# Лабораторная работа №5

Тема: Разработка классов, создание объектов и использование их в программах.

Цель: Научиться разрабатывать классы, создавать объекты и использовать их в программах.

**Технологическое оснащение:** Ноутбук DELL, Windows 10, MS Word 2016, MS Visual Studio 2019, методическое указание к работе.

**Выполнение работы:**

**Пример 1:** Создать класс, моделирующий работу с треугольником (задаются три стороны a, b и c и вычисляется площадь треугольника s).

Результат выполнения программы (рисунок 1):

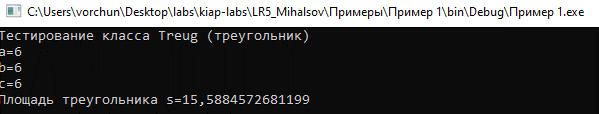


Рисунок 1 – Результат работы примера 1.

**Пример 2**: Для класса, моделирующего работу с окружностью, создадим свойство для изменения радиуса окружности.

Результат выполнения программы (рисунок 2):

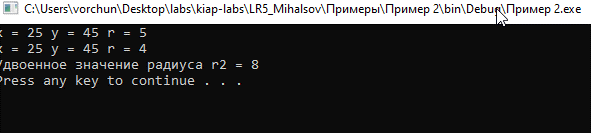


Рисунок 2 – Результат работы программы 2

**Пример 3:** Описание класса и создание объектов.

Результат выполнения программы (рисунок 3):

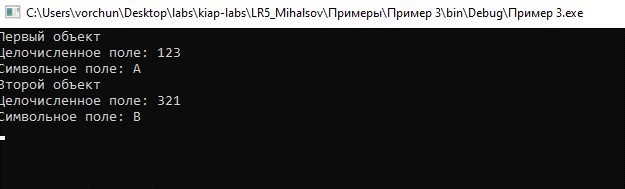


Рисунок 3 – Выполнение работы примера 3.

**Пример 4:** Присваивание объектов.

Результат выполнения программы (рисунок 4):

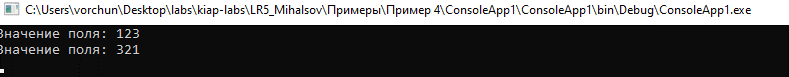


Рисунок 4 – Результата работы примера 4.

**Пример 5:** Закрытые члены класса и перегрузка методов.

Результат выполнения программы (рисунок 5):

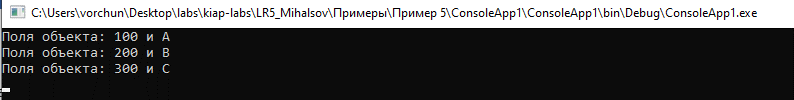


Рисунок 5 – Результат работы примера 5.

**Пример 6:** Использование конструктора.

Результат выполнения программы (рисунок 6):

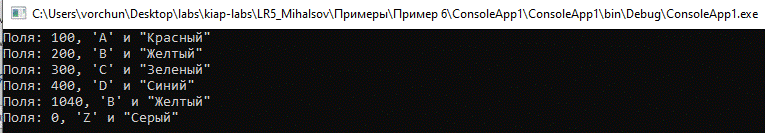


Рисунок 6 – Результат работы примера 6.

**Задание 1:** Составить программу для вычисления скорости в конце пути и путь, пройденный телом за время t с ускорением a и начальной скорости v0. Вычисление значений переменных оформить в виде класса, в который передать исходные данные в виде входных параметров. Исходные данные для отладки программы подобрать самостоятельно.

