**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе № 1

«Основные конструкции языка C#»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-34Б |  | преподаватель каф. ИУ5 |
| Бромберг Е.А. |  | Нардид А.Н. |
|  |  |  |

Москва, 2024 г.

**Постановка задачи**

**Задание:**

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 А, В, С, вычисляет дискриминант и корни уравнения (в зависимости от дискриминанта).

3. Если коэффициент А, В, С 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 А, В, С 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**

Program.cs

using System;

using System.Collections.Generic;

namespace SquareRoot

{

    /// <summary>

    /// Простое вычисление корней

    /// </summary>

    class SquareRoot\_Simple

    {

        /// <summary>

        /// Вычисление корней

        /// </summary>

        public List<double> CalculateRoots(double a, double b, double c)

        {

            List<double> roots = new List<double>();

            double D = b \* b - 4 \* a \* c;

            if (D == 0.0)

            {

                if ((-b / (2 \* a)) > 0.0) {

                    double root1 = -Math.Sqrt(-b / (2 \* a));

                    double root2 = Math.Sqrt(-b / (2 \* a));

                    roots.Add(root1);

                    roots.Add(root2);

                }

                if ((-b / (2 \* a)) == 0.0) {

                    double root = 0.0;

                    roots.Add(root);

                }

            }

            else if (D > 0.0)

            {

                double sqrtD = Math.Sqrt(D);

                if (((-b - sqrtD) / (2 \* a)) > 0.0) {

                    double root1 = -Math.Sqrt((-b - sqrtD) / (2 \* a));

                    double root2 = Math.Sqrt((-b - sqrtD) / (2 \* a));

                    roots.Add(root1);

                    roots.Add(root2);

                }

                if (((-b - sqrtD) / (2 \* a)) == 0.0) {

                    double root = 0.0;

                    roots.Add(root);

                }

                if (((-b + sqrtD) / (2 \* a)) > 0.0) {

                    double root1 = -Math.Sqrt((-b + sqrtD) / (2 \* a));

                    double root2 = Math.Sqrt((-b + sqrtD) / (2 \* a));

                    roots.Add(root1);

                    roots.Add(root2);

                }

                if (((-b + sqrtD) / (2 \* a)) == 0.0) {

                    double root = 0.0;

                    roots.Add(root);

                }

            }

            return roots;

        }

        /// <summary>

        /// Вывод корней

        /// </summary>

        public void PrintRoots(double a, double b, double c)

        {

            List<double> roots = this.CalculateRoots(a, b, c);

            Console.Write("Коэффициенты: a={0}, b={1}, c={2}. ", a, b, c);

            if(roots.Count == 0)

            {

                Console.ForegroundColor = ConsoleColor.Red;

                Console.WriteLine("Корней нет.");

                Console.ResetColor();

            }

            else if (roots.Count == 1)

            {

                Console.ForegroundColor = ConsoleColor.Green;

                Console.WriteLine("Один корень {0}", roots[0]);

                Console.ResetColor();

            }

            else if (roots.Count == 2)

            {

                Console.ForegroundColor = ConsoleColor.Green;

                Console.WriteLine("Два корня {0} и {1}", roots[0], roots[1]);

                Console.ResetColor();

            }

            else if (roots.Count == 3)

            {

                Console.ForegroundColor = ConsoleColor.Green;

                Console.WriteLine("Три корня {0}, {1} и {2}", roots[0], roots[1], roots[2]);

                Console.ResetColor();

            }

            else if (roots.Count == 4)

            {

                Console.ForegroundColor = ConsoleColor.Green;

                Console.WriteLine("Четыре корня {0}, {1}, {2} и {3}", roots[0], roots[1], roots[2], roots[3]);

                Console.ResetColor();

            }

        }

    }

    class Program

    {

        static void Main(string[] args)

        {

            double a = 0, b = 0, c = 0;

            bool flag = true;

            if(args.Length > 0)

            {

                bool test1, test2, test3;

                test1 = double.TryParse(args[0], out a);

                test2 = double.TryParse(args[1], out b);

                test3 = double.TryParse(args[2], out c);

                if (!test1 || !test2 || !test3) {

                    Console.WriteLine("Enter numeric arguments!");

                    flag = false;

                }

                else {

                    if (a == 0)

                    {

                        Console.WriteLine("A must not be equal to 0!");

                        flag = false;

                    }

                }

            }

            if (!flag || args.Length == 0) {

                while (true) {

                    Console.WriteLine("Enter A");

                    try {

                        a = Convert.ToDouble(Console.ReadLine());

                        if (a == 0)

                        {

                            Console.WriteLine("A must not be equal to 0!");

                            continue;

                        }

                    }

                    catch {

                        Console.WriteLine("Enter numeric arguments!");

                        continue;

                    }

                    Console.WriteLine("Enter B");

                    try {

                        b = Convert.ToDouble(Console.ReadLine());

                    }

                    catch {

                        Console.WriteLine("Enter numeric arguments!");

                        continue;

                    }

                    Console.WriteLine("Enter C");

                    try {

                        c = Convert.ToDouble(Console.ReadLine());

                    }

                    catch {

                        Console.WriteLine("Enter numeric arguments!");

                        continue;

                    }

                    break;

                }

            }

            SquareRoot\_Simple r = new SquareRoot\_Simple();

            r.PrintRoots(a, b, c);

        }

    }

}

**Экранные формы с примерами выполнения программы**



