Лабораторная работа №4

«Программирование алгоритмов разветвляющихся структур с использованием поиска максимального и минимального значений»

Москва, 2019

# Оглавление

[Оглавление 2](#_Toc26983147)

[1 Задание 3](#_Toc26983148)

[1.1 Условие 3](#_Toc26983149)

[2 Выполнение работы 3](#_Toc26983150)

[2.1 Формализация и уточнение задания 3](#_Toc26983151)

[2.2 Создание класса Lw4c. 3](#_Toc26983152)

[2.2.1 Создание метода Ex 4](#_Toc26983153)

[2.2.2 Код класса LW4c 5](#_Toc26983154)

[2.3 Основная программа 6](#_Toc26983155)

[2.4 Создание формы-заставки 7](#_Toc26983156)

[2.5 Форма с решением 8](#_Toc26983157)

[Список источников 9](#_Toc26983158)

# Задание

## Условие

Вычислить значение функции z(a,b,c,x,y) ,если

(1)

Вычисления производить в программе Visual Studio на языке C# с помощью создания методов и классов. В нашей работе необходимо использовать 2 класса и 3 метода:

Класс VV, в котором будет 2 метода Vvod(ввод данных) и Vivod(вывод данных). И класс LW4c, в котором будет метод Ex для проверки и поиска подходящего условия вводимых данных, а также их вычисления по формулам представленным ранее.

# Выполнение работы

## Формализация и уточнение задания

Данная задача должна быть реализована с помощью класса и метода с разветвляющимися условиями Ex. Для выполнения задания и нахождения результата функции, необходим ввод трех переменных (a,b,c,x,y). Из которых, на условие разветвляющегося алгоритма будет влиять переменная x.

## Создание класса Lw4c.

Создадим класс LW4c, содержащий метод Ex, алгоритм класса представлен на рисунке 1.Ы



Рисунок 1 – Класс LW4c

### Создание метода Ex

Создав метод Ex мы сможем использовать каждую полученную переменную для проверки условий. После чего, проведем арифметические действия над числами в подходящем условии. Алгоритм метода Ex представлен на рисунке 2.



Рисунок 2 – Метод Ex

### Код класса LW4c

public class LW4C

{

public double EX(double a,double b,double c,double x,double y)

{

if (x>=3 & x <= 4)

{

return Math.Sqrt(b \* b + c \* c);

}

else if (x < 3)

{

return Math.Min(a, Math.Max(x \* x,Math.Max( y, c)));

}

else

{

return Math.Max(a \* x + c, Math.Pow(y, 3));

}

}

}

## Основная программа

Алгоритм представлен на рисунке 3, код после него.



Рисунок 3 – Алгоритм основной 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;

using LWlib;

namespace WindowsFormsApp1.LW4

{

public partial class LW4 : Form

{

public LW4()

{

InitializeComponent();

}

private void Button1\_Click(object sender, EventArgs e)

{

LWlib.VV x=new VV();

LWlib.LW4C a=new LW4C();

x.Vivod(a.EX(x.Vvod(textBox1), x.Vvod(textBox2), x.Vvod(textBox3), x.Vvod(textBox4), x.Vvod(textBox5)),textBox6);

}

private void Button2\_Click(object sender, EventArgs e)

{

Form x = new title();

x.Show();

this.Close();

}

}

}

## Создание формы-заставки

Создаём исчезающую заставку, сама заставка представлена на рисунке 4. Код представлен после рисунка.

