

Учреждение образования

«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра электронных вычислительных машин

Системное программное обеспечение вычислительных
Машин

Описание к лабораторной работе № 1 “Система управления версиями Git”

Выполнил: Попов И. Д.
студент заочного обучения
группа № 050541

Проверил старший преподаватель: Смирнов Ю. В.

Минск 2023

Цели работы:

- 1) Изучить порядок работы с Git
- 2) Научиться выполнять основные операции с репозиторием

Инструментарий:

- 1) Git
- 2) Microsoft Visual Studio
- 3) NUnit

Порядок выполнения работы:

- 1) Получить начальный проект, который обрабатывает и выводит фрактал. По умолчанию проект должен браться с репозитория <https://github.com/trtpo/labal>, но в виду другой IDE и другого языка он был написан с нуля и находится <https://github.com/Ivan-Popov-050541/trtpo-start>.
- 2) Выполнить коммит с добавлением редактирования изображения по гамме
- 3) Выполнить коммит с добавлением нового уравнения фрактала, в данной работе был добавлен фрактал Жулиа.
- 4) Выполнить коммит с добавлением дополнительных операций с комплексными числами. Были добавлены возведение в степень и вычитание.
- 5) Выполнить коммит с добавлением Unit-тестов для проверки новых операций.
- 6) Выполнить push в свой репозиторий на GitHub. Проект лежит по адресу <https://github.com/Ivan-Popov-050541/trtpo/>.

Листинг кода и работы проекта:

