

Университет им. Н.Э. Баумана

Факультет Радиотехнический

Кафедра РТ5

Курс «Парадигмы и конструкции языков программирования»

**Отчет по лабораторной работе №5
«Лабораторная работа №1 по C#»**

Выполнил: Кудрявцев Р. В.
Студент группы: РТ5-31Б

Проверяющий: Гапанюк Ю.Е.
Доцент кафедры ИУ5

Москва, 2023 г.

Описание задания

Разработать программу для решения биквадратного уравнения.

1. Программа должна быть разработана в виде консольного приложения на языке C#.
2. Программа осуществляет ввод с клавиатуры коэффициентов A, B, C, вычисляет дискриминант и корни уравнения (в зависимости от дискриминанта).
3. Если коэффициент A, B, C введен некорректно (не приводится к действительному числу), то необходимо проигнорировать некорректное значение и ввести коэффициент повторно.
4. Корни уравнения выводятся зеленым цветом. Если корней нет, то сообщение выводится красным цветом.
5. Коэффициенты A, B, C задаются в виде параметров командной строки. Если они не указаны, то вводятся с клавиатуры в соответствии с пунктом 2. Проверка из пункта 3 в этом случае производится для параметров командной строки без повторного ввода с клавиатуры.

Текст программы

```
using System;
using System.ComponentModel.Design.Serialization;
using System.Runtime.InteropServices;

class BiquadraticEquation
{
    protected double a;
    protected double b;
    protected double c;
    protected int num_roots;
    protected HashSet<double> roots = new HashSet<double>();

    public BiquadraticEquation(string[] coefs, double a = 0, double b = 0, double c
= 0)
    {
        coef_setter(coefs, a, b, c);
    }
    protected void coef_setter(string[] coefs, double a, double b, double c)
    {
        if (coefs.Length == 0)
        {
            if (a == 0 && b == 0 && c == 0)
            {
                a_setter();
                b_setter();
                c_setter();
            }
        }
    }
}
```

```

        else if (a == 0)
        {
            Console.WriteLine("A не может быть равно 0");
            a_setter();
            this.b = b;
            this.c = c;
        }
    }
    else
    {
        if (double.TryParse(coefs[0], out double A))
        {
            if (A == 0)
            {
                Console.WriteLine("A не может быть равно 0");
                a_setter();
            }
            else
            {
                this.a = A;
            }
        }
        else
        {
            Console.WriteLine("A введено не верно");
            a_setter();
        }
        if (double.TryParse(coefs[1], out double B))
        {
            this.b = B;
        }
        else
        {
            Console.WriteLine("B введено не верно");
            b_setter();
        }
        if (double.TryParse(coefs[2], out double C))
        {
            this.c = C;
        }
        else
        {
            Console.WriteLine("C введено не верно");
            c_setter();
        }
    }
}

protected void a_setter()
{
    Console.WriteLine("Введите коэффициент A:");
    while (true) {
        if (double.TryParse(Console.ReadLine(), out double A))
        {
            a = A;
            break;
        }
        else
        {
            Console.WriteLine("Не получается считать");
        }
    }
}

protected void b_setter()
{
    Console.WriteLine("Введите коэффициент B:");

```

```

while (true)
{
    if (double.TryParse(Console.ReadLine(), out double B))
    {
        b = B;
        break;
    }
    else
    {
        Console.WriteLine("Не получается считать");
    }
}
}
protected void c_setter()
{
    Console.WriteLine("Введите коэффициент C:");
    while (true)
    {
        if (double.TryParse(Console.ReadLine(), out double C))
        {
            c = C;
            break;
        }
        else
        {
            Console.WriteLine("Не получается считать");
        }
    }
}
public void calculate_roots()
{
    roots.Clear();
    num_roots = 0;
    double d = b * b - 4 * a * c;
    if (d == 0)
    {
        double t = (-1 * b) / (2 * a);
        if (t == 0)
        {
            roots.Add(0);
        }
        else if (t > 0)
        {
            roots.Add(Math.Sqrt(t));
            roots.Add(-1 * Math.Sqrt(t));
        }
    }
    else if (d > 0)
    {
        double t1 = (-1 * b + Math.Sqrt(d)) / (2 * a);
        double t2 = (-1 * b - Math.Sqrt(d)) / (2 * a);
        if (t1 == 0)
        {
            roots.Add(0);
        }
        else if (t1 > 0)
        {
            roots.Add(Math.Sqrt(t1));
            roots.Add(-1 * Math.Sqrt(t1));
        }
        if (t2 == 0)
        {
            roots.Add(0);
        }
        else if (t2 > 0)
    }
}

```

```

        {
            roots.Add(Math.Sqrt(t2));
            roots.Add(-1 * Math.Sqrt(t2));
        }
    }
    num_roots = roots.Count;
}
public void roots_print()
{
    if (num_roots == 0)
    {
        Console.ForegroundColor = ConsoleColor.Red;
        Console.WriteLine("Нет корней");
        Console.ResetColor();
    }
    else
    {
        Console.ForegroundColor = ConsoleColor.Green;
        Console.WriteLine("Корни уравнения: ");
        foreach (var root in roots)
        {
            Console.Write($"{root.ToString()} ");
        }
        Console.WriteLine();
        Console.ResetColor();
    }
}
}

class Program
{
    static void Main(string[] args)
    {
        BiquadraticEquation Equation = new BiquadraticEquation(args);
        Equation.calculate_roots();
        Equation.roots_print();
    }
}

```

Экранные формы с примерами выполнения программы

```

Введите коэффициент А:
2
Введите коэффициент В:
-54
Введите коэффициент С:
7
Корни уравнения:
5,183603182837295 -5,183603182837295 0,3609127912378025 -0,3609127912378025

```

```

Командная строка
Microsoft Windows [Version 10.0.22621.2861]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Roma>d:

D:\>cd D:\учеба\парадигмы программирования\лабы\ла65\lab05\ConsoleApp1\bin\Debug\net8.0

D:\учеба\парадигмы программирования\лабы\ла65\lab05\ConsoleApp1\bin\Debug\net8.0>ConsoleApp1.exe 1 1 1
Нет корней

D:\учеба\парадигмы программирования\лабы\ла65\lab05\ConsoleApp1\bin\Debug\net8.0>

```