Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Глазовский государственный педагогический институт имени В.Г. Короленко»

**ОТЧЕТ**

**по учебной практике**

**ПМ 03 Участие в интеграции программных модулей**

**Специальность 09.02.03 Программирование в компьютерных системах**

|  |
| --- |
| Выполнил(а): |
| студент(ка) ОП СПО группа 31 |
| Поздеев И.Д. |
|  |
| Руководитель практики: ст. преподаватель Касаткин К.А. |

г. Глазов 2019

Оглавление

[Введение 3](#_Toc4161122)

[Цель разработки 4](#_Toc4161123)

[Обзор аналогов 5](#_Toc4161124)

[Постановка задачи 6](#_Toc4161125)

[Описание алгоритма работы программы 7](#_Toc4161126)

[Исходный код программы 8](#_Toc4161127)

[Тестирование программы 16](#_Toc4161128)

[Руководство по установке 16](#_Toc4161129)

# Введение

Калькулятор заменил механические вычислительные устройства, такие, как абаки, счёты, логарифмические линейки, механические или электромеханические арифмометры, а также математические таблицы (прежде всего — таблицы логарифмов).

В зависимости от возможностей и целевой сферы применения калькуляторы делятся на простейшие, бухгалтерские, инженерные (научные), финансовые. В отдельные классы обычно выделяют программируемые калькуляторы, дающие возможность выполнения сложных вычислений по предварительно заложенной программе, а также графические — поддерживающие построение и отображение графиков. Специализированные калькуляторы предназначены для выполнения вычислений в достаточно узкой сфере (статистические, медицинские, специальные финансовые расчёты и т. п.); такие калькуляторы сейчас чаще реализуются в виде программ для универсальных персональных компьютеров, КПК, планшетов, хотя могут изготавливаться и «в железе».

По исполнению калькуляторы могут быть настольными или компактными (карманными). Отдельные модели имеют интерфейсы для подключения персонального компьютера, печатающего устройства, внешнего модуля памяти или иных внешних устройств. Существуют калькуляторы, встроенные в персональные компьютеры, сотовые телефоны, КПК и даже наручные часы. Специализированные калькуляторы встраиваются в бытовую технику (например, простой медицинский калькулятор может встраиваться в спортивный тренажёр).

Приложения реализована для демонстрации работы для имитирования работы и выполнения операций над числами или алгебраическими формулами на языке C# в среде разработки Visual Studio 2013/

# Цель разработки

Создать приложения калькулятор для имитирования работы и выполнения операций над числами или алгебраическими формулами.

# Обзор аналогов

<https://calculator888.ru/> - Калькулятор онлайн для расчетов на работе, дома или учебе.

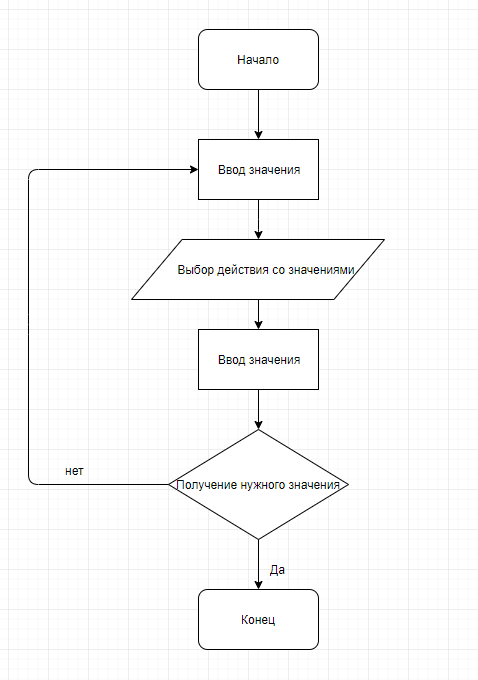
<https://calc.by/math-calculators/scientific-calculator.html> - Инженерный онлайн калькулятор.

# Постановка задачи

Задача – создать приложения калькулятор имитирующий работу обычного карманного калькулятора и выполнения операций над числами или алгебраическими формулами. Операции вычисления значений.

