131;
        this->label5->Text = L"BIN";
        this->label5->TextAlign =
System::Drawing::ContentAlignment::MiddleLeft;
        //
        // label4
        //
        this->label4->BackColor =
System::Drawing::SystemColors::Control;
        this->label4->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label4->ForeColor =
System::Drawing::Color::DimGray;
        this->label4->Location = System::Drawing::Point(16,
108);
        this->label4->MaximumSize =
System::Drawing::Size(500, 90);
        this->label4->Name = L"label4";
        this->label4->Size = System::Drawing::Size(73, 24);
        this->label4->TabIndex = 130;
        this->label4->Text = L"OCT";

```

```

        this->label4->TextAlign =
System::Drawing::ContentAlignment::MiddleLeft;
        //
        // label3
        //
        this->label3->BackColor =
System::Drawing::SystemColors::Control;
        this->label3->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label3->ForeColor =
System::Drawing::Color::Red;
        this->label3->Location = System::Drawing::Point(16,
84);
        this->label3->MaximumSize =
System::Drawing::Size(500, 90);
        this->label3->Name = L"label3";
        this->label3->Size = System::Drawing::Size(73, 24);
        this->label3->TabIndex = 129;
        this->label3->Text = L"DEC";
        this->label3->TextAlign =
System::Drawing::ContentAlignment::MiddleLeft;
        //
        // Mminus
        //
        this->Mminus->BackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->Cursor =
System::Windows::Forms::Cursors::Default;
        this->Mminus->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatAppearance->BorderSize = 0;
        this->Mminus->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->Mminus->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->Mminus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->Mminus->Location = System::Drawing::Point(249,
237);
        this->Mminus->Name = L"Mminus";
        this->Mminus->Size = System::Drawing::Size(72, 50);
        this->Mminus->TabIndex = 128;
        this->Mminus->Text = L"M-";
        this->Mminus->UseVisualStyleBackColor = false;
        this->Mminus->Click += gcnew
System::EventHandler(this, &programmatic::Mminus_Click);

```

```

        //
        // Mplus
        //
        this->Mplus->BackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->Cursor =
System::Windows::Forms::Cursors::Default;
        this->Mplus->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatAppearance->BorderSize = 0;
        this->Mplus->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->Mplus->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->Mplus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->Mplus->Location = System::Drawing::Point(171,
237);

        this->Mplus->Name = L"Mplus";
        this->Mplus->Size = System::Drawing::Size(72, 50);
        this->Mplus->TabIndex = 127;
        this->Mplus->Text = L"M+";
        this->Mplus->UseVisualStyleBackColor = false;
        this->Mplus->Click += gcnew
System::EventHandler(this, &programmatic::Mplus_Click);
        //
        // MC
        //
        this->MC->BackColor =
System::Drawing::SystemColors::Control;
        this->MC->Cursor =
System::Windows::Forms::Cursors::Default;
        this->MC->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatAppearance->BorderSize = 0;
        this->MC->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->MC->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->MC->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,

```

```

System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
    static_cast<System::Byte>(204)));
    this->MC->Location = System::Drawing::Point(17,
237);

    this->MC->Name = L"MC";
    this->MC->Size = System::Drawing::Size(72, 50);
    this->MC->TabIndex = 126;
    this->MC->Text = L"MC";
    this->MC->UseVisualStyleBackColor = false;
    this->MC->Click += gcnew System::EventHandler(this,
&programmatic::MC_Click);
    //
    // MR
    //
    this->MR->BackColor =
System::Drawing::SystemColors::Control;
    this->MR->Cursor =
System::Windows::Forms::Cursors::Default;
    this->MR->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;
    this->MR->FlatAppearance->BorderSize = 0;
    this->MR->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
    this->MR->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
    this->MR->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
    this->MR->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
    this->MR->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
    static_cast<System::Byte>(204)));
    this->MR->Location = System::Drawing::Point(94,
237);

    this->MR->Name = L"MR";
    this->MR->Size = System::Drawing::Size(72, 50);
    this->MR->TabIndex = 125;
    this->MR->Text = L"MR";
    this->MR->UseVisualStyleBackColor = false;
    this->MR->Click += gcnew System::EventHandler(this,
&programmatic::MR_Click);
    //
    // MS
    //
    this->MS->BackColor =
System::Drawing::SystemColors::Control;
    this->MS->Cursor =
System::Windows::Forms::Cursors::Default;
    this->MS->FlatAppearance->BorderColor =
System::Drawing::SystemColors::Control;

```

