

Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Пояснительная записка
к курсовой работе
по курсу «Программирование»
на тему «Разработка программы сложной структуры методом нисходящего
программирования»

Выполнил:

Студент группы 21ВВ1

Антропов Данила

Принял:

Токарев А.Н.

077

18.06.22

Пенза 2020

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет Вычислительной техники

Кафедра "Вычислительная техника"

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ



М.А. Митрохин

«___» _____ 20__ г

ЗАДАНИЕ

на курсовое проектирование по курсу

Программирование

Студенту Антоннову Даниле Вадимовичу Группа 21 ВВ1

Тема проекта: *Разработка программы сложной структуры методом нисходящего программирования.*

Исходные данные (технические требования) на проектирование

Приложение интерпретатор

Функции: сложение, вычитание, умножение, деление, извлечение квадратного корня, тригонометрические ср-ия.

Возможность сохранения результата в файл.

Вставки на языке Ассемблер.

Обязательные требования к программе:

1. Многомодульность.
2. Использование сложных типов данных – структур, списков, файлов.
3. Режим работы видеосистемы – текстовый / графический.
4. Устройство ввода информации – клавиатура, мышь.
5. Пользовательский интерфейс должен быть построен на основе меню и панели инструментов.
6. Наличие заставки.
7. Операционная система MS Windows.
8. Язык программирования – Си и Ассемблер.
9. 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 "22" 02. 2022

Дата защиты проекта " " "

Руководитель Токарев Фриц

Задание получил "22" 02 2022г.

Студент Антронов Д.В. Андреев

Содержание

Введение	5
Постановка задачи и обязательные требования.....	6
Описание разработки программы	7
Описание структуры программы.....	9
Отладка и тестирование	17
Описание программы	19
Руководство пользователя	20
Заключение.....	20
Приложение Листинги программы	21

Введение

В современном мире информационные технологии стали неотъемлемой частью жизни практически любого человека.

Снижение трудоёмкости любых производственных, ускорение социальных процессов и, как следствие, повышение их эффективности сейчас связано с существенно изменившимися за последние десятилетия возможностями сбора, обработки, хранения, передачи и представления информации.

Реализация полученной в результате этих процессов информации совершенно нового качества позволяет обеспечить процесс создания и использования материальных и духовных ценностей на более высоком и принципиально новом уровне.

Таким образом изучение современных информационных технологий обеспечивает возможность человеку без проблем ориентироваться в современных тенденциях и путях развития общества, процессах производства и повышать качество социальной жизни.

В таком мире современные информационные системы имеют всё большее значение и применение. Одной из таких является инженерный калькулятор, который используется в самых разных сферах и самыми разными людьми без каких-либо ограничений. Точность вычислений важна как при подсчёте финансового плана на квартал, так и при отправке ракеты в космос. Именно поэтому современные калькуляторы - очень важная информационная система, помогающая человеку, обеспечивающая работу приборов и общую безопасность.

Постановка задачи и обязательные требования

Необходимо разработать программу «Инженерный калькулятор», которая должна отвечать следующим требованиям:

1. Многомодульность. Программа должна быть поделена на логические модули. Это упростит дальнейшее совершенствование программы, а также позволит локализовать ошибки.

2. Использование сложных типов данных - структур, списков, файлов. Программа должна использовать сложные типы данных и работать с ними.

3. Режим работы видеосистемы - текстовый/графический. Необходимо определиться с типом управления, видом интерфейса и изучить приёмы реализации графики.

4. Устройство ввода информации - клавиатура/мышь. Необходимо определиться с механикой взаимодействия игрока с игрой и её интерфейсом.

5. Пользовательский интерфейс должен быть построен на основе меню или панели инструментов. Необходимо реализовать пользовательское меню, с заставкой.

6. Наличие заставки. Перед началом работы в программе, она должна демонстрировать заставку.

7. Операционная система: MS Windows. При разработке использовать среду MS Windows.

8. Язык программирования - Си и Ассемблер, а также некоторые функции из языка C++ для работы с графикой.

9. Среда разработки ПО - Microsoft Visual Studio. При разработке использовать систему программирования MS Visual Studio.

Программа должна реализовывать следующие функции:

1. Операции:

1) Сложение

2) Вычитание

3) Умножение

4) Деление

- 5) Извлечение квадратного корня
- 6) Тригонометрические функции
- 2. Возможность сохранения результата в файл.

Описание разработки программы

Для написания данной программы будет использован язык программирования C++ с некоторыми элементами языка C. Этот язык является одним из самых распространённых языков программирования. При разработке языка C был принят компромисс между низким уровнем языка Assembler и высоким уровнем других языков. В языке C предусмотрено много операций, непосредственно исполняемых большинством микропроцессоров; в то же время он даёт программисту возможность выражать свои алгоритмы и данные наиболее подходящими средствами. Также язык C обеспечивает возможности структурирования данных. Он позволяет разрабатывать большие, но структурированные программы, предоставляя возможность отдельной разработки подпрограмм.

Целью создателей языка C 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 C, 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 C.

В качестве среды программирования, в соответствии с обязательными требованиями, был выбран программный продукт MS Visual Studio 2019, как одна из наиболее современных программных сред, позволяющих реализовывать свои алгоритмы. Также подключались необходимые для Windows Forms библиотеки, так как при работе с Windows Forms наиболее удобно работать с графическим

интерфейсом, который был выбран для реализации данного проекта в связи с тем, что он является наиболее удобным и интуитивно понятным для пользователя.

Интерфейс информационной системы схож со стандартным калькулятором, который предлагает Windows, поэтому у пользователя не должно возникнуть проблем с её использованием.

При запуске программы по центру экрана появляется заставка, которая исчезает через несколько секунд, после чего на экран выводится основной интерфейс информационной системы. Реализация всех основных математических операций осуществляется с помощью кнопок с соответствующими надписями на них. Ввод данных пользователем и вывод результатов вычислений выполняется в специально выделенные текстовые поля, которые были ограничены на ввод неправильных данных для избежания возможных ошибок. Помимо этого присутствуют кнопки управления данными в текстовых полях: пользователь может очистить все поля, либо же удалить последний символ в поле ввода данных, а также при необходимости сохранить вычисление в файл по нажатию кнопки.

Разработка программы началась с реализации графического интерфейса в виде размещения на поле кнопок и подгон их размера под размер поля, размер шрифта и относительно друг друга, чтобы пользователю было понятно, какая кнопка за что отвечает. Далее была реализована работа всех кнопок, отвечающих за вычисления и изменения полей, а потом кнопки, сохраняющей вычисление в файл, что было сделано с использованием языка Asstmblr. Поскольку Windows Forms не позволяет делать вставки языка Assembler в любом месте программы, для реализации работы этой кнопки была написана функция записи в файл, выбранный пользователем в соответствующем диалоге, в отдельном asm файле, которая вызывается в основной программе.

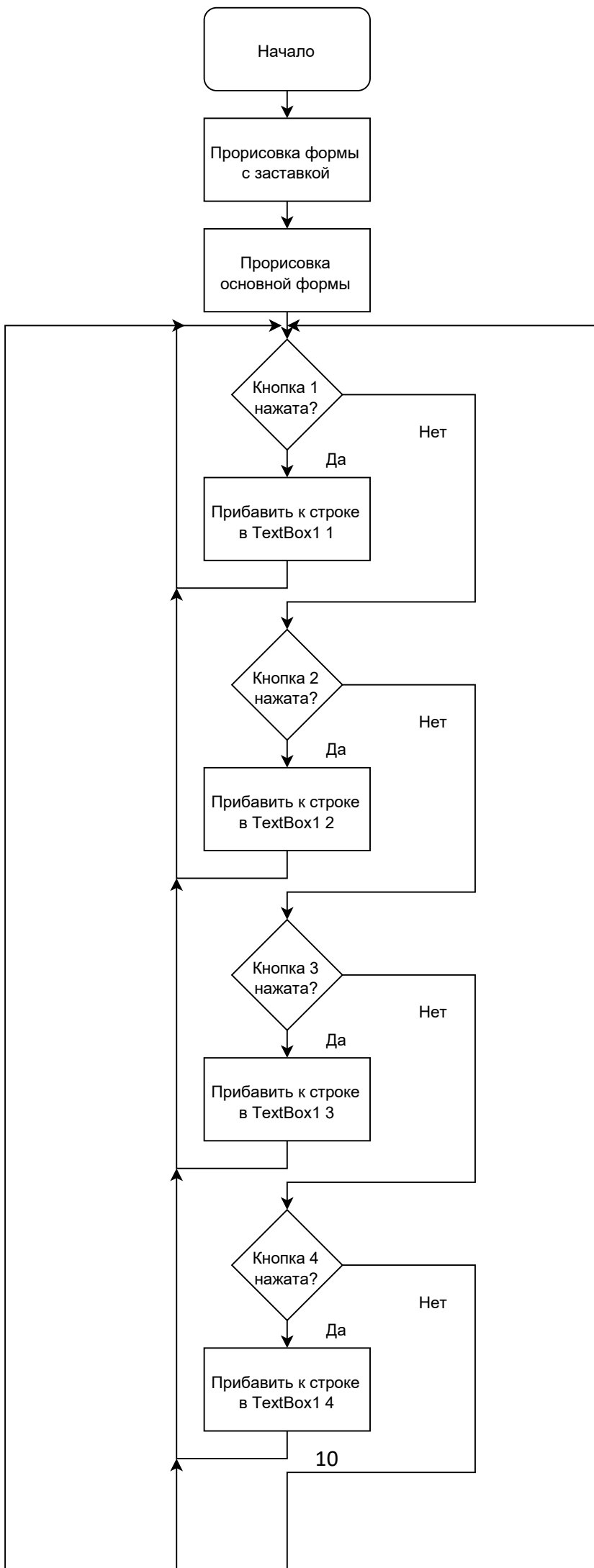
С помощью создания новой формы была реализована заставка приложения, появляющаяся на определённое время перед основным интерфейсом.