# Описание алгоритма работы программы

1. Открытие приложения.
2. Ввод значения.
3. Выбор действия со значениями (операцаия над ними).
4. Ввод значения.
5. Получили нужное значения если нет то вернуться к шагу 2?.



# Исходный код программы

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace calculator

{

public partial class Form1 : Form

{

Calc C;

int k; //количество нажатий кнопки MRC

public Form1()

{

InitializeComponent();

C = new Calc();

labelNumber.Text = "0";

}

//кнопка Очистка (CE)

private void buttonClear\_Click(object sender, EventArgs e)

{

labelNumber.Text = "0";

C.Clear\_A();

FreeButtons();

k = 0;

}

//кнопка изменения знака у числа

private void buttonChangeSign\_Click(object sender, EventArgs e)

{

if (labelNumber.Text[0] == '-')

labelNumber.Text = labelNumber.Text.Remove(0, 1);

else

labelNumber.Text = "-" + labelNumber.Text;

}

private void buttonPoint\_Click(object sender, EventArgs e)

{

if ((labelNumber.Text.IndexOf(",") == -1) && (labelNumber.Text.IndexOf("∞") == -1))

labelNumber.Text += ",";

}

private void button0\_Click(object sender, EventArgs e)

{

labelNumber.Text += "0";

CorrectNumber();

}

private void button1\_Click(object sender, EventArgs e)

{

labelNumber.Text += "1";

CorrectNumber();

}

private void button2\_Click(object sender, EventArgs e)

{

labelNumber.Text += "2";

CorrectNumber();

}

private void button3\_Click(object sender, EventArgs e)

{

labelNumber.Text += "3";

CorrectNumber();

}

private void button4\_Click(object sender, EventArgs e)

{

labelNumber.Text += "4";

CorrectNumber();

}

private void button5\_Click(object sender, EventArgs e)

{

labelNumber.Text += "5";

CorrectNumber();

}

private void button6\_Click(object sender, EventArgs e)

{

labelNumber.Text += "6";

CorrectNumber();

}

private void button7\_Click(object sender, EventArgs e)

{

labelNumber.Text += "7";

CorrectNumber();

}

private void button8\_Click(object sender, EventArgs e)

{

labelNumber.Text += "8";

CorrectNumber();

}

private void button9\_Click(object sender, EventArgs e)

{

labelNumber.Text += "9";

CorrectNumber();

}

//удаляем лишний ноль впереди числа, если таковой имеется

private void CorrectNumber()

{

//если есть знак "бесконечность" - не даёт писать цифры после него

if (labelNumber.Text.IndexOf("∞") != -1)

labelNumber.Text = labelNumber.Text.Substring(0, labelNumber.Text.Length - 1);

//ситуация: слева ноль, а после него НЕ запятая, тогда ноль можно удалить

if (labelNumber.Text[0] == '0' && (labelNumber.Text.IndexOf(",") != 1))

labelNumber.Text = labelNumber.Text.Remove(0, 1);

//аналогично предыдущему, только для отрицательного числа

if (labelNumber.Text[0] == '-')

if (labelNumber.Text[1] == '0' && (labelNumber.Text.IndexOf(",") != 2))

labelNumber.Text = labelNumber.Text.Remove(1, 1);

}

//кнопка Равно

private void buttonCalc\_Click(object sender, EventArgs e)

{

if (!buttonMult.Enabled)

labelNumber.Text = C.Multiplication(Convert.ToDouble(labelNumber.Text)).ToString();

if (!buttonDiv.Enabled)

labelNumber.Text = C.Division(Convert.ToDouble(labelNumber.Text)).ToString();

if (!buttonPlus.Enabled)

labelNumber.Text = C.Sum(Convert.ToDouble(labelNumber.Text)).ToString();

if (!buttonMinus.Enabled)

labelNumber.Text = C.Subtraction(Convert.ToDouble(labelNumber.Text)).ToString();

if (!buttonSqrtX.Enabled)

labelNumber.Text = C.SqrtX(Convert.ToDouble(labelNumber.Text)).ToString();

if (!buttonDegreeY.Enabled)

labelNumber.Text = C.DegreeY(Convert.ToDouble(labelNumber.Text)).ToString();