```

        this->MS->FlatAppearance->BorderSize = 0;
        this->MS->FlatAppearance->CheckedBackColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatAppearance->MouseDownBackColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatAppearance->MouseOverBackColor =
System::Drawing::SystemColors::Control;
        this->MS->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->MS->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 11.25F,
System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->MS->Location = System::Drawing::Point(327,
237);

        this->MS->Name = L"MS";
        this->MS->Size = System::Drawing::Size(72, 50);
        this->MS->TabIndex = 124;
        this->MS->Text = L"MS";
        this->MS->UseVisualStyleBackColor = false;
        this->MS->Click += gcnew System::EventHandler(this,
&programmatic::MS_Click);
        //
        // label2
        //
        this->label2->BackColor =
System::Drawing::SystemColors::Control;
        this->label2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 10));
        this->label2->ForeColor =
System::Drawing::Color::DimGray;
        this->label2->Location = System::Drawing::Point(16,
60);

        this->label2->MaximumSize =
System::Drawing::Size(500, 90);
        this->label2->Name = L"label2";
        this->label2->Size = System::Drawing::Size(73, 24);
        this->label2->TabIndex = 123;
        this->label2->Text = L"HEX";
        this->label2->TextAlign =
System::Drawing::ContentAlignment::MiddleLeft;
        //
        // textBox1
        //
        this->textBox1->Font = (gcnew
System::Drawing::Font(L"Leelawadee", 15.75F));
        this->textBox1->Location =
System::Drawing::Point(16, 190);
        this->textBox1->Name = L"textBox1";
        this->textBox1->Size = System::Drawing::Size(382,
33);

        this->textBox1->TabIndex = 122;

```

```

        this->textBox1->Text = L"0";
        this->textBox1->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;
        //
        // change
        //
        this->change->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->change->Location = System::Drawing::Point(205,
502);

        this->change->Name = L"change";
        this->change->Size = System::Drawing::Size(87, 45);
        this->change->TabIndex = 121;
        this->change->Text = L"±";
        this->change->UseVisualStyleBackColor = true;
        this->change->Click += gcnew
System::EventHandler(this, &programmatic::change_Click);
        //
        // erase2
        //
        this->erase2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->erase2->Location = System::Drawing::Point(205,
293);

        this->erase2->Name = L"erase2";
        this->erase2->Size = System::Drawing::Size(87, 50);
        this->erase2->TabIndex = 120;
        this->erase2->Text = L"CE";
        this->erase2->UseVisualStyleBackColor = true;
        this->erase2->Click += gcnew
System::EventHandler(this, &programmatic::erase2_Click);
        //
        // erasel
        //
        this->erasel->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->erasel->Location = System::Drawing::Point(110,
293);

        this->erasel->Name = L"erasel";
        this->erasel->Size = System::Drawing::Size(87, 50);
        this->erasel->TabIndex = 119;
        this->erasel->Text = L"C";
        this->erasel->UseVisualStyleBackColor = true;
        this->erasel->Click += gcnew
System::EventHandler(this, &programmatic::erasel_Click);
        //
        // clear
        //
        this->clear->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14,
System::Drawing::FontStyle::Bold));
        this->clear->Location = System::Drawing::Point(16,
293);

```