Описание структуры программы

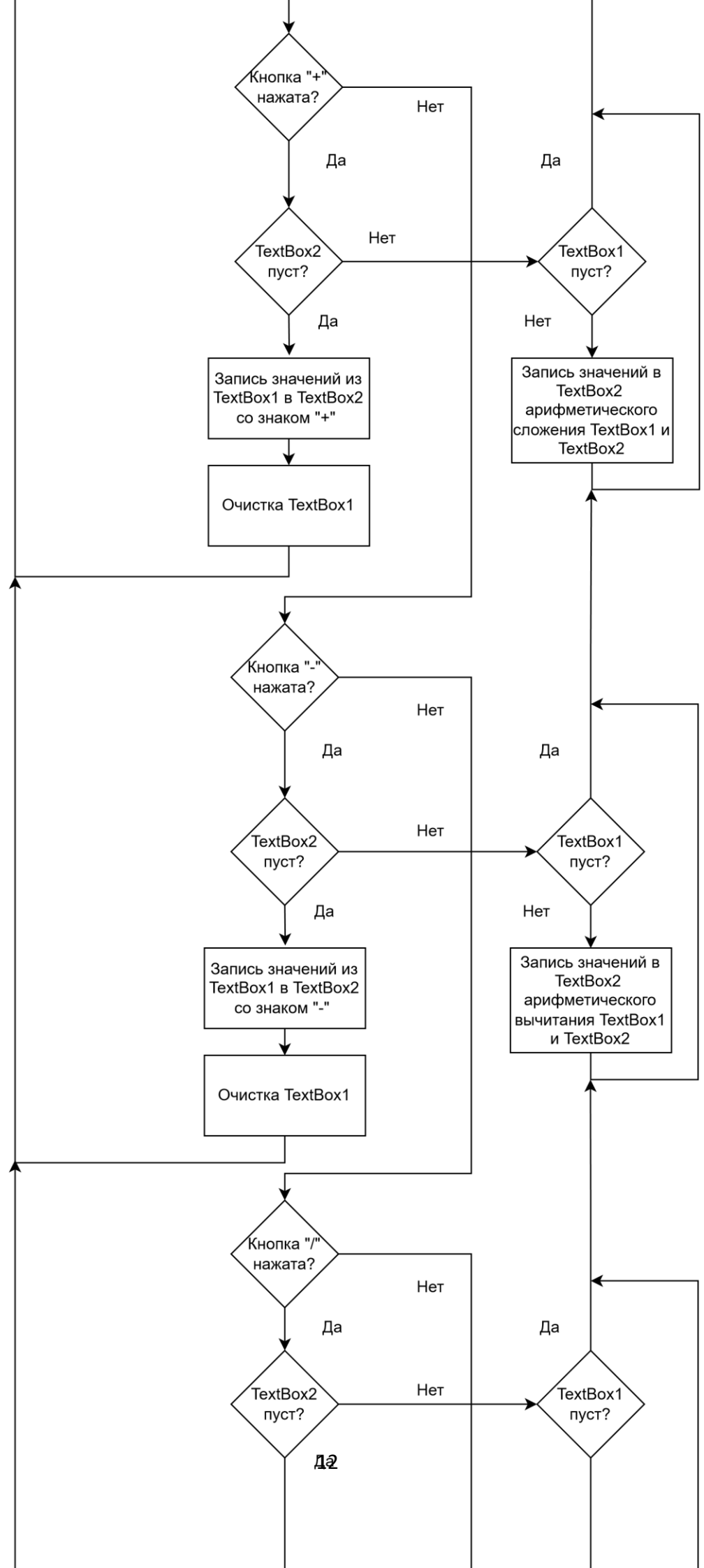
Поскольку программный продукт Windows Forms размещает весь код, связанный с формой, в заголовочном файле MyForm.h, то в нём будут располагаться используемые переменные, а также инициализироваться функции, которые используются в кодах для кнопок, которые также находятся в этом файле. А в основной файл формы MyForm.cpp, который включает в себя

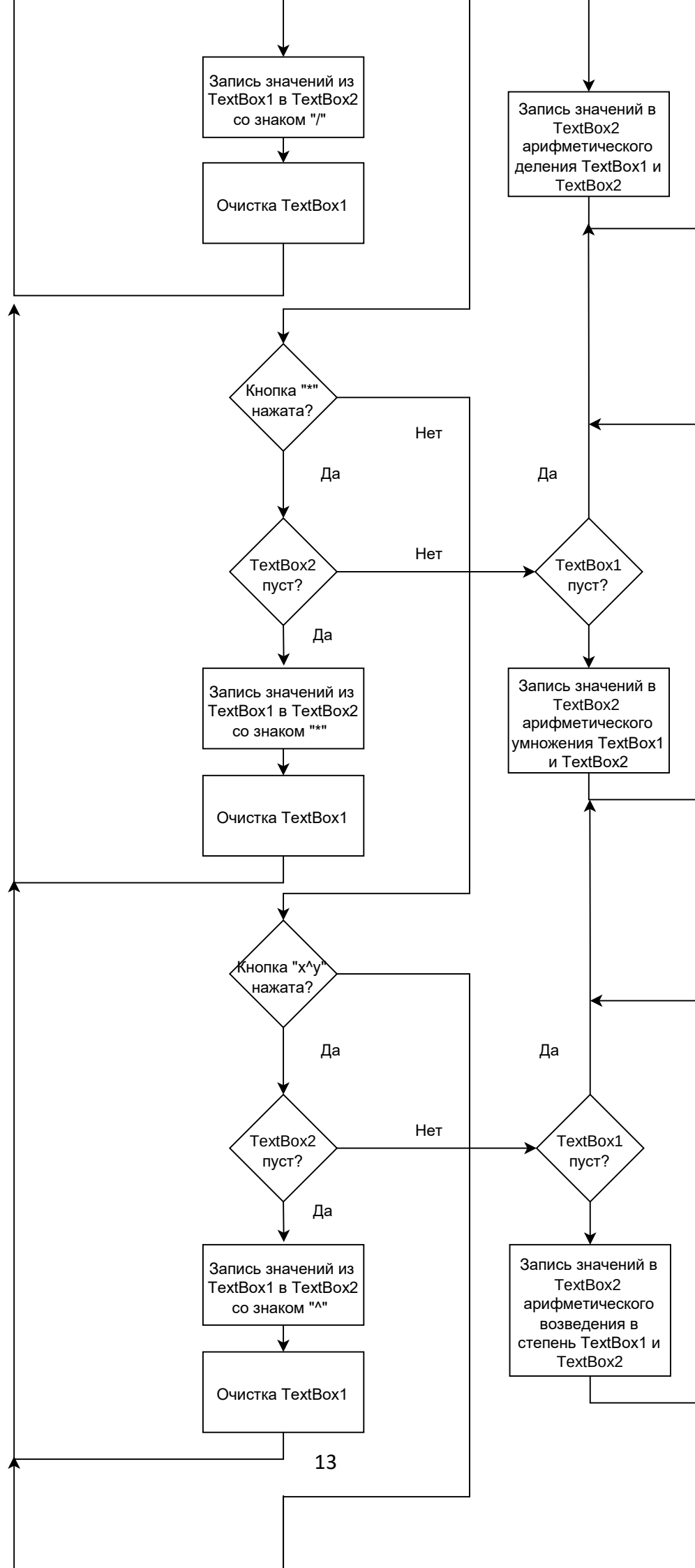
точку входа в приложения и вывод дополнительной формы(заставки) в течение 3 секунд и основной формы по очереди, подключаем заголовочные файлы обоих форм MyForm.h и MyForm1.h. Также в программе присутствует файл Source.asm, в котором производится запись данных в файл.

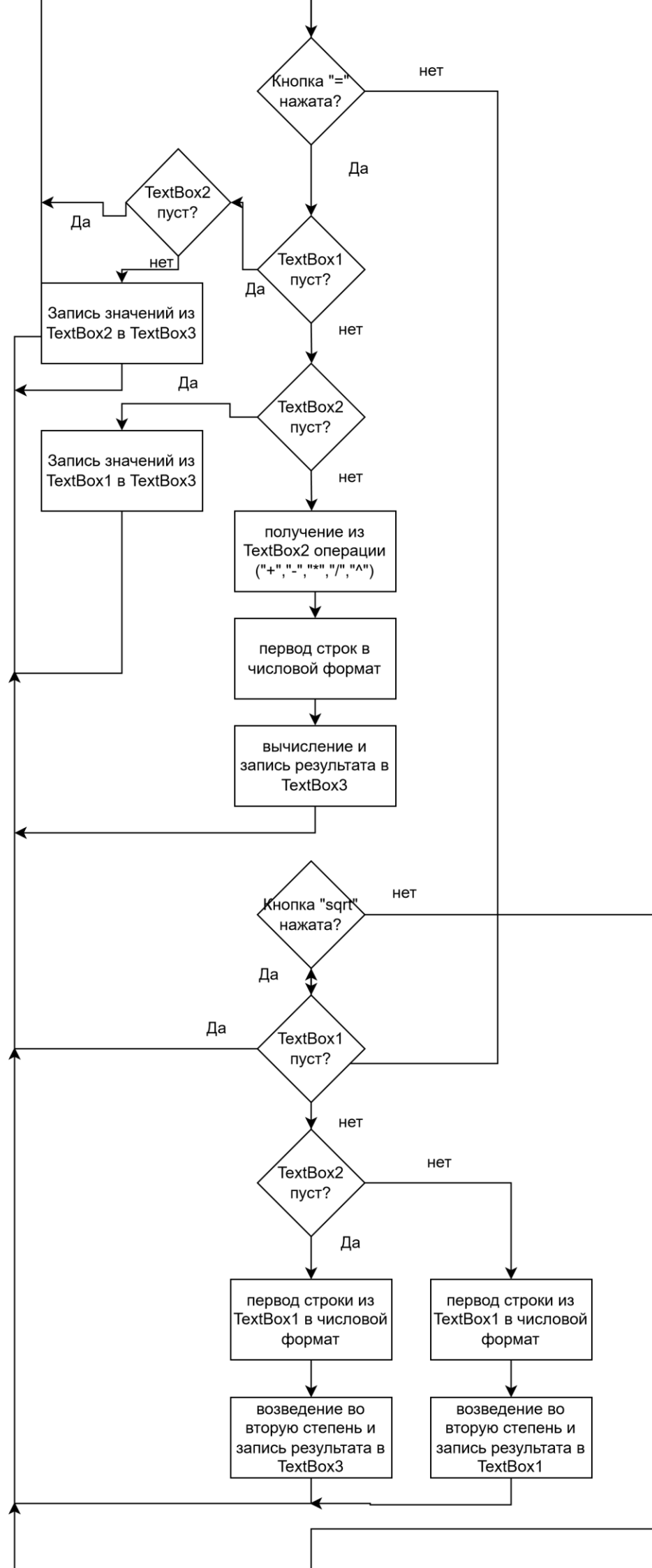
Алгоритм работы программы представлен на рисунке 1.

