7МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

#### Факультет информационных технологий и компьютерной безопасности

(факультет)

### Кафедра Систем автоматизированного проектирования и информационных систем

# ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 8

по дисциплине Программирование на платформе .NET Framework

Тема «Разработка пользовательских интерфейсов приложения для работы с файловой системой на платформе .NET Framework» ㅤ

Выполнили студенты группы ИСТ-222 Е.Г. Гладнева

Подпись, дата Инициалы, фамилия

А.Н. Грипунова

Подпись, дата Инициалы, фамилия

Е.В. Журавлёва

Подпись, дата Инициалы, фамилия

## Руководитель Э.И. Воробьев

Подпись, дата Инициалы, фамилия

Защищена \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

дата

2023

Лабораторная работа № 8

«Разработка пользовательских интерфейсов приложения для работы с файловой системой на платформе .NET Framework»

1 Цель работы

Целью лабораторной работы является изучение основ построения интерфейса для работы с файловой системой на платформе .NET Framework.

2 Задание на лабораторную работу

Был выбран вариант №9.

Выполнить задания в консольном приложении:

Задание 1. Написать программу, которая создает файл, содержащий действительные числа, и находит сумму нечетных чисел, содержащихся в файле.

Задание 2. Разработать интерфейс приложения и реализовать функции создания папки.

Задание 3. Разработать интерфейс приложения и реализовать функции создания текстового файла различными методами.

Задание 4. Разработать интерфейс приложения и реализовать функции чтения из файла.

3 Ход выполнения

1. В первую очередь создадим интерактивное меню (рисунок 1), с помощью которого можно будет перейти к любому заданию. Для этого добавим элемент ListBox с вариантами заданий. Флажок позволяет выбрать каким образом открывать задание: в консоли или в приложении.

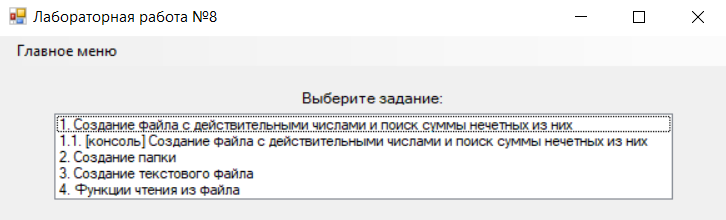


Рисунок 1 – Меню программы

2. Затем создаём внутри приложения Windows Forms консольное приложение ConsoleApp. Вызов конкретного задания происходит с помощью метода Process.Start, которому передаётся путь на исполняемый файл консольного приложения и аргумент запуска, который является номером задания.

3. Для выполнения первого задания создаём новый файл и инициализируем потоковый писатель и массив чисел с плавающей точкой. Это массив в цикле заполняем случайными дробными действительными числами от -10 до 10, параллельно записывая их в файл. Закрывай этот файл и вновь открываем, но уже с потоковым читателем. Считываем первую строку, разделяем числа по пробелам, проверяем каждое число на чётность и, если оно нечётное, то прибавляем к сумме. Пример выполнения данного задания в консоли представлен на рисунке 2, в приложении – на рисунке 3.

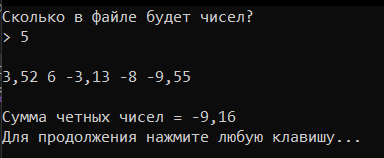


Рисунок 2 – Задание 1 в консоли

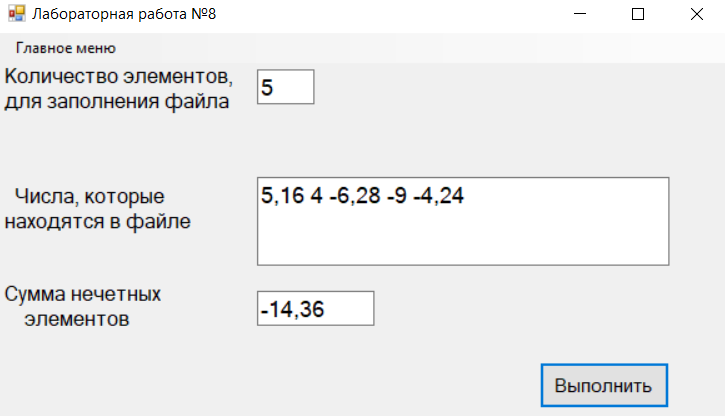


Рисунок 3 – Задание 1 в приложении

4. При выполнении второго задания запрашиваем у пользователя имя папки, которую он хочет создать, и просим выбрать её расположение, используя FolderBrowserDialog. Как только пользователь нажмёт на кнопку создания папки, проверяем, чтобы имя папки было введено и не имело недопустимых символов, путь до расположения был выбран и папки с таким названием по такому расположению. Если все условия выполняются, создаём папку. Пример выполнения данного задания в приложении представлен на рисунке 4.