```

        this->clear->Name = L"clear";
        this->clear->Size = System::Drawing::Size(87, 50);
        this->clear->TabIndex = 118;
        this->clear->Text = L"←";
        this->clear->UseVisualStyleBackColor = true;
        this->clear->Click += gcnew
System::EventHandler(this, &programmatic::clear_Click);
        //
        // division
        //
        this->division->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->division->Location =
System::Drawing::Point(299, 502);
        this->division->Name = L"division";
        this->division->Size = System::Drawing::Size(99,
45);

        this->division->TabIndex = 117;
        this->division->Text = L"÷";
        this->division->UseVisualStyleBackColor = true;
        this->division->Click += gcnew
System::EventHandler(this, &programmatic::arithmetic);
        //
        // multiplication
        //
        this->multiplication->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->multiplication->Location =
System::Drawing::Point(299, 451);
        this->multiplication->Name = L"multiplication";
        this->multiplication->Size =
System::Drawing::Size(99, 45);
        this->multiplication->TabIndex = 116;
        this->multiplication->Text = L"×";
        this->multiplication->UseVisualStyleBackColor =
true;

        this->multiplication->Click += gcnew
System::EventHandler(this, &programmatic::arithmetic);
        //
        // minus
        //
        this->minus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->minus->Location = System::Drawing::Point(299,
400);

        this->minus->Name = L"minus";
        this->minus->Size = System::Drawing::Size(99, 45);
        this->minus->TabIndex = 115;
        this->minus->Text = L"-";
        this->minus->UseVisualStyleBackColor = true;
        this->minus->Click += gcnew
System::EventHandler(this, &programmatic::arithmetic);
        //

```

```

        // plus
        //
        this->plus->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->plus->Location = System::Drawing::Point(299,
349);

        this->plus->Name = L"plus";
        this->plus->Size = System::Drawing::Size(99, 45);
        this->plus->TabIndex = 114;
        this->plus->Text = L"+";
        this->plus->UseVisualStyleBackColor = true;
        this->plus->Click += gcnew
System::EventHandler(this, &programmatic::arithmetic);
        //
        // equalsign
        //
        this->equalsign->BackColor =
System::Drawing::SystemColors::GradientActiveCaption;
        this->equalsign->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->equalsign->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->equalsign->Location =
System::Drawing::Point(299, 293);
        this->equalsign->Name = L"equalsign";
        this->equalsign->Size = System::Drawing::Size(99,
50);

        this->equalsign->TabIndex = 113;
        this->equalsign->Text = L="";
        this->equalsign->UseVisualStyleBackColor = false;
        this->equalsign->Click += gcnew
System::EventHandler(this, &programmatic::equalsign_Click);
        //
        // point
        //
        this->point->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->point->Location = System::Drawing::Point(16,
502);

        this->point->Name = L"point";
        this->point->Size = System::Drawing::Size(87, 45);
        this->point->TabIndex = 112;
        this->point->Text = L",";
        this->point->UseVisualStyleBackColor = true;
        this->point->Click += gcnew
System::EventHandler(this, &programmatic::point_Click);
        //
        // button9
        //
        this->button9->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button9->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;

```

```

        this->button9->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button9->Location =
System::Drawing::Point(205, 349);
        this->button9->Name = L"button9";
        this->button9->Size = System::Drawing::Size(87, 45);
        this->button9->TabIndex = 111;
        this->button9->Text = L"9";
        this->button9->UseVisualStyleBackColor = false;
        this->button9->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button8
        //
        this->button8->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button8->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button8->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button8->Location =
System::Drawing::Point(110, 349);
        this->button8->Name = L"button8";
        this->button8->Size = System::Drawing::Size(87, 45);
        this->button8->TabIndex = 110;
        this->button8->Text = L"8";
        this->button8->UseVisualStyleBackColor = false;
        this->button8->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button7
        //
        this->button7->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button7->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button7->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button7->Location = System::Drawing::Point(16,
349);
        this->button7->Name = L"button7";
        this->button7->Size = System::Drawing::Size(87, 45);
        this->button7->TabIndex = 109;
        this->button7->Text = L"7";
        this->button7->UseVisualStyleBackColor = false;
        this->button7->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button6
        //
        this->button6->BackColor =
System::Drawing::SystemColors::ScrollBar;

```