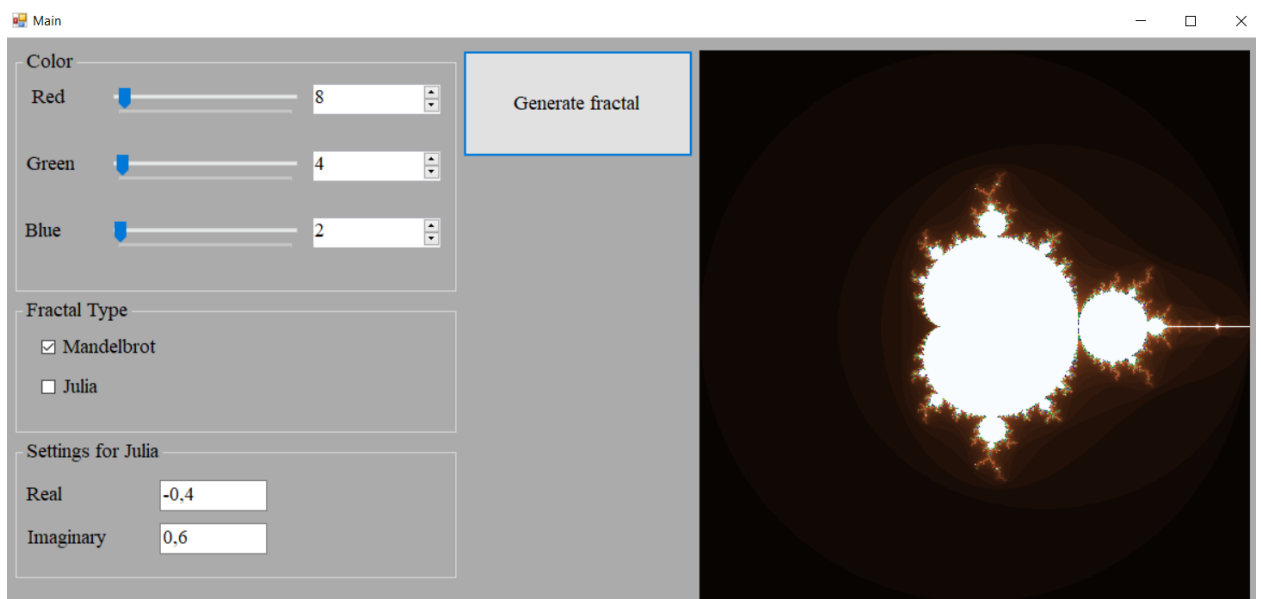


Рис. 1 – фрактал Мандельброта

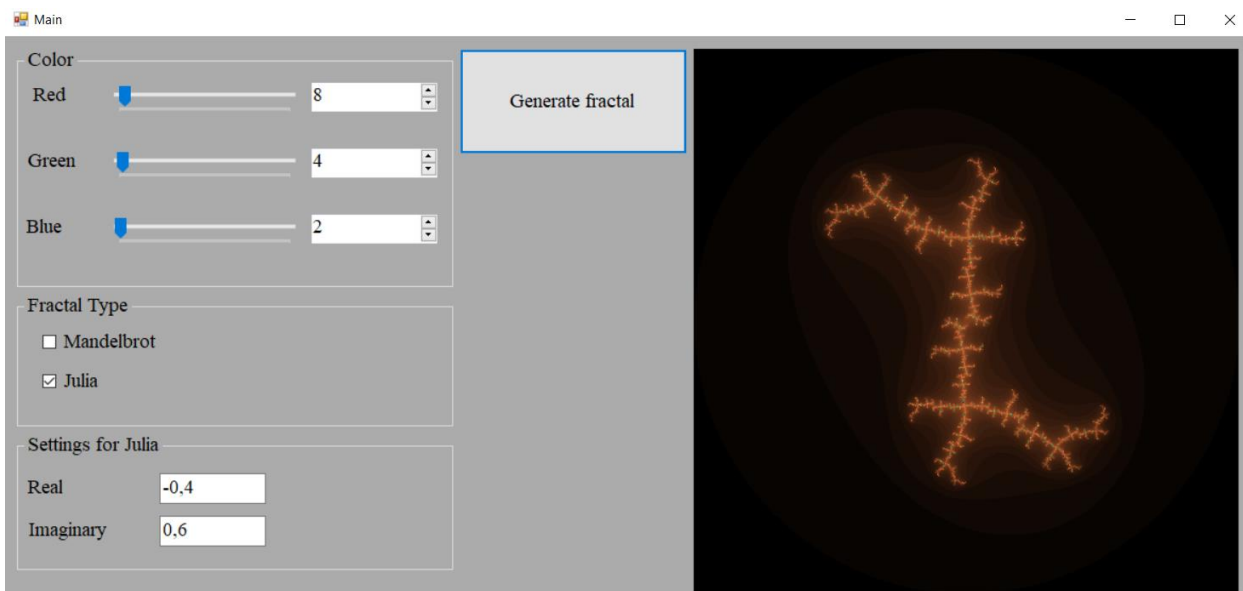


Рис. 2 – фрактал Жулиа

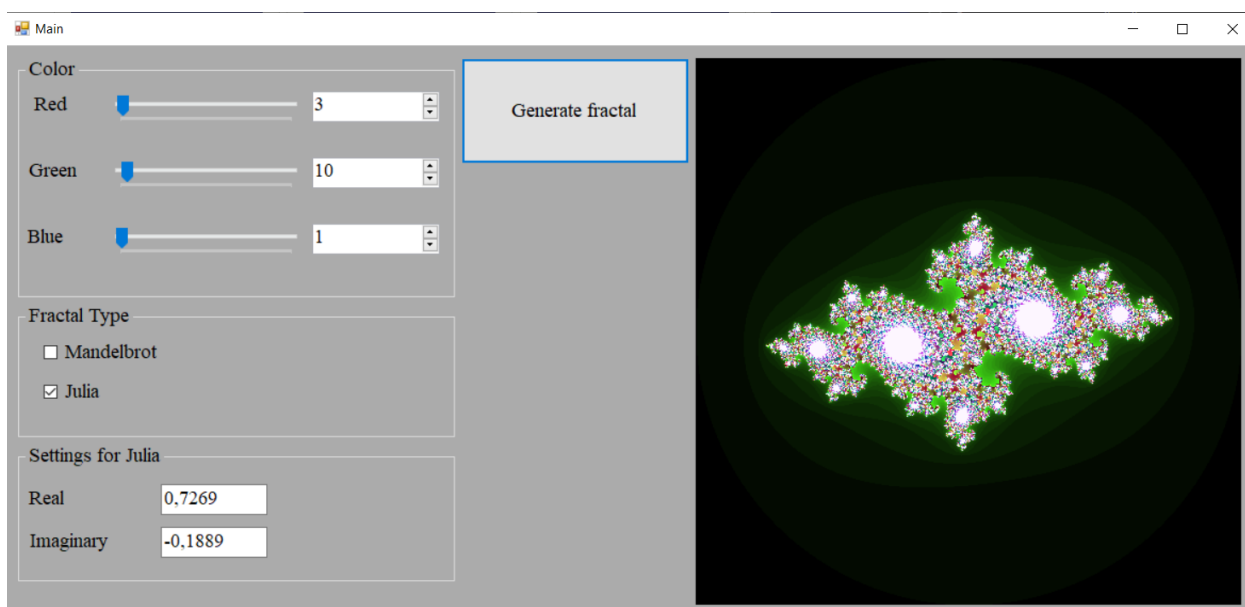


Рис. 3 – фрактал Жулиа

Ввиду применения новой операции вычитания в формуле расчета фрактала, текущие фракталы имеют несоответствие с обычным набором Жулиа: https://en.wikipedia.org/wiki/Julia_set. Можно откатиться на четвертый коммит или в коде изменить формулу на прибавление коэффициента. (см. рис. 5)

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace trtpo_fractal_lab1
{
    Ссылка: 26
    public class Complex
    {
        private double re;
        private double im;

        Ссылка: 15
        public double Re { get => re; set => re = value; }
        Ссылка: 15
        public double Im { get => im; set => im = value; }

        Ссылка: 11
        public Complex(double r1, double ig)
        {
            Re = r1;
            Im = ig;
        }

        Ссылка: 0
        public static Complex operator +(Complex a, Complex b) => new Complex(a.Re + b.Re, a.Im + b.Im);
        Ссылка: 3
        public static Complex operator -(Complex a, Complex b) => new Complex(a.Re - b.Re, a.Im - b.Im);
        Ссылка: 0