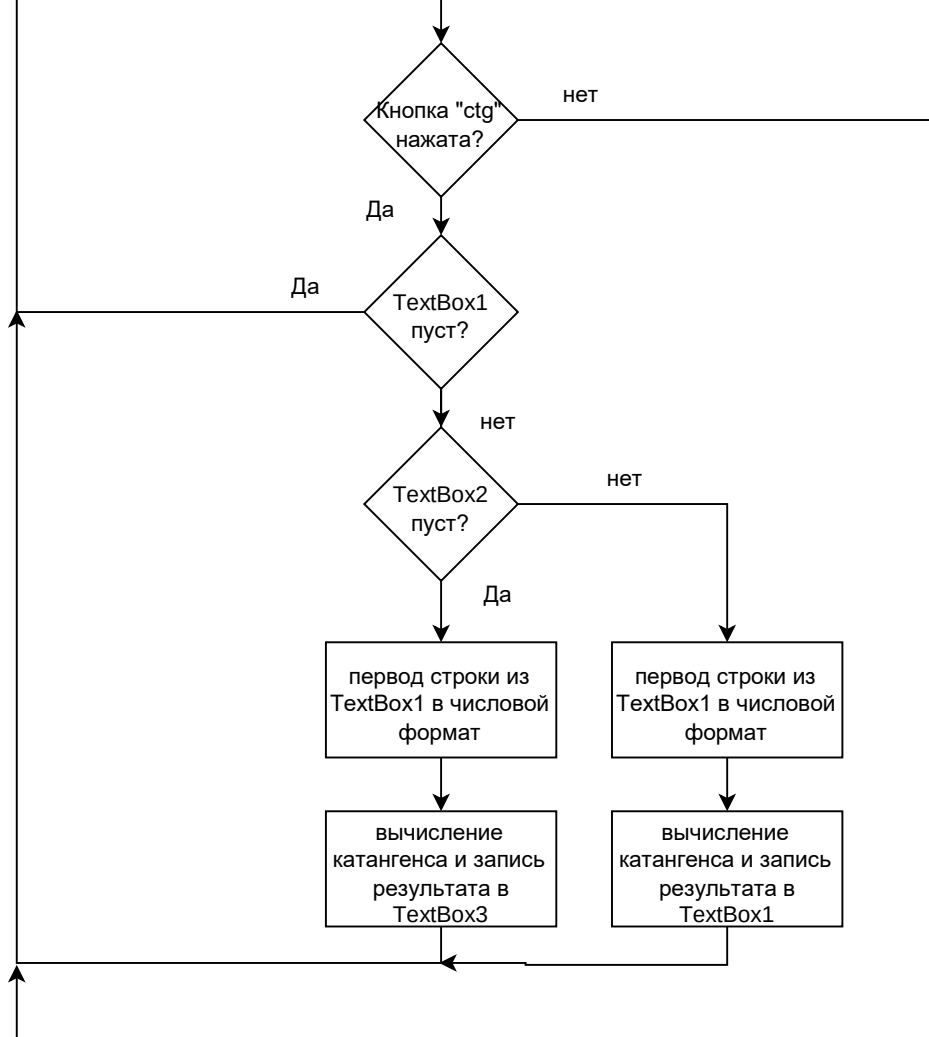
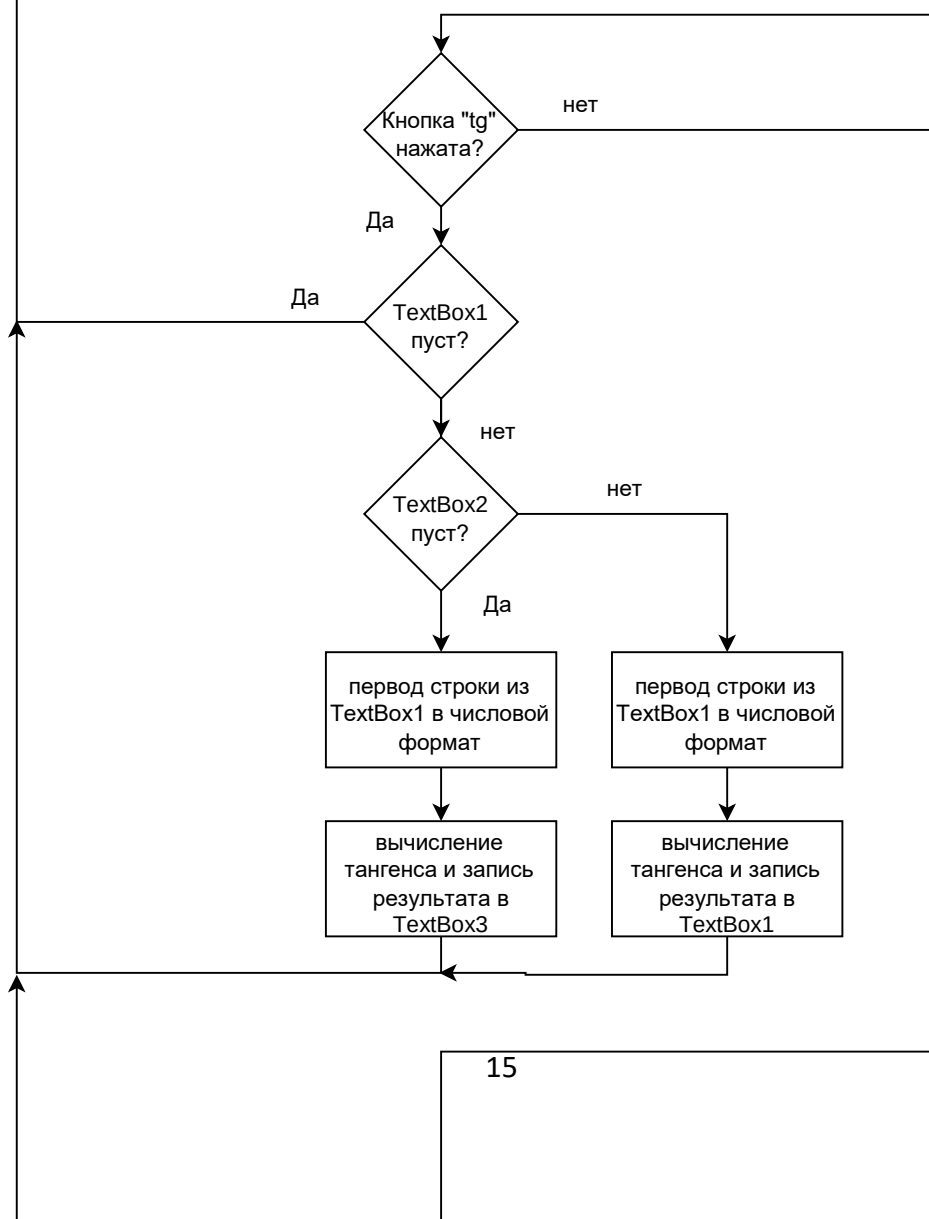


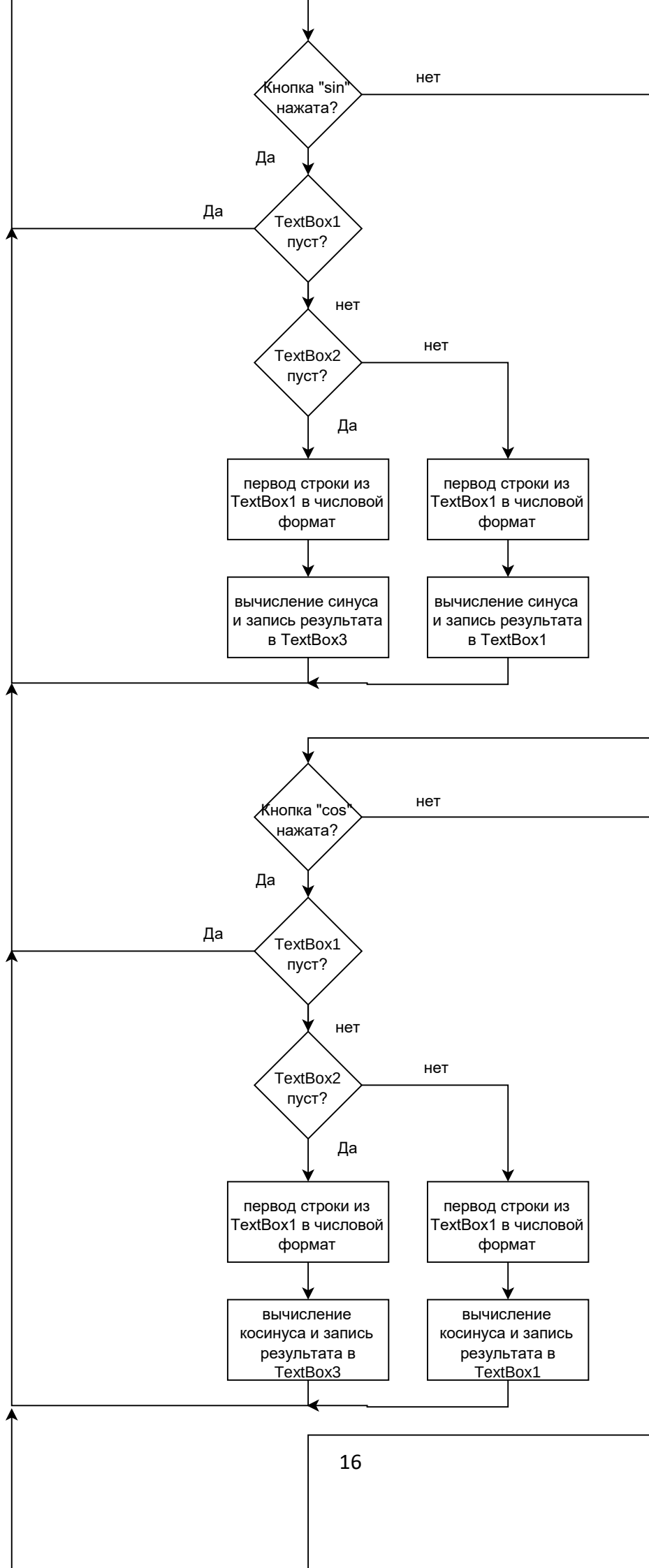


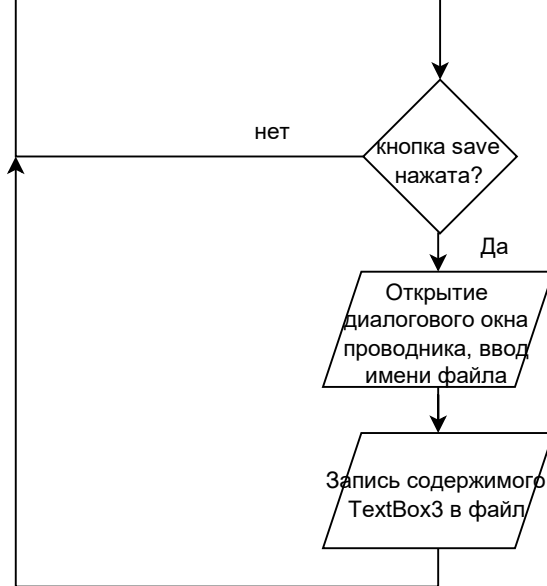












Отладка и тестирование

Тестирование проводилось в рабочем порядке, в процессе разработки, после завершения написания программы.

