**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Радиотехнический»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет технологий»

Отчет по лабораторной работе №1

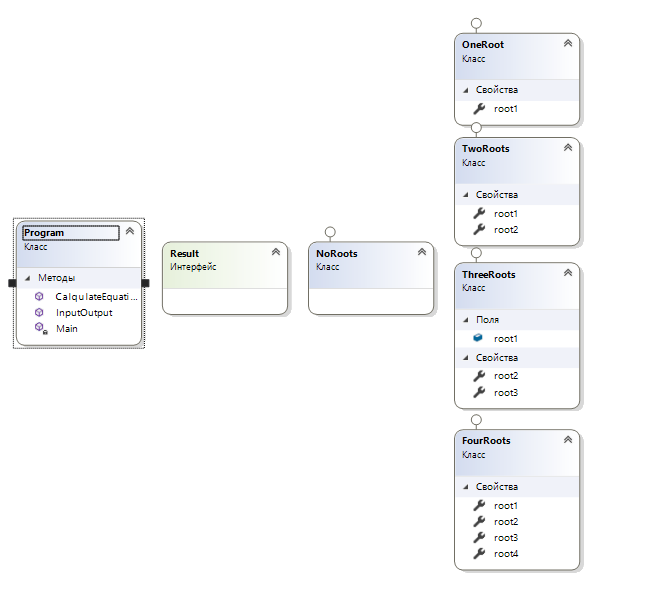
|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы РТ5-31Б |  | преподаватель каф. ИУ5 |
| Грызин Алексей |  | Гапанюк Ю.Е. |
| Подпись и дата: |  | Подпись и дата: |

# Описание задания

Разработать программу для решения биквадратного уравнения – <https://ru.wikipedia.org/wiki/%D0%A3%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5_%D1%87%D0%B5%D1%82%D0%B2%D1%91%D1%80%D1%82%D0%BE%D0%B9_%D1%81%D1%82%D0%B5%D0%BF%D0%B5%D0%BD%D0%B8#%D0%91%D0%B8%D0%BA%D0%B2%D0%B0%D0%B4%D1%80%D0%B0%D1%82%D0%BD%D0%BE%D0%B5_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5>

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

class Program

{

//Функция для решения биквадратного уравнения

public Result CalqulateEquation(double A, double B, double C)

{

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

if (C == 0)

{

roots.Add(0);

if (B < 0)

{

roots.Add(Math.Sqrt((-B) / A));

roots.Add(-Math.Sqrt((-B) / A));

}

}

else {

double D = B \* B - 4 \* A \* C;

if (D == 0)

{

double root;

root = -B / (2 \* A);

if (root > 0)

{

roots.Add(Math.Sqrt(root));

roots.Add(-Math.Sqrt(root));

}

}

else if (D > 0)

{

double root1, root2;

root1 = (-B + Math.Sqrt(D)) / (2 \* A);

root2 = (-B - Math.Sqrt(D)) / (2 \* A);

if (root1 > 0)

{

roots.Add(Math.Sqrt(root1));

roots.Add(-Math.Sqrt(root1));

}

if (root2 > 0)

{

roots.Add(Math.Sqrt(root2));

roots.Add(-Math.Sqrt(root2));

}

}

}

roots.Sort();

if (roots.Count == 0)

{

return new NoRoots();

}

else if (roots.Count == 1)

{

return new OneRoot() {

root1 = roots[0]

};

}

else if (roots.Count == 2)

{

return new TwoRoots()

{

root1 = roots[0],

root2 = roots[1]

};

}

else if (roots.Count == 3)

{

return new ThreeRoots()

{

root1 = roots[0],

root2 = roots[1],

root3 = roots[2]

};

}

else

{

return new FourRoots()

{

root1 = roots[0],

root2 = roots[1],

root3 = roots[2],

root4 = roots[3]

};

}

}

public void InputOutput()

{

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

List<char> abc = new List<char>();

abc.Add('A');

abc.Add('B');

abc.Add('C');

//Ввод переменных с клавиатуры

//Проверка правильности ввода

for (int i = 0; i < 3; i++)

{

bool flag = false;

double coef = 0;

while (flag == false)

{

string buf = "";

Console.WriteLine("Введите коэффициент {0}", abc[i]);

buf = Console.ReadLine();

if (buf != "")

{

int c = 0;

for (int j = 0; j < buf.Length; j++)

{

if (Char.IsDigit(buf[j]) == false && buf[j] != '.' && buf[j] != '-' && buf[j] != ' ') c++;

}

if (c == 0)

{

coef = Convert.ToDouble(buf);

if (i != 0) flag = true;

else

{

if (coef != 0) flag = true;

}

}

}

}

coefficients.Add(coef);

}

Result result = CalqulateEquation(coefficients[0], coefficients[1], coefficients[2]);

//Вывод результата на консоль

Console.Write("Коэффициенты биквадратного уравнения: A={0}, B={1}, C={2}\n", coefficients[0], coefficients[1], coefficients[2]);

string resultType = result.GetType().Name;

if (resultType == "NoRoots")

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Корней нет!");

Console.ResetColor();

}

else if (resultType == "OneRoot")

{

OneRoot rt1 = (OneRoot)result;

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("Один корень: x={0}", rt1.root1);

Console.ResetColor();

}

else if (resultType == "TwoRoots")

{

TwoRoots rt2 = (TwoRoots)result;

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("Два корня: x1={0}, x2={1}", rt2.root1, rt2.root2);

Console.ResetColor();

}

else if (resultType == "ThreeRoots")

{

ThreeRoots rt3 = (ThreeRoots)result;

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("Три корня: x1={0}, x2={1}, x3={2}", rt3.root1, rt3.root2, rt3.root3);

Console.ResetColor();

}

else if (resultType == "FourRoots")

{

FourRoots rt4 = (FourRoots)result;

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("Четыре корня: x1={0}, x2={1}, x3={2}, x4={3}", rt4.root1, rt4.root2, rt4.root3, rt4.root4);

Console.ResetColor();

}

}

static void Main(string[] args)

{

Console.Write("Выполнил: Грызин А.Н. РТ5-31\n");

Program equation = new Program();

equation.InputOutput();

}

}

}

interface Result { }

class NoRoots : Result { }

class OneRoot : Result

{

public double root1 { get; set; }

}

class TwoRoots : Result

{

public double root1 { get; set; }

public double root2 { get; set; }

}

class ThreeRoots : Result

{

public double root1 = 0;

public double root2 { get; set; }

public double root3 { get; set; }

}

class FourRoots : Result

{

public double root1 { get; set; }

public double root2 { get; set; }

public double root3 { get; set; }

public double root4 { get; set; }

}

# Результат работы программы