```

        this->button6->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button6->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button6->Location =
System::Drawing::Point(206, 400);
        this->button6->Name = L"button6";
        this->button6->Size = System::Drawing::Size(87, 45);
        this->button6->TabIndex = 108;
        this->button6->Text = L"6";
        this->button6->UseVisualStyleBackColor = false;
        this->button6->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button5
        //
        this->button5->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button5->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button5->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button5->Location =
System::Drawing::Point(110, 400);
        this->button5->Name = L"button5";
        this->button5->Size = System::Drawing::Size(87, 45);
        this->button5->TabIndex = 107;
        this->button5->Text = L"5";
        this->button5->UseVisualStyleBackColor = false;
        this->button5->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button4
        //
        this->button4->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button4->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button4->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button4->Location = System::Drawing::Point(16,
400);
        this->button4->Name = L"button4";
        this->button4->Size = System::Drawing::Size(87, 45);
        this->button4->TabIndex = 106;
        this->button4->Text = L"4";
        this->button4->UseVisualStyleBackColor = false;
        this->button4->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button3
        //

```

```

        this->button3->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button3->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button3->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button3->Location =
System::Drawing::Point(205, 451);
        this->button3->Name = L"button3";
        this->button3->Size = System::Drawing::Size(87, 45);
        this->button3->TabIndex = 105;
        this->button3->Text = L"3";
        this->button3->UseVisualStyleBackColor = false;
        this->button3->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button2
        //
        this->button2->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button2->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button2->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button2->Location =
System::Drawing::Point(110, 451);
        this->button2->Name = L"button2";
        this->button2->Size = System::Drawing::Size(87, 45);
        this->button2->TabIndex = 104;
        this->button2->Text = L"2";
        this->button2->UseVisualStyleBackColor = false;
        this->button2->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button1
        //
        this->button1->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button1->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button1->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button1->Location = System::Drawing::Point(16,
451);
        this->button1->Name = L"button1";
        this->button1->Size = System::Drawing::Size(87, 45);
        this->button1->TabIndex = 103;
        this->button1->Text = L"1";
        this->button1->UseVisualStyleBackColor = false;
        this->button1->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // button0

```

```

        //
        this->button0->BackColor =
System::Drawing::SystemColors::ScrollBar;
        this->button0->FlatStyle =
System::Windows::Forms::FlatStyle::Popup;
        this->button0->Font = (gcnew
System::Drawing::Font(L"Microsoft Sans Serif", 14));
        this->button0->Location =
System::Drawing::Point(109, 502);
        this->button0->Name = L"button0";
        this->button0->Size = System::Drawing::Size(87, 45);
        this->button0->TabIndex = 102;
        this->button0->Text = L"0";
        this->button0->UseVisualStyleBackColor = false;
        this->button0->Click += gcnew
System::EventHandler(this, &programmatic::numberbuttons);
        //
        // programmatic
        //
        this->AutoScaleDimensions =
System::Drawing::SizeF(6, 13);
        this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(414, 561);
        this->Controls->Add(this->textBox5);
        this->Controls->Add(this->textBox4);
        this->Controls->Add(this->textBox3);
        this->Controls->Add(this->textBox2);
        this->Controls->Add(this->label1);
        this->Controls->Add(this->buttonordinary1);
        this->Controls->Add(this->label5);
        this->Controls->Add(this->label4);
        this->Controls->Add(this->label3);
        this->Controls->Add(this->Mminus);
        this->Controls->Add(this->Mplus);
        this->Controls->Add(this->MC);
        this->Controls->Add(this->MR);
        this->Controls->Add(this->MS);
        this->Controls->Add(this->label2);
        this->Controls->Add(this->textBox1);
        this->Controls->Add(this->change);
        this->Controls->Add(this->erase2);
        this->Controls->Add(this->erasel);
        this->Controls->Add(this->clear);
        this->Controls->Add(this->division);
        this->Controls->Add(this->multiplication);
        this->Controls->Add(this->minus);
        this->Controls->Add(this->plus);
        this->Controls->Add(this->equalsign);
        this->Controls->Add(this->point);
        this->Controls->Add(this->button9);
        this->Controls->Add(this->button8);
        this->Controls->Add(this->button7);

```

```

        this->Controls->Add(this->button6);
        this->Controls->Add(this->button5);
        this->Controls->Add(this->button4);
        this->Controls->Add(this->button3);
        this->Controls->Add(this->button2);
        this->Controls->Add(this->button1);
        this->Controls->Add(this->button0);
        this->Name = L"programmatic";
        this->StartPosition =
System::Windows::Forms::FormStartPosition::CenterScreen;
        this->Text = L"programmatic";
        this->ResumeLayout(false);
        this->PerformLayout();