1. Тестирование на вывод заставки. При запуске программы окно должно принять указанный размер (640x360) и отобразиться по центру экрана. Изображение заставки не обрезано, отображается корректно.



2. Тестирование на вывод основной формы

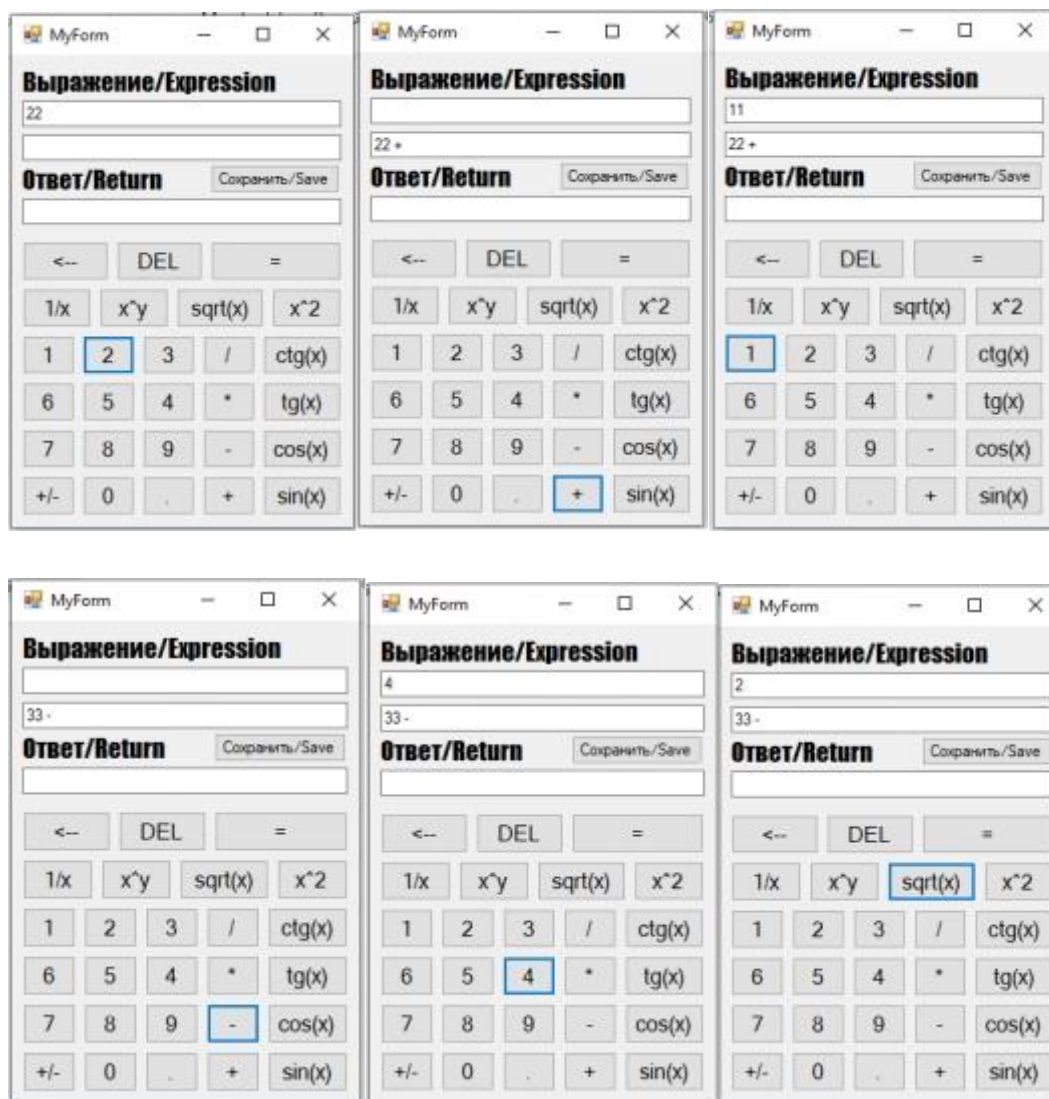
После вывода заставки спустя 3 секунды она должна пропасть и появиться основная форма. Отображается корректно.

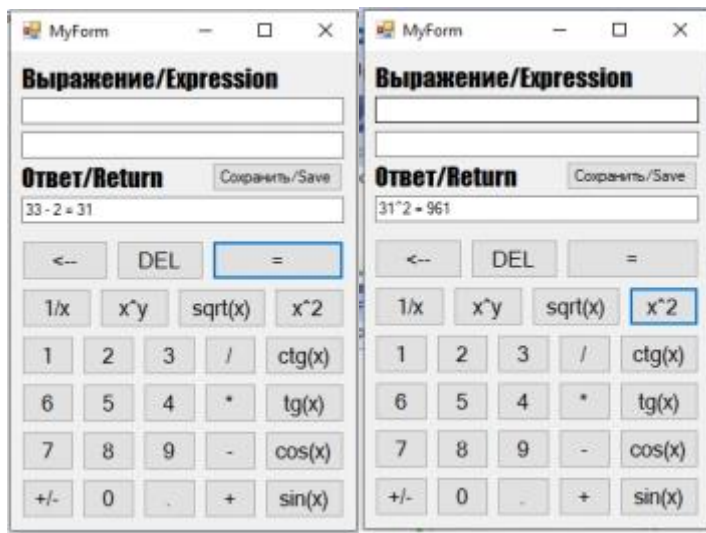
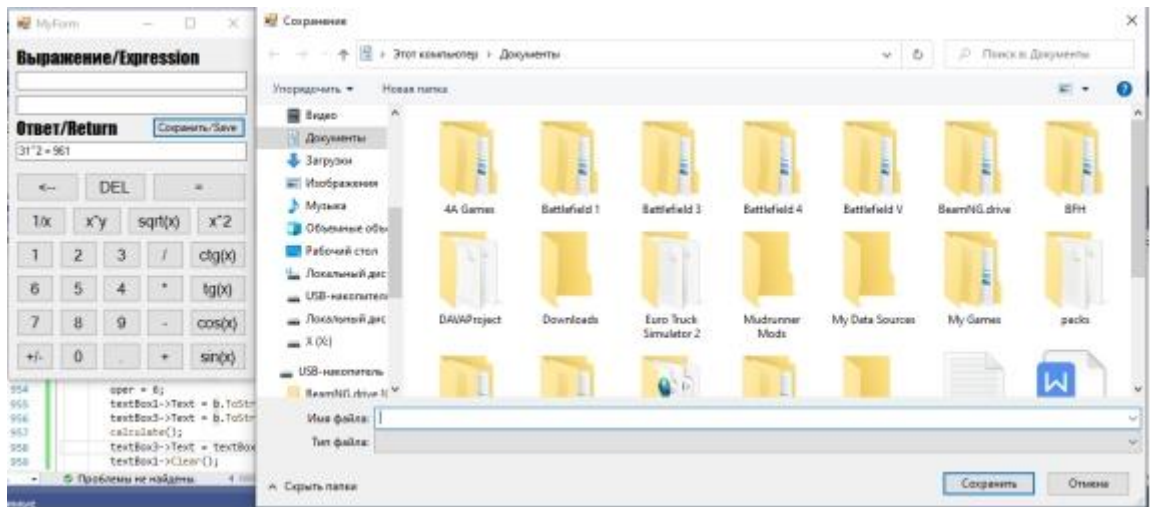
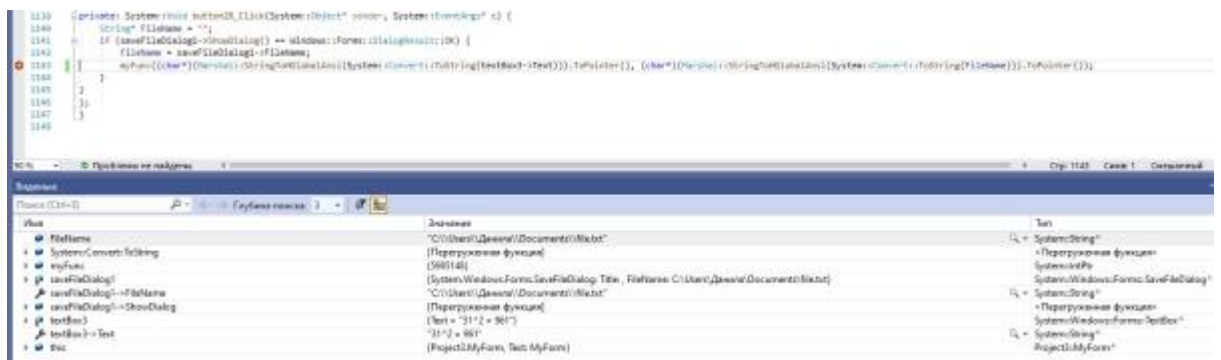


3. Тестирование на расчёт формулы.

Расчёт в программе формулы $(22+11-\sqrt{4})^2$ и запись результата в файл.

Расчёт верен, запись в файл производится, программа работает верно.





Описание программы

Кнопки работают по 2 основным алгоритмам:

1 алгоритм(сложение, вычитание, умножение, деление, возведение в степень):

Если все 3 textBox пусты, то ничего не делать, так же, как и если textBox1 пустБ а textBox2 не пуст. Если 1 и 2 textBox пусты, то операндом становится ответ в textBox3, который переносится в textBox2 со знаком операции. Если 1 и 2 textBox не пусты, то в textBox2 записывается результат вычисления по знаку, который там был, вместе с новым знаком операции. В остальных случаях число из textBox1 переносится в textBox2 со знаком операции.

