**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Дисциплина «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №1

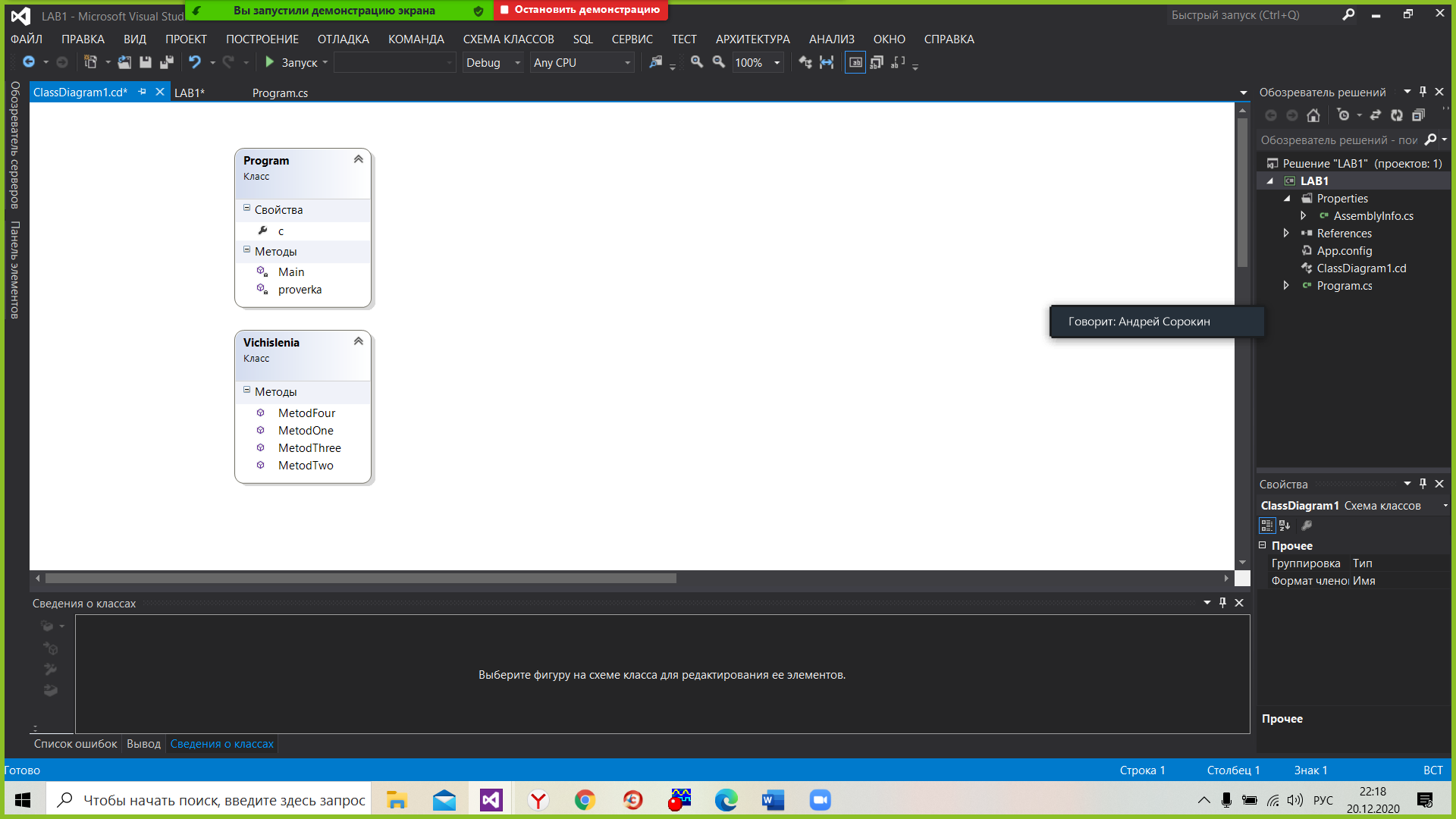
|  |  |  |
| --- | --- | --- |
| Выполнила: |  | Проверил: |
| студентка группы  ИУ5-34Б |  | преподаватель каф. ИУ5 |
| Тахтамышева Анна |  | Гапанюк Ю.Е. |

**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 LAB1

{

class Program

{

//public double[] abc;

static void Main(string[] args)

{

Console.ForegroundColor = ConsoleColor.DarkCyan;

Console.WriteLine("Выполнила Тахтамышева Анна");

Console.WriteLine("Группа ИУ5-34Б");

Console.ResetColor();

while (proverka(args) == false)

{

Console.WriteLine("Вы ввели неправильные данные! Повторите ввод.");

Console.WriteLine("Введите a, b, c.");

//args[0]=Console.ReadLine();

//args[1] = Console.ReadLine();

//args[2] = Console.ReadLine();

for (int j = 0; j <= 2; j = j + 1)