C.Clear\_A();

FreeButtons();

k = 0;

}

//кнопка Умножение

private void buttonMult\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

buttonMult.Enabled = false;

labelNumber.Text = "0";

}

}

//кнопка Деление

private void buttonDiv\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

buttonDiv.Enabled = false;

labelNumber.Text = "0";

}

}

//кнопка Сложение

private void buttonPlus\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

buttonPlus.Enabled = false;

labelNumber.Text = "0";

}

}

//кнопка Вычитание

private void buttonMinus\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

buttonMinus.Enabled = false;

labelNumber.Text = "0";

}

}

//кнопка Корень произвольной степени

private void buttonSqrtX\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

buttonSqrtX.Enabled = false;

labelNumber.Text = "0";

}

}

//кнопка Возведение в произвольную степень

private void buttonDegreeY\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

buttonDegreeY.Enabled = false;

labelNumber.Text = "0";

}

}

//кнопка Корень квадратный

private void buttonSqrt\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

labelNumber.Text = C.Sqrt().ToString();

C.Clear\_A();

FreeButtons();

}

}

//кнопка Квадрат числа

private void buttonSquare\_Click(object sender, EventArgs e)

{

if (CanPress())

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

labelNumber.Text = C.Square().ToString();

C.Clear\_A();

FreeButtons();

}

}

//кнопка Факториал

private void buttonFactorial\_Click(object sender, EventArgs e)

{

if (CanPress())

{

if ((Convert.ToDouble(labelNumber.Text) == (int)(Convert.ToDouble(labelNumber.Text))) &&

((Convert.ToDouble(labelNumber.Text) >= 0.0)))

{

C.Put\_A(Convert.ToDouble(labelNumber.Text));

labelNumber.Text = C.Factorial().ToString();

C.Clear\_A();

FreeButtons();

}

else

MessageBox.Show("Число должно быть >= 0 и целым!");

}

}

//кнопка М+

private void buttonMPlus\_Click(object sender, EventArgs e)

{

C.M\_Sum(Convert.ToDouble(labelNumber.Text));

}

//кнопка М-

private void buttonMMinus\_Click(object sender, EventArgs e)

{

C.M\_Subtraction(Convert.ToDouble(labelNumber.Text));

}

//кнопка М\*

private void buttonMMult\_Click(object sender, EventArgs e)

{

C.M\_Multiplication(Convert.ToDouble(labelNumber.Text));

}

//кнопка М/

private void buttonMDiv\_Click(object sender, EventArgs e)

{

C.M\_Division(Convert.ToDouble(labelNumber.Text));

}

//кнопка МRC

private void buttonMRC\_Click(object sender, EventArgs e)

{

if (CanPress())

{

k++;

if (k == 1)

labelNumber.Text = C.MemoryShow().ToString();

if (k == 2)

{

C.Memory\_Clear();

labelNumber.Text = "0";

k = 0;

}

}

}

//проверяет не нажата ли еще какая-либо из кнопок мат.операций

private bool CanPress()

{

if (!buttonMult.Enabled)

return false;

if (!buttonDiv.Enabled)

return false;

if (!buttonPlus.Enabled)

return false;

if (!buttonMinus.Enabled)

return false;

if (!buttonSqrtX.Enabled)

return false;

if (!buttonDegreeY.Enabled)

return false;

return true;

}

//снятие нажатия всех кнопок мат.операций

private void FreeButtons()

{

buttonMult.Enabled = true;

buttonDiv.Enabled = true;

buttonPlus.Enabled = true;

buttonMinus.Enabled = true;

buttonSqrtX.Enabled = true;

buttonDegreeY.Enabled = true;

}

private void labelNumber\_Click(object sender, EventArgs e)

{

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void groupBox1\_Enter(object sender, EventArgs e)

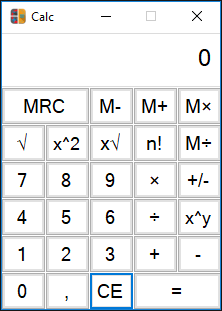
{

}

}

}

# Тестирование программы



# Руководство по установке

Открыть скаченный файл, запустить calculator.exe.