2 алгоритм(корень, возведение в квадрат, деление единицы на число, синус, косинус, тангенс, котангенс):

Если все 3 textBox пусты либо 1 textBox пуст и 2 textBox не пуст - ничего не делать. Если 1 и 2 textBox не пусты, то операция применяется к числу в textBox1, и туда же записывается ответ, чтобы можно было продолжить вычисление с textBox2. Если 1 и 2 textBox пусты, то операндом становится ответ в textBox3, с которым производится операция и ответ записывается в textBox3. В остальных случаях операция производится с числом в textBox1 и ответ записывается в textBox3.

Кнопка со знаком меняет знак операнда в textBox1, что меняет знак числа.

Кнопка DEL очищает все поля, а кнопка «<-» стирает 1 знак с конца в textBox1.

Кнопка сохранения запускает пользовательский диалог по сохранению файла.

Руководство пользователя

Программа работает по принципу, схожему со стандартным калькулятором: по кнопкам цифр в поле ввода добавляются соответствующие цифры, операции сложения, вычитания, умножения и деления выполняются по стандарту, остальные - сразу по нажатию. Сохранение в файл осуществляется по удобному диалогу - пользователь сам может выбрать название файла и его место расположения.

Заключение

При выполнении данной курсовой работы были получены навыки разработки многомодульных программ. Были освоены принципы создания графических приложений с помощью «WinApi». Изучены способы обработки событий с

клавиатуры. Получены знания и навыки по работе с файлами. Получены базовые навыки программирования на языке Си и Ассемблера, а также некоторых функций из Си++. Получены навыки отладки и тестирования программ. В рамках данной курсовой было написано приложение «Инженерный калькулятор». Данная программа не требует большого количества системных ресурсов, имеет полноценный графический интерфейс и управление с помощью клавиш. В дальнейшем, есть идеи по усовершенствованию программы: добавить больше команд, панель с историей операций, усовершенствование интерфейса.

Приложение Листинги программы

Файл MyForm.h

```
#pragma once
```

```
#include <string>
```

```
#include <cmath>
```

```
#define PI 3.14159265
```

```
extern "C" void myFunc(char* stroka, char* filename);
```

```
namespace Project3 {
```

```
    using namespace System;
```

```
    using namespace System::ComponentModel;
```

```
    using namespace System::Collections;
```

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

```
    using namespace System::Data;
```

```
    using namespace System::Drawing;
```

```
using namespace System::IO;
```

```
using namespace System::Runtime::InteropServices;
```

```
float a, b;
```

```
int oper, zap;
```

```
/// <summary>
```

```
/// Сводка для MyForm
```

```
/// </summary>
```

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

```
{
```

```
public:
```

```
    MyForm(void)
```

```
{
```

```
    InitializeComponent();
```

```
    //
```

```
    //TODO: добавьте код конструктора
```

```
    //
```

```
}
```

protected:

```
    /// <summary>

    /// Освободить все используемые ресурсы.

    /// </summary>

    ~MyForm()

    {

        if (components)

        {

            delete components;

        }

    }

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::Button^ button3;

private: System::Windows::Forms::Button^ button4;

private: System::Windows::Forms::Button^ button5;

private: System::Windows::Forms::Button^ button6;

private: System::Windows::Forms::Button^ button7;

private: System::Windows::Forms::Button^ button8;

private: System::Windows::Forms::Button^ button9;

private: System::Windows::Forms::Button^ button10;

private: System::Windows::Forms::Button^ button11;
```

```
private: System::Windows::Forms::Button^ button12;  
private: System::Windows::Forms::Button^ button13;  
private: System::Windows::Forms::Button^ button14;  
private: System::Windows::Forms::Button^ button15;  
private: System::Windows::Forms::Button^ button16;  
private: System::Windows::Forms::Button^ button17;  
private: System::Windows::Forms::Button^ button18;  
private: System::Windows::Forms::Button^ button19;  
private: System::Windows::Forms::Button^ button20;  
private: System::Windows::Forms::Button^ button21;  
private: System::Windows::Forms::Button^ button22;  
private: System::Windows::Forms::Button^ button23;  
private: System::Windows::Forms::Button^ button24;  
private: System::Windows::Forms::Button^ button25;  
private: System::Windows::Forms::Button^ button26;  
private: System::Windows::Forms::Label^ label1;  
private: System::Windows::Forms::Label^ label2;  
private: System::Windows::Forms::Button^ button27;  
private: System::Windows::Forms::Button^ button28;  
  
private: System::Windows::Forms::TextBox^ textBox1;  
private: System::Windows::Forms::TextBox^ textBox2;
```

```
private: System::Windows::Forms::SaveFileDialog^ saveFileDialog1;
```

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

```
private: void calculate() {
```

```
    switch (oper)
```

```
    {
```

```
    case 1:
```

```
        b = a + float::Parse(textBox1->Text);
```

```
        break;
```

```
    case 2:
```

```
        b = a - float::Parse(textBox1->Text);
```

```
        break;
```

```
    case 3:
```

```
        b = a * float::Parse(textBox1->Text);
```

```
        break;
```

```
    case 4:
```

```
        if (float::Parse(textBox1->Text) == 0) {
```

```
            textBox3->Text = "Делить на 0 нельзя!";
```

```
        }
```

```

        else {

            b = a / float::Parse(textBox1->Text);

        }

        break;

    case 5:

        b = 1 / float::Parse(textBox1->Text);

        break;

    case 6:

        b = float::Parse(textBox1->Text) * float::Parse(textBox1->Text);

        break;

    case 7:

        if (float::Parse(textBox1->Text) < 0) {

            textBox3->Text = "Корня из отрицательного числа
нет!";

        }

        else {

            b = sqrt(float::Parse(textBox1->Text));

        }

```

break;

case 8:

$b = \sin(\text{float}::\text{Parse}(\text{textBox1} \rightarrow \text{Text}) * \text{PI} / 180);$

break;

case 9:

$b = \cos(\text{float}::\text{Parse}(\text{textBox1} \rightarrow \text{Text}) * \text{PI} / 180);$

break;

case 10:

$b = \tan(\text{float}::\text{Parse}(\text{textBox1} \rightarrow \text{Text}) * \text{PI} / 180);$

break;

case 11:

$b = 1/\tan(\text{float}::\text{Parse}(\text{textBox1} \rightarrow \text{Text}) * \text{PI} / 180);$

break;

case 12:

$b = \text{pow}(a, \text{float}::\text{Parse}(\text{textBox1} \rightarrow \text{Text}));$

default:

break;

```

    }

}

private: void DeleteLasttextBox2() {

    int lenght = textBox2->Text->Length - 3;

    auto text = textBox2->Text;

    textBox2->Clear();

    for (int i = 0; i < lenght; i++) {

        textBox2->Text = textBox2->Text + text[i];

    }

}

```

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->button1 = (gcnew System::Windows::Forms::Button());  
this->button2 = (gcnew System::Windows::Forms::Button());  
this->button3 = (gcnew System::Windows::Forms::Button());  
this->button4 = (gcnew System::Windows::Forms::Button());  
this->button5 = (gcnew System::Windows::Forms::Button());  
this->button6 = (gcnew System::Windows::Forms::Button());  
this->button7 = (gcnew System::Windows::Forms::Button());  
this->button8 = (gcnew System::Windows::Forms::Button());  
this->button9 = (gcnew System::Windows::Forms::Button());  
this->button10 = (gcnew System::Windows::Forms::Button());  
this->button11 = (gcnew System::Windows::Forms::Button());  
this->button12 = (gcnew System::Windows::Forms::Button());  
this->button13 = (gcnew System::Windows::Forms::Button());  
this->button14 = (gcnew System::Windows::Forms::Button());  
this->button15 = (gcnew System::Windows::Forms::Button());  
this->button16 = (gcnew System::Windows::Forms::Button());  
this->button17 = (gcnew System::Windows::Forms::Button());  
this->button18 = (gcnew System::Windows::Forms::Button());  
this->button19 = (gcnew System::Windows::Forms::Button());  
this->button20 = (gcnew System::Windows::Forms::Button());  
this->button21 = (gcnew System::Windows::Forms::Button());
```

```

this->button22 = (gcnew System::Windows::Forms::Button());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->button23 = (gcnew System::Windows::Forms::Button());

this->button24 = (gcnew System::Windows::Forms::Button());

this->button25 = (gcnew System::Windows::Forms::Button());

this->button26 = (gcnew System::Windows::Forms::Button());

this->label1 = (gcnew System::Windows::Forms::Label());

this->label2 = (gcnew System::Windows::Forms::Label());

this->textBox3 = (gcnew System::Windows::Forms::TextBox());

this->button27 = (gcnew System::Windows::Forms::Button());

this->button28 = (gcnew System::Windows::Forms::Button());

this->saveFileDialog1 = (gcnew
System::Windows::Forms::SaveFileDialog());

this->SuspendLayout();

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button1->Location = System::Drawing::Point(8, 215);

```

```

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(40, 30);

this->button1->TabIndex = 0;

this->button1->Text = L"1";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this,
&MyForm::button1_Click);

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button2->Location = System::Drawing::Point(54, 215);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(40, 30);

this->button2->TabIndex = 1;

this->button2->Text = L"2";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this,
&MyForm::button2_Click);

//

```

```

        // button3

        //

        this->button3->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans                Serif",                12,                System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button3->Location = System::Drawing::Point(100, 215);

        this->button3->Name = L"button3";

        this->button3->Size = System::Drawing::Size(40, 30);

        this->button3->TabIndex = 2;

        this->button3->Text = L"3";

        this->button3->UseVisualStyleBackColor = true;

        this->button3->Click += gcnew System::EventHandler(this,
&MyForm::button3_Click);

        //

        // button4

        //

        this->button4->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans                Serif",                12,                System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button4->Location = System::Drawing::Point(100, 251);

        this->button4->Name = L"button4";

        this->button4->Size = System::Drawing::Size(40, 30);

```

```

this->button4->TabIndex = 3;

this->button4->Text = L"4";

this->button4->UseVisualStyleBackColor = true;

this->button4->Click += gcnew System::EventHandler(this,
&MyForm::button4_Click);