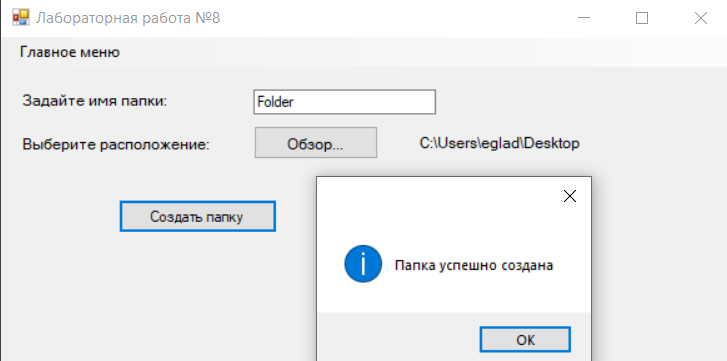


Рисунок 4 – Задание 2 в приложении

5. В третьем задании пользователю предоставляется на выбор несколько способ создания файла, а именно: создать файл или заменить уже существующий (File.Create), создать или открыть для записи (File.CreateText), создать файл и выбросить исключение при его существовании (FileStream("file3.txt", FileMode.CreateNew)), создать или открыть файл для дозаписи (FileStream("file4.txt", FileMode.Append)). Пример выполнения данного задания в приложении представлен на рисунке 5.

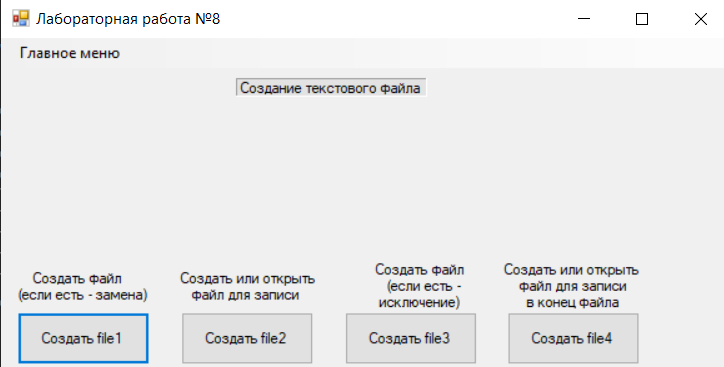


Рисунок 5 – Задание 3 в приложении

6. В четвертом задании пользователь может ознакомиться с операцией чтения из файла. В папку с исполняемым файлом кладём poemfile.txt, содержащий некоторый текст. Затем используем потоковый читатель, чтобы считать из файла первые 4 строчки и вывести их в соответствующие поля. Пример выполнения данного задания в приложении представлен на рисунке 6.

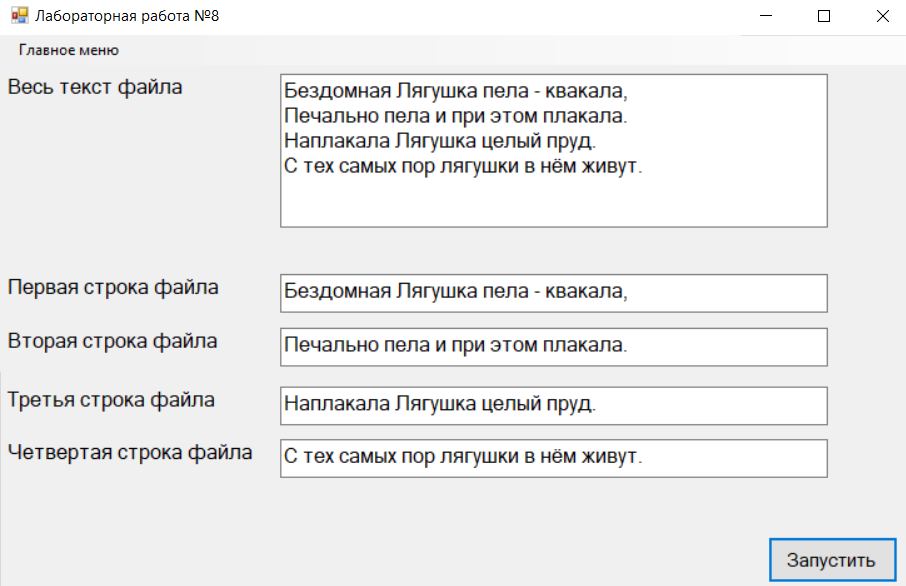


Рисунок 6 – Задание 4 в приложении

Вывод

В результате выполнения лабораторной работы мы изучили основы разработки пользовательского интерфейса для работы с файловой системой на платформе .NET Framework.

Приложение А

Листинг программного кода языка C#

Консольное приложение

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp

{

internal class Program

{

static double exercise1(int numbers)

{

FileStream file = new FileStream("file.txt", FileMode.OpenOrCreate); //создаем файл

StreamWriter writer = new StreamWriter(file); //создаем потоковый писатель

Random random = new Random();

double[] massive = new double[numbers]; //массив дробных чисел

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

{

if (i > numbers)

break;

if (i == numbers - 1)

{

massive[i] = Math.Round(random.NextDouble() + random.Next(-10, 10), 2);

writer.Write(massive[i] + " ");

}

else

{

massive[i] = Math.Round(random.NextDouble() + random.Next(-10, 10), 2);

massive[i + 1] = random.Next(-10, 10);

writer.Write(massive[i] + " ");

writer.Write(massive[i + 1] + " ");

i++;