Листинг программы:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Задание\_1

{

class SpeedWay

{

float \_t;

float \_a;

float \_v0;

public double EndSpeed

{

get

{

return \_v0 + (\_a \* \_t);

}

}

public double Distance

{

get

{

return Math.Pow(\_v0, 2) + (\_a \* Math.Pow(\_t, 2) / 2);

}

}

public SpeedWay(float t, float a, float v0)

{

\_t = t;

\_a = a;

\_v0 = v0;

}

}

class Program

{

static void Main(string[] args)

{

Console.Write("Введите время пути (t): "); float t = float.Parse(Console.ReadLine());

Console.Write("Введите ускорение (a): "); float a = float.Parse(Console.ReadLine());

Console.Write("Введите начальную скорость (v0): "); float v0 = float.Parse(Console.ReadLine());

SpeedWay A = new SpeedWay(t,a,v0);

Console.WriteLine("Конечная скорость (v): {0}", A.EndSpeed);

Console.WriteLine("Пройденное растояние (S): {0}", A.Distance);

Console.ReadKey(true);

}

}

}

Результат работы программы (рисунок 7):

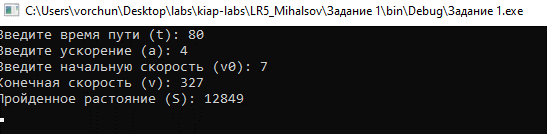


Рисунок 7 – Результат работы программы 1

**Задание 2:** Написать методы для вычисления приведенных функций с использование классов, выбор функции предоставить пользователю с помощью консоли.

Листинг программы

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Задание\_2

{

class functionChose

{

int \_chose;

float \_x;

double f1(float x)

{

return Math.Pow(Math.Sin(x), 4);

}

double f2(float x)

{

return Math.Pow(x, 3 - 0.2 \* x);

}

double f3(float x)

{

return Math.Pow(x, 2) - 6 \* x + 18;

}

public functionChose(int chose, float x)

{

\_chose = chose;

\_x = x;

}

public double Y

{

get

{

switch (\_chose)

{

case 1:

return f1(\_x);

break;

case 2:

return f2(\_x);

break;

case 3:

return f3(\_x);

break;

default:

return -1.0;

}

}

}

}

class Program

{

static void Main(string[] args)

{

// y - переменная для сохранеия подсчитанного значения

double y = 0;

// n - переменная для сохранеия выбранной пользователем функции

int n;

// Ввходим x

Console.Write("Введите x : "); float x = float.Parse(Console.ReadLine());

// Пока пользователь не введет значения в нужном диапозоне (0-3), начинать цикл заново

do

{

Console.Write("Введите номер функции для расчета (1 <= n <= 3) : ");

n = int.Parse(Console.ReadLine());

} while (!(n >= 1 && n <= 3));

functionChose F = new functionChose(n,x);

Console.WriteLine("y : {0}", F.Y);

Console.ReadKey(true);

}

}

}

Проверка работы программы.

Проверка программы по первому условию (рисунок 8):

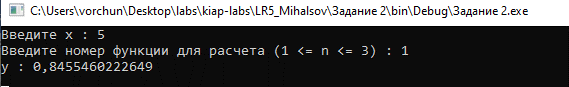


Рисунок 8 – Проверка по первому условию

Проверка программы по второму условию (рисунок 9):

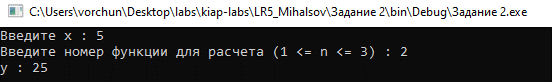


Рисунок 9 – Проверка по второму условию

Проверка программы по третьему условию (рисунок 10):

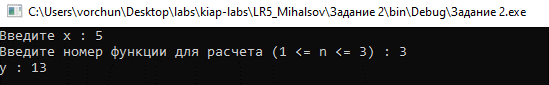


Рисунок 10 – Проверка по третьему условию

**Задание 3:** Написать методы для вычисления приведенных функций с использование классов, выбор функции предоставить пользователю с помощью формы с компонентами.

Форма программы (рисунок 11).