//

// button5

//

this->button5->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(0)));

this->button5->Location = System::Drawing::Point(54, 251);

this->button5->Name = L"button5";

this->button5->Size = System::Drawing::Size(40, 30);

this->button5->TabIndex = 4;

this->button5->Text = L"5";

this->button5->UseVisualStyleBackColor = true;

this->button5->Click += gcnew System::EventHandler(this,
&MyForm::button5_Click);

//

// button6

//

```

```

        this->button6->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans                Serif",                12,                System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button6->Location = System::Drawing::Point(8, 251);

        this->button6->Name = L"button6";

        this->button6->Size = System::Drawing::Size(40, 30);

        this->button6->TabIndex = 5;

        this->button6->Text = L"6";

        this->button6->UseVisualStyleBackColor = true;

        this->button6->Click += gcnew System::EventHandler(this,
&MyForm::button6_Click);

        //

        // button7

        //

        this->button7->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans                Serif",                12,                System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button7->Location = System::Drawing::Point(8, 287);

        this->button7->Name = L"button7";

        this->button7->Size = System::Drawing::Size(40, 30);

        this->button7->TabIndex = 6;

        this->button7->Text = L"7";

```

```

        this->button7->UseVisualStyleBackColor = true;

        this->button7->Click += gcnew System::EventHandler(this,
&MyForm::button7_Click);

        //

        // button8

        //

        this->button8->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

        static_cast<System::Byte>(0)));

        this->button8->Location = System::Drawing::Point(54, 287);

        this->button8->Name = L"button8";

        this->button8->Size = System::Drawing::Size(40, 30);

        this->button8->TabIndex = 7;

        this->button8->Text = L"8";

        this->button8->UseVisualStyleBackColor = true;

        this->button8->Click += gcnew System::EventHandler(this,
&MyForm::button8_Click);

        //

        // button9

        //

        this->button9->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

```

```

        static_cast<System::Byte>(0)));

this->button9->Location = System::Drawing::Point(100, 287);

this->button9->Name = L"button9";

this->button9->Size = System::Drawing::Size(40, 30);

this->button9->TabIndex = 8;

this->button9->Text = L"9";

this->button9->UseVisualStyleBackColor = true;

this->button9->Click += gcnew System::EventHandler(this,
&MyForm::button9_Click);

//

// button10

//

this->button10->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

        static_cast<System::Byte>(0)));

this->button10->Location = System::Drawing::Point(54, 323);

this->button10->Name = L"button10";

this->button10->Size = System::Drawing::Size(40, 30);

this->button10->TabIndex = 9;

this->button10->Text = L"0";

this->button10->UseVisualStyleBackColor = true;

```

```

        this->button10->Click += gcnew System::EventHandler(this,
&MyForm::button10_Click);

        //

        // button11

        //

        this->button11->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

        static_cast<System::Byte>(0)));

        this->button11->Location = System::Drawing::Point(146, 323);

        this->button11->Name = L"button11";

        this->button11->Size = System::Drawing::Size(40, 30);

        this->button11->TabIndex = 10;

        this->button11->Text = L"+";

        this->button11->UseVisualStyleBackColor = true;

        this->button11->Click += gcnew System::EventHandler(this,
&MyForm::button11_Click);

        //

        // button12

        //

        this->button12->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

        static_cast<System::Byte>(0)));

```

```

this->button12->Location = System::Drawing::Point(146, 287);

this->button12->Name = L"button12";

this->button12->Size = System::Drawing::Size(40, 30);

this->button12->TabIndex = 11;

this->button12->Text = L"-";

this->button12->UseVisualStyleBackColor = true;

this->button12->Click += gcnew System::EventHandler(this,
&MyForm::button12_Click);

//

// button13

//

this->button13->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button13->Location = System::Drawing::Point(146, 251);

this->button13->Name = L"button13";

this->button13->Size = System::Drawing::Size(40, 30);

this->button13->TabIndex = 12;

this->button13->Text = L"*";

this->button13->UseVisualStyleBackColor = true;

this->button13->Click += gcnew System::EventHandler(this,
&MyForm::button13_Click);

```

```

//

// button14

//

this->button14->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans          Serif",          12,          System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

          static_cast<System::Byte>(0)));

this->button14->Location = System::Drawing::Point(146, 215);

this->button14->Name = L"button14";

this->button14->Size = System::Drawing::Size(40, 30);

this->button14->TabIndex = 13;

this->button14->Text = L"/";

this->button14->UseVisualStyleBackColor = true;

this->button14->Click += gcnew System::EventHandler(this,
&MyForm::button14_Click);

//

// button15

//

this->button15->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans          Serif",          12,          System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

          static_cast<System::Byte>(0)));

this->button15->Location = System::Drawing::Point(126, 179);

this->button15->Name = L"button15";

```

```

this->button15->Size = System::Drawing::Size(67, 30);

this->button15->TabIndex = 14;

this->button15->Text = L"sqrt(x)";

this->button15->UseVisualStyleBackColor = true;

this->button15->Click += gcnew System::EventHandler(this,
&MyForm::button15_Click);

//

// button16

//

this->button16->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button16->Location = System::Drawing::Point(192, 287);

this->button16->Name = L"button16";

this->button16->Size = System::Drawing::Size(60, 30);

this->button16->TabIndex = 15;

this->button16->Text = L"cos(x)";

this->button16->UseVisualStyleBackColor = true;

this->button16->Click += gcnew System::EventHandler(this,
&MyForm::button16_Click);

//

// button17

```

```

//

this->button17->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans          Serif",          12,          System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button17->Location = System::Drawing::Point(192, 251);

this->button17->Name = L"button17";

this->button17->Size = System::Drawing::Size(60, 30);

this->button17->TabIndex = 16;

this->button17->Text = L"tg(x)";

this->button17->UseVisualStyleBackColor = true;

this->button17->Click += gcnew System::EventHandler(this,
&MyForm::button17_Click);

//

// button18

//

this->button18->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans          Serif",          12,          System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button18->Location = System::Drawing::Point(192, 215);

this->button18->Name = L"button18";

this->button18->Size = System::Drawing::Size(60, 30);

this->button18->TabIndex = 17;

```

```

this->button18->Text = L"ctg(x)";

this->button18->UseVisualStyleBackColor = true;

this->button18->Click += gcnew System::EventHandler(this,
&MyForm::button18_Click);

//

// button19

//

this->button19->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button19->Location = System::Drawing::Point(192, 323);

this->button19->Name = L"button19";

this->button19->Size = System::Drawing::Size(60, 30);

this->button19->TabIndex = 18;

this->button19->Text = L"sin(x)";

this->button19->UseVisualStyleBackColor = true;

this->button19->Click += gcnew System::EventHandler(this,
&MyForm::button19_Click);

//

// button20

//

```

```

        this->button20->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans                Serif",                12,                System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button20->Location = System::Drawing::Point(152, 143);

        this->button20->Name = L"button20";

        this->button20->Size = System::Drawing::Size(100, 30);

        this->button20->TabIndex = 19;

        this->button20->Text = L"";

        this->button20->UseVisualStyleBackColor = true;

        this->button20->Click += gcnew System::EventHandler(this,
&MyForm::button20_Click);

        //

        // button21

        //

        this->button21->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans                Serif",                12,                System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button21->Location = System::Drawing::Point(8, 143);

        this->button21->Name = L"button21";

        this->button21->Size = System::Drawing::Size(66, 30);

        this->button21->TabIndex = 20;

        this->button21->Text = L"<--";

```

```

        this->button21->UseVisualStyleBackColor = true;

        this->button21->Click += gcnew System::EventHandler(this,
&MyForm::button21_Click);

        //

        // button22

        //

        this->button22->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

        static_cast<System::Byte>(0)));

        this->button22->Location = System::Drawing::Point(80, 143);

        this->button22->Name = L"button22";

        this->button22->Size = System::Drawing::Size(66, 30);

        this->button22->TabIndex = 21;

        this->button22->Text = L"DEL";

        this->button22->UseVisualStyleBackColor = true;

        this->button22->Click += gcnew System::EventHandler(this,
&MyForm::button22_Click);

        //

        // textBox1

        //

        this->textBox1->Location = System::Drawing::Point(8, 35);

        this->textBox1->Name = L"textBox1";

```

```

        this->textBox1->Size = System::Drawing::Size(244, 20);

        this->textBox1->TabIndex = 22;

        this->textBox1->TextChanged += gcnew
System::EventHandler(this, &MyForm::textBox1_TextChanged);

        this->textBox1->KeyPress += gcnew
System::Windows::Forms::KeyPressEventHandler(this, &MyForm::textBox1_KeyPress);

        //

        // textBox2

        //

        this->textBox2->Location = System::Drawing::Point(8, 61);

        this->textBox2->Name = L"textBox2";

        this->textBox2->Size = System::Drawing::Size(244, 20);

        this->textBox2->TabIndex = 23;

        this->textBox2->KeyPress += gcnew
System::Windows::Forms::KeyPressEventHandler(this, &MyForm::textBox2_KeyPress);

        //

        // button23

        //

        this->button23->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

        static_cast<System::Byte>(0)));

        this->button23->Location = System::Drawing::Point(8, 323);

        this->button23->Name = L"button23";

```

```

this->button23->Size = System::Drawing::Size(40, 30);

this->button23->TabIndex = 24;

this->button23->Text = L"+/-";

this->button23->UseVisualStyleBackColor = true;

this->button23->Click += gcnew System::EventHandler(this,
&MyForm::button23_Click);

//

// button24

//

this->button24->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button24->Location = System::Drawing::Point(100, 323);

this->button24->Name = L"button24";

this->button24->Size = System::Drawing::Size(40, 30);

this->button24->TabIndex = 25;

this->button24->Text = L".";

this->button24->UseVisualStyleBackColor = true;

this->button24->Click += gcnew System::EventHandler(this,
&MyForm::button24_Click);

//

// button25

```

```

//

this->button25->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans          Serif",          12,          System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button25->Location = System::Drawing::Point(8, 179);

this->button25->Name = L"button25";

this->button25->Size = System::Drawing::Size(53, 30);

this->button25->TabIndex = 26;

this->button25->Text = L"1/x";

this->button25->UseVisualStyleBackColor = true;

this->button25->Click += gcnew System::EventHandler(this,
&MyForm::button25_Click);

//

// button26

//

this->button26->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans          Serif",          12,          System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,

static_cast<System::Byte>(0)));

this->button26->Location = System::Drawing::Point(199, 179);

this->button26->Name = L"button26";

this->button26->Size = System::Drawing::Size(53, 30);

this->button26->TabIndex = 27;

```

```

this->button26->Text = L"x^2";

this->button26->UseVisualStyleBackColor = true;

this->button26->Click += gcnew System::EventHandler(this,
&MyForm::button26_Click);