{

args[j] = Console.ReadLine();

}

}

//Console.WriteLine("Вы ввели правильные данные!");

double[] abc = { 0, 0, 0 };

for (int k = 0; k <= 2; k = k + 1)

{

abc[k] = Convert.ToDouble(args[k]);

//Console.WriteLine("{0}", abc[k]);

}

Vichislenia Pervi = new Vichislenia();

int f = 0;

if ((abc[0] == f) & (abc[1] == f) & (abc[2] == f))

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Уравнения не сущеcтвует!");

Console.ResetColor();

}

else

{

if (abc[0] == f)

{

Pervi.MetodOne(abc[1], abc[2]);

}

if (abc[1] == f)

{

Pervi.MetodTwo(abc[0], abc[2]);

}

if (abc[2] == f)

{

Pervi.MetodThree(abc[0], abc[1]);

}

if ((abc[0] != f) & (abc[1] != f) & (abc[2] != f))

{

Pervi.MetodFour(abc[0], abc[1], abc[2]);

}

//if ((abc[0] == f) & (abc[1] == f) & (abc[2] == f))

// {

// Console.ForegroundColor = ConsoleColor.Red;

// Console.WriteLine("Уравнения не сущетсвует!");

// Console.ResetColor();

// }

}

}

static bool proverka(string[] arg)

{

bool rez = true;

int i = 0;

double[] abc = { 0, 0, 0 };

while (i <= 2)

{

rez = double.TryParse(arg[i], out abc[i]);

if (rez == false)

{

return false;

}

i = i + 1;

}

return rez;

}

public static object c { get; set; }

}

class Vichislenia

{

//Vichislenia();

//static Vichislenia();

public void MetodOne(double b, double c)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Это не биквадратное уравнение!");

Console.ResetColor();

}

public void MetodTwo(double a, double c)

{

double x1;

double x2;

double h = 0;

if (((a > 0) & (c > 0)) | ((a < 0) & (c < 0)))

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Действительных корней не существует.");

Console.ResetColor();

}

else

{

h = (-1) \* (c / a);

//Console.WriteLine("h={0}", h);

x1 = Math.Pow(h, 0.25);

x2 = (-1) \* Math.Pow(h, 0.25);

Console.ForegroundColor = ConsoleColor.Green;

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

Console.ResetColor();

}

}

public void MetodThree(double a, double b)

{

double x1;

double x2;

double x3;

double r = 0;

if (((a > 0) & (b < 0)) | ((a < 0) & (b > 0)))

{

r = (-1)\*(b / a);

//Console.WriteLine("r={0}", r);

x1 = 0;

x2 = Math.Pow(r, 0.5);

x3 = (-1)\*Math.Pow(r, 0.5);

Console.ForegroundColor = ConsoleColor.Green;

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

Console.ResetColor();

}

else

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Действительных корней не существует.");

Console.ResetColor();

}

}

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

{

double D;

double x1;

double x2;

double x3;

double x4;

double t1;

double t2;

D = (b\*b) - (4 \* a \* c);

if (D > 0)

{

t1 = (b \* (-1) - (Math.Pow(D, 0.5))) / (2 \* a);

//Console.WriteLine("t1={0}", t1);

t2 = (b \* (-1) + (Math.Pow(D, 0.5))) / (2 \* a);

//Console.WriteLine("t2={0}", t2);

//if ((t1 > 0) | (t2 > 0))

//{

if (t1 > 0)

{

x1 = Math.Pow(t1, 0.5);

x2 = (-1) \* Math.Pow(t1, 0.5);

Console.ForegroundColor = ConsoleColor.Green;

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

Console.ResetColor();

}

if (t2 > 0)

{

x3 = Math.Pow(t2, 0.5);

x4 = (-1) \* Math.Pow(t2, 0.5);

//Console.ForegroundColor = ConsoleColor.Green;

//Console.WriteLine("x3={0} x4={1}", x3, x4);

//Console.ResetColor();

if (t1 < 0)

{

Console.ForegroundColor = ConsoleColor.Green;

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

Console.ResetColor();

}

else

{

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("x3={0} x4={1}", x3, x4);

Console.ResetColor();

}

}

//}

else

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Действительных корней не существует.");

Console.ResetColor();

}

}

if (D == 0)

{

t1 = (b \* (-1)) / (2 \* a);

if (t1 > 0)

{

x1 = Math.Pow(t1, 0.5);

x2 = (-1) \* Math.Pow(t1, 0.5);

Console.ForegroundColor = ConsoleColor.Green;

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

Console.ResetColor();

}

else

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Действительных корней не существует.");

Console.ResetColor();

}

}

if (D < 0)

{

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("Действительных корней не существует.");

Console.ResetColor();

}

//Console.ReadLine();

//Console.ReadKey();

}

}

}

**Пример экранных форм**