    }
#pragma endregion

    double firstNum = Double::NaN;
    double secondNum = Double::NaN;
    double result = Double::NaN;
    double memory = 0;
    String^ currentOper;
    bool isCalculated = false;
    bool flag = false;

    void calculate()
    {
        if (!Double::IsNaN(firstNum) &&
!Double::IsNaN(secondNum))
        {
            if (currentOper == L"+")
            {
                result = firstNum + secondNum;
            }
            else if (currentOper == L"-")
            {
                result = firstNum - secondNum;
            }
            else if (currentOper == L"x")
            {
                result = firstNum * secondNum;
            }
            else if (currentOper == L"÷")
            {
                if (secondNum != 0)
                {
                    result = firstNum / secondNum;
                }
                else
                {
                    MessageBox::Show("Деление на ноль невозможно", "Error",
MessageBoxButtons::OK, MessageBoxIcon::Error);
                }
            }
        }
    }

```

```

        return;
    }
}

    label1->Text += String::Format(" {0} =",
secondNum);
    textBox1->Text =
System::Convert::ToString(result);
    isCalculated = true;
}
}

private: System::Void buttonordinary1_Click(System::Object^
sender, System::EventArgs^ e)
{
    Owner->Show();
    this->Close();
}
private: System::Void numberbuttons(System::Object^ sender,
System::EventArgs^ e)
{
    Button^ numbers = safe_cast<Button^>(sender);

    if (textBox1->Text == "0")
    {
        textBox1->Text = numbers->Text;
    }
    else if (textBox1->Text->Length <= 8)
    {
        textBox1->Text += numbers->Text;
    }
    isCalculated = false;

    int a = double::Parse(textBox1->Text);
    textBox5->Text = System::Convert::ToString(a, 2);
    textBox4->Text = System::Convert::ToString(a, 8);
    textBox3->Text = System::Convert::ToString(a, 10);
    textBox2->Text = System::Convert::ToString(a, 16);
}
private: System::Void arithmetic(System::Object^ sender,
System::EventArgs^ e)
{
    Button^ operators = safe_cast<Button^>(sender);
    String^ outNumber;

    // second operation
    if (label1->Text != "")
    {
        // second number
        if (!isCalculated)
        {
            secondNum = Double::Parse(textBox1->Text);
        }
    }
}

```

```

        if (Double::IsNaN(result))
        {
            calculate();
        }
        else
        {
            firstNum = result;
            if (!isCalculated)
            {
                calculate();
            }
            outNumber = System::Convert::ToString(result);
        }
    else // first operation
    {
        firstNum = Double::Parse(textBox1->Text);
        outNumber = System::Convert::ToString(firstNum);
    }

    currentOper = operators->Text;
    textBox1->Text = "";
    label1->Text = outNumber + " " + currentOper;
}

private: System::Void equalsign_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!isCalculated)
    {
        //firstNum = Double::IsNaN(result) ? firstNum : result;
        secondNum = Double::Parse(textBox1->Text);
    }
    else
    {
        //firstNum = result;
        label1->Text = String::Format("{0} {1}", firstNum,
currentOper);
    }

    firstNum = Double::IsNaN(result) ? firstNum : result;
    //secondNum = isCalculated ? secondNum :
Double::Parse(CalculationBox->Text);

    calculate();

    int a = int::Parse(textBox1->Text);
    textBox5->Text = System::Convert::ToString(a, 2);
    textBox4->Text = System::Convert::ToString(a, 8);
    textBox3->Text = System::Convert::ToString(a, 10);
    textBox2->Text = System::Convert::ToString(a, 16);
}

```