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Impact",
14.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(4, 9);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(204, 23);

this->label1->TabIndex = 28;

this->label1->Text = L"Выражение/Expression";

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Impact",
14.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));

```

```

this->label2->Location = System::Drawing::Point(4, 84);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(117, 23);

this->label2->TabIndex = 29;

this->label2->Text = L"Ответ/Return";

//

// textBox3

//

this->textBox3->Location = System::Drawing::Point(8, 110);

this->textBox3->Name = L"textBox3";

this->textBox3->Size = System::Drawing::Size(244, 20);

this->textBox3->TabIndex = 30;

this->textBox3->KeyPress += gcnew
System::Windows::Forms::KeyPressEventHandler(this, &MyForm::textBox3_KeyPress);

//

// button27

//

this->button27->Font = (gcnew System::Drawing::Font(L"Microsoft
Sans Serif", 12, System::Drawing::FontStyle::Regular,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));

this->button27->Location = System::Drawing::Point(67, 179);

this->button27->Name = L"button27";

```

```

this->button27->Size = System::Drawing::Size(53, 30);

this->button27->TabIndex = 31;

this->button27->Text = L"x^y";

this->button27->UseVisualStyleBackColor = true;

this->button27->Click += gcnew System::EventHandler(this,
&MyForm::button27_Click);

//

// button28

//

this->button28->Location = System::Drawing::Point(152, 84);

this->button28->Name = L"button28";

this->button28->Size = System::Drawing::Size(99, 22);

this->button28->TabIndex = 32;

this->button28->Text = L"Сохранить/Save";

this->button28->UseVisualStyleBackColor = true;

this->button28->Click += gcnew System::EventHandler(this,
&MyForm::button28_Click);

//

// MyForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;

```

```
this->ClientSize = System::Drawing::Size(261, 362);

this->Controls->Add(this->button28);

this->Controls->Add(this->button27);

this->Controls->Add(this->textBox3);

this->Controls->Add(this->label2);

this->Controls->Add(this->label1);

this->Controls->Add(this->button26);

this->Controls->Add(this->button25);

this->Controls->Add(this->button24);

this->Controls->Add(this->button23);

this->Controls->Add(this->textBox2);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->button22);

this->Controls->Add(this->button21);

this->Controls->Add(this->button20);

this->Controls->Add(this->button19);

this->Controls->Add(this->button18);

this->Controls->Add(this->button17);

this->Controls->Add(this->button16);

this->Controls->Add(this->button15);

this->Controls->Add(this->button14);

this->Controls->Add(this->button13);
```

```
this->Controls->Add(this->button12);

this->Controls->Add(this->button11);

this->Controls->Add(this->button10);

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->Name = L"MyForm";

this->Padding = System::Windows::Forms::Padding(10);

this->Text = L"MyForm";

this->ResumeLayout(false);

this->PerformLayout();
```

```
}
```

```
#pragma endregion
```

```
private: System::Void button15_Click(System::Object^ sender, System::EventArgs^ e)
```

```
{
```

```

if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

else if ((textBox1->Text != "") && (textBox2->Text != "")) {

    int oper1 = oper;

    oper = 7;

    calculate();

    oper = oper1;

    textBox1->Text = b.ToString();

    auto text = textBox1->Text;

    int lenght = textBox1->Text->Length;

    for (int i = 0; i < lenght; i++) {

        if (text[i] == ',') { zap = 1; };

    }

}

else if ((textBox1->Text == "") && (textBox2->Text == "")) {

    textBox3->Clear();

    oper = 7;

    textBox1->Text = b.ToString();

    textBox3->Text = "sqrt(" + b.ToString() + ")";

    calculate();

```

```

        textBox3->Text = textBox3->Text + " = " + b.ToString();

        textBox1->Clear();

    }

    else {

        oper = 7;

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

        zap = 0;

    }

}

private: System::Void button22_Click(System::Object^ sender, System::EventArgs^ e)
{

    textBox1->Text = "";

    textBox2->Text = "";

    textBox3->Text = "";

    zap = 0;

}

private: System::Void textBox1_TextChanged(System::Object^ sender,
System::EventArgs^ e) {

}

```

```

private: System::Void button23_Click(System::Object^ sender, System::EventArgs^ e)
{
    if (textBox1->Text->Contains("-"))
    {
        textBox1->Text = textBox1->Text->Replace("-", "");
    }
    else
    {
        textBox1->Text = "-" + textBox1->Text;
    }
}

private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 1;
}

private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 2;
}

private: System::Void button3_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 3;
}

```

```

private: System::Void button4_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 4;
}

private: System::Void button5_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 5;
}

private: System::Void button6_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 6;
}

private: System::Void button7_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 7;
}

private: System::Void button8_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 8;
}

private: System::Void button9_Click(System::Object^ sender, System::EventArgs^ e)
{
    textBox1->Text = textBox1->Text + 9;
}

```

```

private: System::Void button10_Click(System::Object^ sender, System::EventArgs^ e)
{

    textBox1->Text = textBox1->Text + 0;

}

private: System::Void button11_Click(System::Object^ sender, System::EventArgs^ e)
{

    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();

        textBox2->Text = b.ToString() + " + ";

        a = b;

        oper = 1;

    }

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        calculate();

        textBox1->Clear();

        zap = 0;

        textBox2->Text = b.ToString() + " + ";

```

```

DeleteLasttextBox2();

a = float::Parse(textBox2->Text);

textBox2->Text = b.ToString() + " + ";

oper = 1;

```

```

}

```

```

else {

```

```

    a = float::Parse(textBox1->Text);

    textBox1->Clear();

    zap = 0;

    oper = 1;

    textBox2->Text = a.ToString() + " + ";

```

```

}

```

```

}

```

```

private: System::Void button12_Click(System::Object^ sender, System::EventArgs^ e)
{
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
    "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();

        textBox2->Text = b.ToString() + " - ";

```

```

        a = b;

        oper = 2;

    }

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        calculate();

        textBox1->Clear();

        zap = 0;

        textBox2->Text = b.ToString() + " - ";

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        textBox2->Text = b.ToString() + " - ";

        oper = 2;

    }

    else {

        a = float::Parse(textBox1->Text);

        textBox1->Clear();

        zap = 0;

        oper = 2;

```

```
textBox2->Text = a.ToString() + " - ";
```

```
}
```

```
}
```

```
private: System::Void button13_Click(System::Object^ sender, System::EventArgs^ e)
```

```
{
```

```
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text == "")) {}
```

```
    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}
```

```
    else if ((textBox1->Text == "") && (textBox2->Text == "")) {
```

```
        textBox3->Clear();
```

```
        textBox2->Text = b.ToString() + " * ";
```

```
        a = b;
```

```
        oper = 3;
```

```
    }
```

```
    else if ((textBox1->Text != "") && (textBox2->Text != "")) {
```

```
        DeleteLasttextBox2();
```

```
        a = float::Parse(textBox2->Text);
```

```
        calculate();
```

```
        textBox1->Clear();
```

```
        zap = 0;
```

```
        textBox2->Text = b.ToString() + " * ";
```

```

DeleteLasttextBox2();

a = float::Parse(textBox2->Text);

textBox2->Text = b.ToString() + " * ";

oper = 3;

}

else {

    a = float::Parse(textBox1->Text);

    textBox1->Clear();

    zap = 0;

    oper = 3;

    textBox2->Text = a.ToString() + "*";

}

}

private: System::Void button14_Click(System::Object^ sender, System::EventArgs^ e)
{
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
    "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();

        textBox2->Text = b.ToString() + " / ";

```

```

        a = b;

        oper = 4;

    }

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        calculate();

        textBox1->Clear();

        zap = 0;

        textBox2->Text = b.ToString() + "/";

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        textBox2->Text = b.ToString() + " / ";

        oper = 4;

    }

    else {

        a = float::Parse(textBox1->Text);

        textBox1->Clear();

        zap = 0;

        oper = 4;

```

```

        textBox2->Text = a.ToString() + " / ";

    }

}

private: System::Void button24_Click(System::Object^ sender, System::EventArgs^ e)
{

    if (zap == 1) {}

    else if (textBox1->Text == "") {

        textBox1->Text = textBox1->Text + "0,";

        zap = 1;

    }

    else {

        textBox1->Text = textBox1->Text + ",";

        zap = 1;

    }

}

private: System::Void button20_Click(System::Object^ sender, System::EventArgs^ e)
{

    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {

        DeleteLasttextBox2();

```

```

        textBox3->Text = textBox2->Text + " = " + textBox2->Text;

        textBox2->Clear();

    }

    else if ((textBox1->Text != "") && (textBox2->Text == "")) {

        textBox3->Text = textBox1->Text + " = " + textBox1->Text;

        textBox1->Clear();

        zap = 0;

    }

    else {

        calculate();

        textBox3->Text = textBox2->Text + textBox1->Text + " = " +
b.ToString();

        textBox1->Clear();

        textBox2->Clear();

        zap = 0;

    }

}

private: System::Void button21_Click(System::Object^ sender, System::EventArgs^ e)
{

    int lenght = textBox1->Text->Length - 1;

    auto text = textBox1->Text;

```

```

        textBox1->Clear();

        zap = 0;

        for (int i = 0; i < lenght; i++) {

            textBox1->Text = textBox1->Text + text[i];

        }

        text = textBox1->Text;

        lenght = textBox1->Text->Length;

        for (int i = 0; i < lenght; i++) {

            if (text[i] == ',') { zap = 1; };

        }

    }

    private:      System::Void      textBox1_KeyPress(System::Object^      sender,
System::Windows::Forms::KeyPressEventArgs^ e) {

        if (e->KeyChar <= 47 || e->KeyChar >= 58)

            e->Handled = true;

    }

    private:      System::Void      textBox2_KeyPress(System::Object^      sender,
System::Windows::Forms::KeyPressEventArgs^ e) {

        e->Handled = true;

    }

    private:      System::Void      textBox3_KeyPress(System::Object^      sender,
System::Windows::Forms::KeyPressEventArgs^ e) {

        e->Handled = true;

```

```
}
```

```
private: System::Void button25_Click(System::Object^ sender, System::EventArgs^ e)
```

