**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №1

«Трек курса «Задания на основе языка C#»»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-34Б |  | преподаватель каф. ИУ5 |
| Киреев А.А. |  | Ю.Е. Гапанюк. |
| Подпись и дата:  25.09.2020 |  | Подпись и дата:  25.09.2020 |

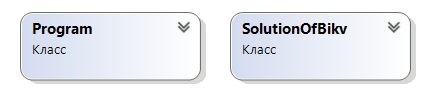
Москва, 2020 г.

**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 А, В, С, вычисляет дискриминант и корни уравнения (в зависимости от дискриминанта).
3. Если коэффициент А, В, С введен некорректно, то необходимо проигнорировать некорректное значение и ввести коэффициент повторно.
4. Первой строкой программа выводит ФИО разработчика и номер группы.
5. Корни уравнения выводятся зеленым цветом. Если корней нет, то сообщение выводится красным цветом.
6. ДОПОЛНИТЕЛЬНОЕ ТРЕБОВАНИЕ. 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.

**Диаграмма классов**



**Текст программы**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApplication2

{

public class SolutionOfBikv

{

double a;

double b;

double c;

public SolutionOfBikv(double A, double B, double C)

{

a = A;

b = B;

c = C;

}

static void Roots(double x1, double x2, double x3, double x4)

{

Console.WriteLine("\nКорни уравнения: ");

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("{0}; {1}; {2}; {3};", x1, x2, x3, x4);

Console.ForegroundColor = ConsoleColor.White;

}

static void Roots(double x1, double x2, double x3)

{

Console.WriteLine("\nКорни уравнения: ");

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("{0}; {1}; {2};", x1, x2, x3);

Console.ForegroundColor = ConsoleColor.White;

}

static void Roots(double x1, double x2)

{

Console.WriteLine("\nКорни уравнения: ");

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("{0}; {1};", x1, x2);

Console.ForegroundColor = ConsoleColor.White;

}

static void Roots(double x1)

{

Console.WriteLine("\nКорни уравнения: ");

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("{0};", x1);

Console.ForegroundColor = ConsoleColor.White;

}

static void No\_answer()

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("\nКорней нет!");

Console.ForegroundColor = ConsoleColor.White;

}

public void Solution()

{

double x1 = 0, x2 = 0, x3 = 0, x4 = 0, discr, x\_sqr1, x\_sqr2, x\_4sqr;

int flag = 0;

//------------------------------

if (a != 0 && b != 0 && c != 0)

{

discr = b \* b - 4 \* a \* c;

if (discr > 0)

{

x\_sqr1 = (-b + Math.Sqrt(discr)) / (2 \* a);

x\_sqr2 = (-b - Math.Sqrt(discr)) / (2 \* a);

if (x\_sqr1 >= 0)

{

x1 = Math.Sqrt(x\_sqr1);

x2 = -Math.Sqrt(x\_sqr1);

flag++;

}

if (x\_sqr2 >= 0)

{

x3 = Math.Sqrt(x\_sqr2);

x4 = -Math.Sqrt(x\_sqr2);

flag++;

}

if (flag == 2)

Roots(x1, x2, x3, x4);

if ((flag == 1) && (x1 != 0) && (x2 != 0))

Roots(x1, x2);

if ((flag == 1) && (x3 != 0) && (x4 != 0))

Roots(x3, x4);

if (flag == 0)

No\_answer();

}

if (discr == 0)

{

x\_sqr1 = -b / (2 \* a);

if (x\_sqr1 >= 0)

{

x1 = Math.Sqrt(x\_sqr1);

x2 = -Math.Sqrt(x\_sqr1);

flag++;

}

if (flag == 1)

Roots(x1, x2);

else No\_answer();

}

if (discr < 0)

{

No\_answer();

}

}

//------------------------------

if (a != 0 && b != 0 && c == 0)

{

flag++;

x1 = 0;

x2 = 0;

x\_sqr1 = -b / a;

if (x\_sqr1 >= 0)

{

x3 = Math.Sqrt(x\_sqr1);

x4 = -Math.Sqrt(x\_sqr1);

flag++;

}

if (flag == 2)

Roots(x1, x2, x3, x4);

else if (flag == 1)

Roots(x1, x2);

}

//-------------------------------

if (a != 0 && b == 0 && c != 0)

{

x\_4sqr = -c / a;

if (x\_4sqr > 0)

{

x\_sqr1 = Math.Sqrt(x\_4sqr);

x\_sqr2 = -Math.Sqrt(x\_4sqr);

if (x\_sqr1 >= 0)

{

flag++;

x1 = Math.Sqrt(x\_sqr1);

x2 = -Math.Sqrt(x\_sqr1);

}

if (x\_sqr2 >= 0)

{

flag++;

x3 = Math.Sqrt(x\_sqr2);

x4 = -Math.Sqrt(x\_sqr2);

}

if (flag == 2)

Roots(x1, x2, x3, x4);

if ((flag == 1) && (x1 != 0) && (x2 != 0))

Roots(x1, x2);

if ((flag == 1) && (x3 != 0) && (x4 != 0))

Roots(x3, x4);

}

if (x\_4sqr == 0)

{

x1 = 0;

x2 = 0;

x3 = 0;

x4 = 0;

Roots(x1, x2, x3, x4);

}

}

//-------------------------------

if (a != 0 && b == 0 && c == 0)

{

x1 = 0;

x2 = 0;

x3 = 0;

x4 = 0;

Roots(x1, x2, x3, x4);

}

//-------------------------------

if (a == 0 && b != 0 && c != 0)

{

x\_sqr1 = -c / b;

if (x\_sqr1 >= 0)

{

x1 = Math.Sqrt(x\_sqr1);

x2 = -Math.Sqrt(x\_sqr1);

flag++;

}

if (flag == 1)

Roots(x1, x2);

else No\_answer();

}

//-------------------------------

if (a == 0 && b != 0 && c == 0)

{

x1 = 0;

x2 = 0;

Roots(x1, x2);

}

//-------------------------------

}

}

class Program

{

static void Main(string[] args)

{

int cont = 0;

double A, B, C;

Console.WriteLine("Киреев Андрей ИУ5-34Б");

Console.WriteLine("Решение биквадратного уравнения вида: ax4 + bx2 + c = 0");

do

{

A:

Console.WriteLine("Введите коэффициенты уравнения: ");

Console.Write("a = ");

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

Console.Write("b = ");

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

Console.Write("c = ");

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

if (A == 0 && B == 0 && C != 0)

{

Console.WriteLine("Введены некорректные коэффициенты ");

goto A;

}

if (A == 0 && B == 0 && C == 0)

{

Console.WriteLine("Введены некорректные коэффициенты ");

goto A;

}

SolutionOfBikv MyEquation = new SolutionOfBikv(A, B, C);

MyEquation.Solution();

Console.WriteLine("Продолжить? (1-да, 0-нет)");

cont = Convert.ToInt32(Console.ReadLine());

} while (cont == 1);

Console.ReadKey();

}

}

}

**Экранные формы с примерами выполнения программы**