```

private: System::Void clear_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text->Length > 0)
    {
        textBox1->Text = textBox1->Text->Remove(textBox1->Text-
>Length - 1, 1);
    }
}
private: System::Void erase1_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = "0";
    label1->Text = "";
    result = Double::NaN;
    firstNum = Double::NaN;
    secondNum = Double::NaN;
    isCalculated = false;
    currentOper = "";
}
private: System::Void erase2_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = "";
}
private: System::Void point_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (!textBox1->Text->Contains(","))
    {
        textBox1->Text += ",";
    }
}
private: System::Void change_Click(System::Object^ sender,
System::EventArgs^ e)
{
    if (textBox1->Text->Contains("-"))
    {
        textBox1->Text = textBox1->Text->Remove(0, 1);
    }
    else
    {
        textBox1->Text = "-" + textBox1->Text;
    }
}
private: System::Void MS_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory = Double::Parse(textBox1->Text);
    textBox1->Text = System::Convert::ToString(memory);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}

```

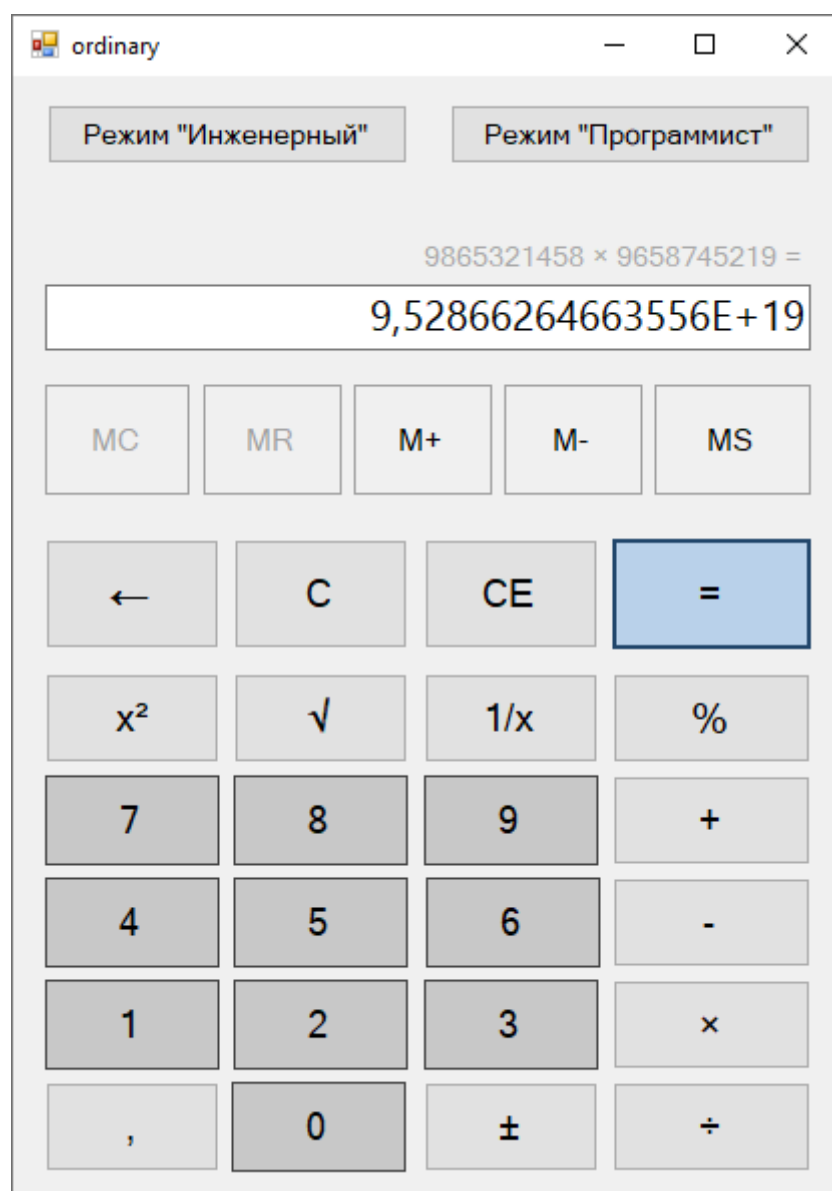
```