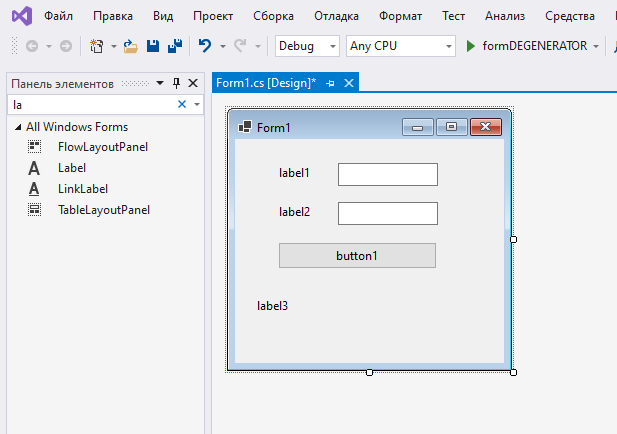


Рисунок 11 – Форма программы

На этой форме присутствуют следующие компоненты: label1 – Служащий для приглашения для ввода значения X в textBox1; label2 – Служащий для приглашения для ввода значения N в textBox2; textBox1 – Служащий для ввода значения X; textBox2 – Служащий для ввода значения N; label3 – служащий для вывода значения Y; button1 – служащий для вызова расчёта Y.

Преобразованная форма (рисунок 12):

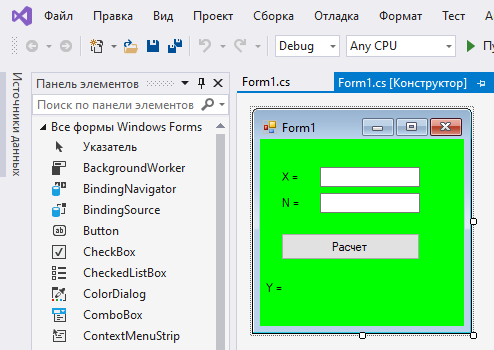


Рисунок 12 – Преобразованная форма программы

Листинг программы

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Задание\_3

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

class functionChose

{

int \_chose;

float \_x;

double f1(float x)

{

return Math.Pow(Math.Sin(x), 4);

}

double f2(float x)

{

return Math.Pow(x, 3 - 0.2 \* x);

}

double f3(float x)

{

return Math.Pow(x, 2) - 6 \* x + 18;

}

public functionChose(int chose, float x)

{

\_chose = chose;

\_x = x;

}

public double Y

{

get

{

switch (\_chose)

{

case 1:

return f1(\_x);

break;

case 2:

return f2(\_x);

break;

case 3:

return f3(\_x);

break;

default:

return -1.0;

}

}

}

}

private void button1\_Click(object sender, EventArgs e)

{

float x;

double y = 0;

// n - переменная для сохранеия выбранной пользователем функции

int n;

x = float.Parse(textBox1.Text);

n = int.Parse(textBox2.Text);

if(!(n >= 1 && n <= 3))

{

label4.Text = "Введиете номер функции от 1 до 3 !!!";

return;

}

functionChose F = new functionChose(n, x);

label4.Text = "Y = " + F.Y.ToString();

}

}

}

Проверка программы.

Проверка по условию №1 (рисунок 13).

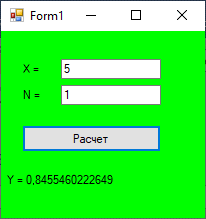


Рисунок 13 – Проверка по первому условию

Проверка по условию №2 (рисунок 14).

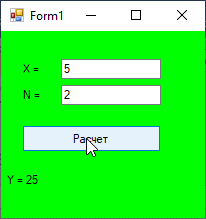


Рисунок 14 – Проверка по второму условию

Проверка по условию №3 (рисунок 15).

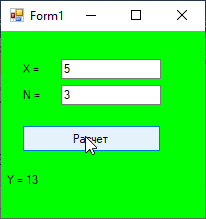


Рисунок 15 – Проверка по третьему условию

**Вывод:** В результате выполнения лабораторной работы были получены практические навыки по преобразованию программного кода с использованиям классов.