```
{
```

```
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text == "")) {}
```

```
    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}
```

```
    else if ((textBox1->Text != "") && (textBox2->Text != "")) {
```

```
        int oper1 = oper;
```

```
        oper = 5;
```

```
        calculate();
```

```
        oper = oper1;
```

```
        textBox1->Text = b.ToString();
```

```
        auto text = textBox1->Text;
```

```
        int lenght = textBox1->Text->Length;
```

```
        for (int i = 0; i < lenght; i++) {
```

```
            if (text[i] == ',') { zap = 1; };
```

```
        }
```

```
    }
```

```
    else if ((textBox1->Text == "") && (textBox2->Text == "")) {
```

```
        textBox3->Clear();
```

```
        oper = 5;
```

```
        textBox1->Text = b.ToString();
```

```

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

    }

    else {

        oper = 5;

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

        zap = 0;

    }

}

private: System::Void button26_Click(System::Object^ sender, System::EventArgs^ e)
{
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        int oper1 = oper;

        oper = 6;

        calculate();

```

```

        oper = oper1;

        textBox1->Text = b.ToString();

        auto text = textBox1->Text;

        int lenght = textBox1->Text->Length;

        for (int i = 0; i < lenght; i++) {

            if (text[i] == ',') { zap = 1; };

        }

    }

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();

        oper = 6;

        textBox1->Text = b.ToString();

        textBox3->Text = b.ToString();

        calculate();

        textBox3->Text = textBox3->Text+ "^2 = " + b.ToString();

        textBox1->Clear();

    }

    else {

        oper = 6;

```

```

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

        zap = 0;

    }

}

private: System::Void button19_Click(System::Object^ sender, System::EventArgs^ e)
{
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
    "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        int oper1 = oper;

        oper = 8;

        calculate();

        oper = oper1;

        textBox1->Text = b.ToString();

        auto text = textBox1->Text;

        int lenght = textBox1->Text->Length;

        for (int i = 0; i < lenght; i++) {

            if (text[i] == ',') { zap = 1; };

        }
    }
}

```

```
}
```

```
else if ((textBox1->Text == "") && (textBox2->Text == "")) {
```

```
    textBox3->Clear();
```

```
    oper = 8;
```

```
    textBox1->Text = b.ToString();
```

```
    calculate();
```

```
    textBox3->Text = b.ToString();
```

```
    textBox1->Clear();
```

```
}
```

```
else {
```

```
    oper = 8;
```

```
    calculate();
```

```
    textBox3->Text = b.ToString();
```

```
    textBox1->Clear();
```

```
    zap = 0;
```

```
}
```

```
}
```

```
private: System::Void button16_Click(System::Object^ sender, System::EventArgs^ e)
```

```
{
```

```

if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

else if ((textBox1->Text != "") && (textBox2->Text != "")) {

    int oper1 = oper;

    oper = 9;

    calculate();

    oper = oper1;

    textBox1->Text = b.ToString();

    auto text = textBox1->Text;

    int lenght = textBox1->Text->Length;

    for (int i = 0; i < lenght; i++) {

        if (text[i] == ',') { zap = 1; };

    }

}

else if ((textBox1->Text == "") && (textBox2->Text == "")) {

    textBox3->Clear();

    oper = 9;

    textBox1->Text = b.ToString();

    calculate();

    textBox3->Text = b.ToString();

```

```

        textBox1->Clear();

    }

    else {

        oper = 9;

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

        zap = 0;

    }

}

private: System::Void button17_Click(System::Object^ sender, System::EventArgs^ e)
{

    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        int oper1 = oper;

        oper = 10;

        calculate();

        oper = oper1;

        textBox1->Text = b.ToString();

```

```

        auto text = textBox1->Text;

        int lenght = textBox1->Text->Length;

        for (int i = 0; i < lenght; i++) {

            if (text[i] == ',') { zap = 1; };

        }

    }

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();

        oper = 10;

        textBox1->Text = b.ToString();

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

    }

    else {

        oper = 10;

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

        zap = 0;

```

```

    }

}

private: System::Void button18_Click(System::Object^ sender, System::EventArgs^ e)
{
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        int oper1 = oper;

        oper = 11;

        calculate();

        oper = oper1;

        textBox1->Text = b.ToString();

        auto text = textBox1->Text;

        int lenght = textBox1->Text->Length;

        for (int i = 0; i < lenght; i++) {

            if (text[i] == ',') { zap = 1; };

        }

    }

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();
    }
}

```

```

        oper = 11;

        textBox1->Text = b.ToString();

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

    }

    else {

        oper = 11;

        calculate();

        textBox3->Text = b.ToString();

        textBox1->Clear();

        zap = 0;

    }

}

private: System::Void button27_Click(System::Object^ sender, System::EventArgs^ e)
{
    if ((textBox1->Text == "") && (textBox2->Text == "") && (textBox3->Text ==
"")) {}

    else if ((textBox1->Text == "") && (textBox2->Text != "")) {}

    else if ((textBox1->Text == "") && (textBox2->Text == "")) {

        textBox3->Clear();

```

```

        textBox2->Text = b.ToString() + " ^ ";

        a = b;

        oper = 12;

    }

    else if ((textBox1->Text != "") && (textBox2->Text != "")) {

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        calculate();

        textBox1->Clear();

        zap = 0;

        textBox2->Text = b.ToString() + "^";

        DeleteLasttextBox2();

        a = float::Parse(textBox2->Text);

        textBox2->Text = b.ToString() + "^";

        oper = 12;

    }

    else {

        a = float::Parse(textBox1->Text);

        textBox1->Clear();

        zap = 0;

```

```

        oper = 12;

        textBox2->Text = a.ToString() + "^";

    }

}

private: System::Void button28_Click(System::Object^ sender, System::EventArgs^ e)
{

    String^ FileName = "";

    if (saveFileDialog1->ShowDialog() == Windows::Forms::DialogResult::OK) {

        FileName = saveFileDialog1->FileName;

        myFunc((char*)(Marshal::StringToHGlobalAnsi(System::Convert::ToString(textBox3-
>Text))).ToPointer(),
(char*)(Marshal::StringToHGlobalAnsi(System::Convert::ToString(FileName))).ToPointer()
);

    }

}

};

}

```

Файл MyForm.cpp

```
#include "MyForm.h"
```

```
#include "MyForm1.h"
```

```
#include <Windows.h>
```

```

using namespace Project3; // Название проекта

float a, b;

int oper;

[STAThreadAttribute]

int WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int) {

    Application::EnableVisualStyles();

    Application::SetCompatibleTextRenderingDefault(false);

    MyForm1^ first = gcnew MyForm1();

    DateTime end = DateTime::Now + TimeSpan::FromSeconds(3);

    first->Show();

    while (end > DateTime::Now) {

        Application::DoEvents();

    }

    first->Close();

    //first->Dispose();

    Application::Run(gcnew MyForm);

    return 0;

}

```

Файл MyForm1.h

```
#pragma once
```

```
namespace Project3 {
```

```
    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 MyForm1
```

```
    /// </summary>
```

```
    public ref class MyForm1 : public System::Windows::Forms::Form
```

```
    {
```

```
    public:
```

```
        MyForm1(void)
```

```
        {
```

```
            InitializeComponent();
```

```
            //
```

```
            //TODO: добавьте код конструктора
```

```
//  
  
timer1->Start();  
  
}
```

protected:

```
/// <summary>  
  
/// Освободить все используемые ресурсы.  
  
/// </summary>  
  
~MyForm1()  
  
{  
  
    if (components)  
    {  
  
        delete components;  
  
    }  
  
}
```

private: System::Windows::Forms::PictureBox^ pictureBox1;

private: System::Windows::Forms::Timer^ timer1;

private: System::ComponentModel::IContainer^ components;

protected:

private:

```
/// <summary>
```

```
/// Обязательная переменная конструктора.
```

```
/// </summary>
```

```
#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->components = (gcnew  
System::ComponentModel::Container());
```

```
    System::ComponentModel::ComponentResourceManager^ resources  
= (gcnew System::ComponentModel::ComponentResourceManager(MyForm1::typeid));
```

```
    this->pictureBox1 = (gcnew  
System::Windows::Forms::PictureBox());
```

```
    this->timer1 = (gcnew System::Windows::Forms::Timer(this-  
>components));
```

```
    (cli::safe_cast<System::ComponentModel::ISupportInitialize^>(this-  
>pictureBox1))->BeginInit();
```

```
    this->SuspendLayout();
```

```
    //
```

```
    // pictureBox1
```

```

//

this->pictureBox1->Image =
(cli::safe_cast<System::Drawing::Image^>(resources->GetObject(L"pictureBox1.Image"))));

this->pictureBox1->Location = System::Drawing::Point(1, 0);

this->pictureBox1->Name = L"pictureBox1";

this->pictureBox1->Size = System::Drawing::Size(640, 360);

this->pictureBox1->SizeMode =
System::Windows::Forms::PictureBoxSizeMode::StretchImage;

this->pictureBox1->TabIndex = 0;

this->pictureBox1->TabStop = false;

//

// timer1

//

this->timer1->Tick += gcnew System::EventHandler(this,
&MyForm1::timer1_Tick);

//

// MyForm1

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 365);

this->Controls->Add(this->pictureBox1);

```

```

        this->FormBorderStyle =
System::Windows::Forms::FormBorderStyle::None;

        this->Name = L"MyForm1";

        this->Opacity = 0;

        this->StartPosition =
System::Windows::Forms::FormStartPosition::CenterScreen;

        this->Text = L"MyForm1";

        (cli::safe_cast<System::ComponentModel::ISupportInitialize^(this-
>pictureBox1))->EndInit();

        this->ResumeLayout(false);

    }

#pragma endregion

private: System::Void timer1_Tick(System::Object^ sender, System::EventArgs^
e) {

    Opacity += 0.1;

}

};

}

```

Файл MyForm1.cpp

```
#include "MyForm1.h"
```

Файл Source.asm

```
includelib legacy_stdio_definitions.lib
```

.386

.model flat,stdcall

extern fprintf:near

extern fopen:near

extern fclose:near

.data

zapis db "w+", 0

zapis1 db "%s", 0

.code

myFunc proc C, stroka:DWORD, filename:DWORD

mov eax, stroka

push offset zapis

push filename

call fopen

add esp, 8

push eax

```
mov edx, stroka  
  
push edx  
  
push offset zapis1  
  
push eax  
  
call fprintf  
  
pop eax  
  
  
add esp, 8  
  
push eax  
  
call fclose  
  
add esp, 8  
  
  
ret  
  
myFunc endp  
  
end
```