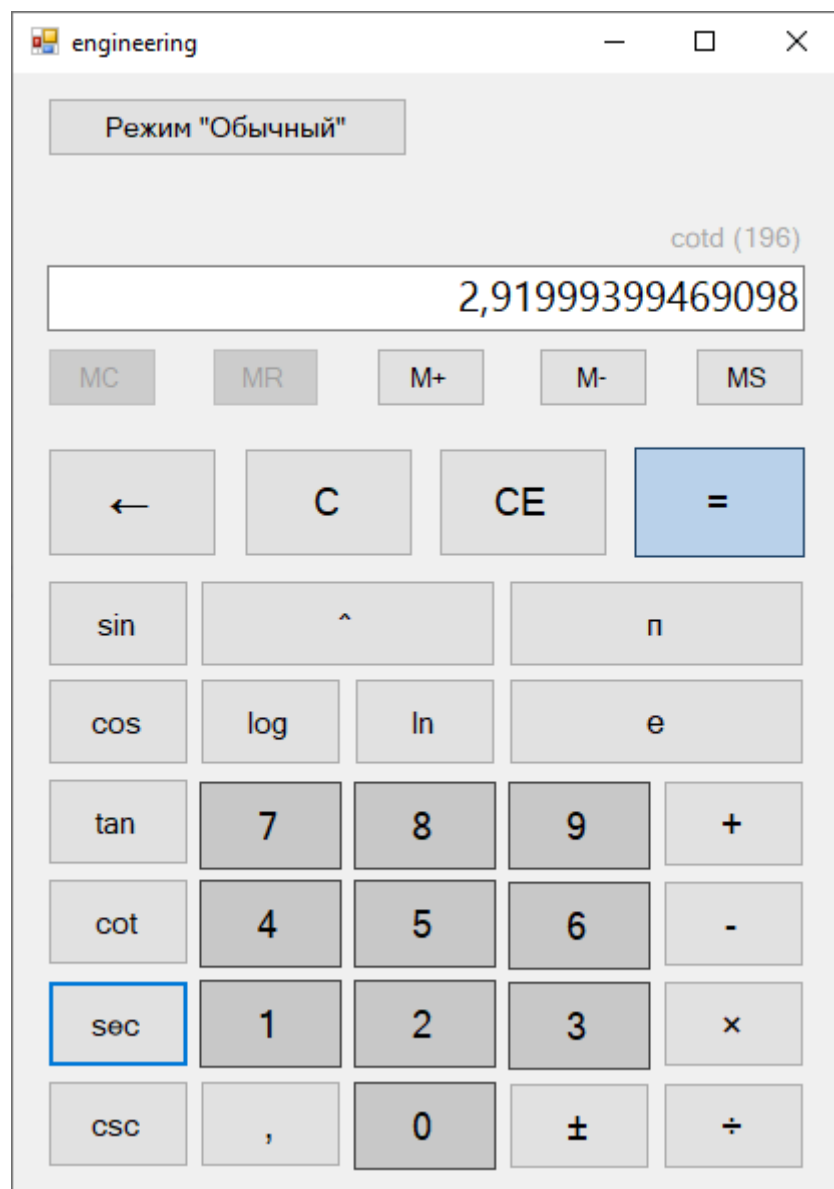
}
private: System::Void MC_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory = 0;
    MC->Enabled = false;
    MR->Enabled = false;
}
private: System::Void MR_Click(System::Object^ sender,
System::EventArgs^ e)
{
    textBox1->Text = System::Convert::ToString(memory);
    isCalculated = true;
}
private: System::Void Mplus_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory += Double::Parse(textBox1->Text);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
private: System::Void Mminus_Click(System::Object^ sender,
System::EventArgs^ e)
{
    memory -= Double::Parse(textBox1->Text);
    isCalculated = true;
    MC->Enabled = true;
    MR->Enabled = true;
}
};
}

```

Приложение В Снимки Экрана

Заголовочный файл «ordinary.h[Конструктор]»





Заголовочный файл «programmatic.h[Конструктор]»

programmatic

Режим "Обычный"

HEX	<input type="text" value="1ce"/>
DEC	<input type="text" value="462"/>
OCT	<input type="text" value="716"/>
BIN	<input type="text" value="111001110"/>

98 + 364 =

MC	MR	M+	M-	MS
←	C	CE	=	
7	8	9	+	
4	5	6	-	
1	2	3	×	
,	0	±	÷	