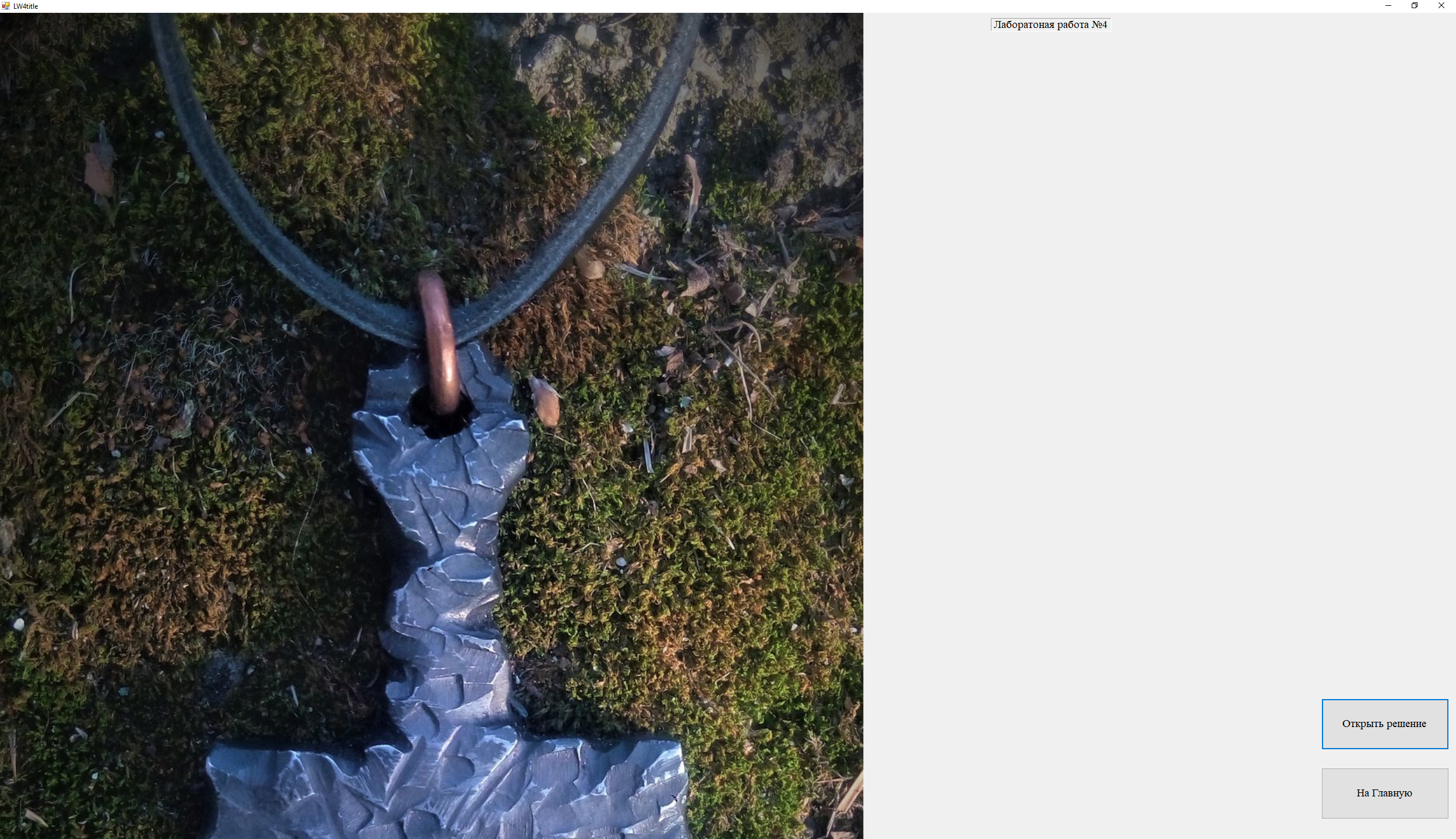


Рисунок 4 – Форма-заставка

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 WindowsFormsApp1.LW4

{

public partial class LW4title : Form

{

public LW4title()

{

InitializeComponent();

}

private void LW4title\_Paint(object sender, PaintEventArgs e)

{

Image my\_Image = Image.FromFile(@"C:\Users\super pc\Documents\GitHub\1-sem-c-\WindowsFormsApp1\3.jpg");

Graphics my\_Graphics = Graphics.FromImage(my\_Image);

Graphics f = e.Graphics;

f.DrawImage(my\_Image, new PointF(0.0F, 0.0F));

my\_Graphics.Dispose();

}

private void Timer1\_Tick(object sender, EventArgs e)

{

this.Opacity = this.Opacity - 0.2;

if (this.Opacity == 0)

{

this.timer1.Enabled = false;

Form f = new LW4();

this.Hide();

f.ShowDialog();

}

}

private void Button1\_Click(object sender, EventArgs e)

{

timer1.Start();

}

private void LW4title\_Load(object sender, EventArgs e)

{

}

private void Button2\_Click(object sender, EventArgs e)

{

Form l = new title();

l.Show();

this.Close();

}

}

}

## Форма с решением

Код формы с решение представлен в алгоритме основной программы, внешний вид и пример решения представлен на рисунке 5.

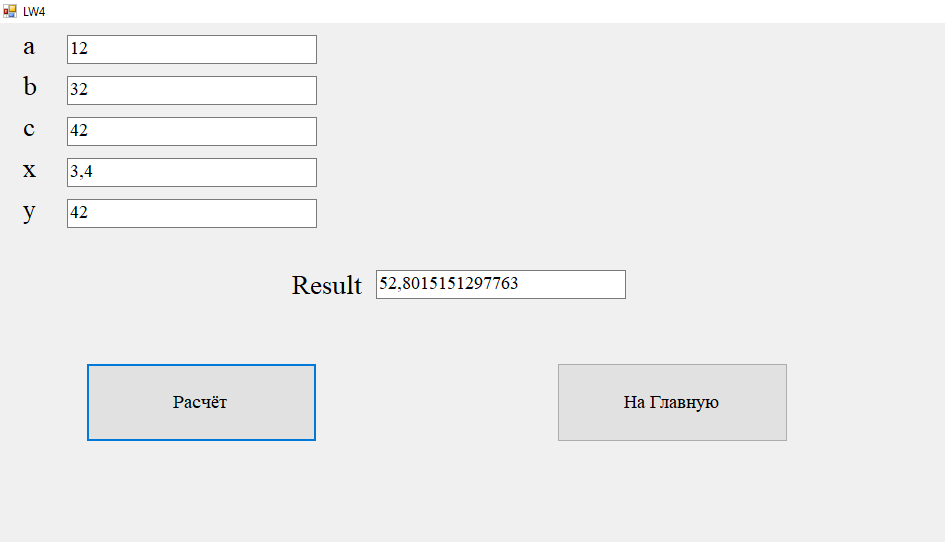


Рисунок 5 – Форма с решением

# Список источников

1 Гуриков С.Р. Введение в программирование на языке Visual C# : учебное пособие / С.Р. Гуриков – М.: ФОРУМ : ИНФРА-М,2013. – 448c. – (Высшее образование. Бакалавриат).

2 ГОСТ 7.32-2017 «Отчёт о научно-исследовательской работе» ([http://docs.cntd.ru/document/1200157208](https://vk.com/away.php?to=http%3A%2F%2Fdocs.cntd.ru%2Fdocument%2F1200157208&cc_key=))   
3 ГОСТ 7.1-2003 «Библиографическая запись. Библиографическое описание» ([http://docs.cntd.ru/document/1200034383](https://vk.com/away.php?to=http%3A%2F%2Fdocs.cntd.ru%2Fdocument%2F1200034383&cc_key=))   
4 ГОСТ 7.82-2001 «Библиографическая запись. Библиографическое описание электронных ресурсов» ([http://docs.cntd.ru/document/1200025968](https://vk.com/away.php?to=http%3A%2F%2Fdocs.cntd.ru%2Fdocument%2F1200025968&cc_key=))