        public static Complex operator *(Complex a, Complex b) => new Complex(a.Re * b.Re - a.Im * b.Im, 2 * a.Re * b.Im);
        Ссылка: 2
        public double getAbs() => Re * Re + Im * Im;
        Ссылка: 3
        public Complex sqr() => new Complex(Re * Re - Im * Im, 2 * Re * Im);
    }
}

```

Рис. 4 – класс, реализующий работу с комплексными числами

```

Ссылка: 1
int getMandelbrod(int x, int y)
{
    Complex z = new Complex(0, 0);
    Complex c = new Complex(x * 4.0 / FractalPictureBox.Width - 2.0, y * 4.0 / FractalPictureBox.Height - 2.0);
    int i = 0;
    while (i < 255 && z.getAbs() < 4.0)
    {
        z = z.sqr() - c;
        ++i;
    }
    return i;
}

Ссылка: 1
int getJulia(int x, int y)
{
    Complex z = new Complex(x * 4.0 / FractalPictureBox.Width - 2.0, y * 4.0 / FractalPictureBox.Height - 2.0);
    Complex c = new Complex(Convert.ToDouble(RealTextBox.Text), Convert.ToDouble(ImaginaryTextBox.Text));
    int i = 0;
    while (i < 255 && z.getAbs() < 4.0)
    {
        z = z.sqr() - c;
        ++i;
    }
    return i;
}

```

Рис. 5 – методы для расчета фракталов.

Вывод:

Были изучены базовые принципы работы с Git: создание репозитория, коммитов, работа с удаленными репозиториями.