}

}

writer.WriteLine();

writer.Close();

Console.WriteLine();

FileStream file1 = new FileStream("file.txt", FileMode.Open); //открываем файл

StreamReader reader = new StreamReader(file1);

BinaryReader reader2 = new BinaryReader(file1);

string number = reader.ReadLine();

double[] massive\_read = new double[numbers];

double sum = 0;

foreach (var numberr in number.Split()) //в переменную numberr записываем числа, которые отделены друг от друга пробелом

{

if (numberr == "") //если числа заканчиваются,то цикл прекращается

break;

if (Double.Parse(numberr) % 2 != 0) //проверяем четность

{

sum += Double.Parse(numberr); //находим сумму

}

Console.Write(numberr + " ");

}

Console.WriteLine();

reader.Close();

Console.WriteLine();

return sum;

}

static void Main(string[] args)

{

while (true)

{

try

{

int numbers;

Console.WriteLine("Сколько в файле будет чисел?");

Console.Write("> ");

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

Console.WriteLine("Сумма четных чисел = " + exercise1(numbers));

Console.WriteLine("Для продолжения нажмите любую клавишу...");

Console.ReadKey();

break;

}

catch

{

Console.WriteLine("Вы ввели символ или не ввели ничего, введите число");

}

}

}

}

}

Приложение Б

Листинг программного кода языка C#

Задание 1

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace LW\_08\_App

{

public partial class Ex\_1 : UserControl

{

public Ex\_1()

{

InitializeComponent();

}

void exercise1(int numbers)

{

FileStream file = new FileStream("file.txt", FileMode.OpenOrCreate); //создаем файл

StreamWriter writer = new StreamWriter(file); //создаем потоковый писатель

Random random = new Random();

double[] massive = new double[numbers]; //массив дробных чисел

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

{

if (i > numbers)

break;

if (i == numbers - 1)

{

massive[i] = Math.Round(random.NextDouble() + random.Next(-10, 10), 2);

writer.Write(massive[i] + " ");

}

else

{

massive[i] = Math.Round(random.NextDouble() + random.Next(-10, 10), 2);

massive[i + 1] = random.Next(-10, 10);

writer.Write(massive[i] + " ");

writer.Write(massive[i + 1] + " ");

i++;

}

}

writer.WriteLine();

writer.Close();

FileStream file1 = new FileStream("file.txt", FileMode.Open); //открываем файл

StreamReader reader = new StreamReader(file1);

BinaryReader reader2 = new BinaryReader(file1);

string number = reader.ReadLine();

double[] massive\_read = new double[numbers];

double sum = 0;

foreach (var numberr in number.Split()) //в переменную numberr записываем числа, которые отделены друг от друга пробелом

{

if (numberr == "") //если числа заканчиваются,то цикл прекращается

break;

if (Double.Parse(numberr) % 2 != 0) //проверяем четность

{

sum += Double.Parse(numberr); //находим сумму

}

textBox2.Text += (numberr + " ");

}

textBox3.Text = Convert.ToString(sum);

reader.Close();

}

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

{

try

{

textBox2.Clear();

textBox3.Clear();

int numbers;

numbers = Convert.ToInt32(textBox1.Text);

exercise1(numbers);

}

catch

{

MessageBox.Show("Вы ввели символ или ничего не ввели!", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

}

}

}

Приложение В

Листинг программного кода языка C#

Задание 2

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Diagnostics;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace LW\_08

{

public partial class Ex\_2 : UserControl

{

public Ex\_2()

{

InitializeComponent();

}

private DialogResult result;

private string pathToFolder;

private void btn\_FileDialog\_Click(object sender, EventArgs e)

{

FolderBrowserDialog fbd = new FolderBrowserDialog();

fbd.ShowNewFolderButton = false;

result = fbd.ShowDialog();

if (result == DialogResult.OK)

{

pathToFolder = fbd.SelectedPath;

Label path = new Label();

path.Text = fbd.SelectedPath;

path.Font = new System.Drawing.Font("Microsoft Sans Serif", 9F);

path.Location = new Point(btn\_FileDialog.Location.X + 130, 52);

path.AutoSize = true;

Controls.Add(path);

}

}

private void btn\_createFolder\_Click(object sender, EventArgs e)

{

string bad\_symbols = "+=[]:\*?;«,./\\<>|' ";

if (folderName.Text.Length == 0)

{

MessageBox.Show("Введите имя создаваемой папки", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

foreach (char bad\_symb in bad\_symbols)

{

foreach(char ch in folderName.Text)

{

if (ch == bad\_symb)

{

MessageBox.Show($"Недопустимый символ: {bad\_symb}", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

}

}

if (result != DialogResult.OK)

{

MessageBox.Show("Введите расположение создаваемой папки", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

string name = pathToFolder + "\\" + folderName.Text;

if (Directory.Exists(name))

{

MessageBox.Show("Папка с таким названием в данном расположении уже существует", "Ошибка",

MessageBoxButtons.OK, MessageBoxIcon.Error);

return;

}

Directory.CreateDirectory(name);

MessageBox.Show("Папка успешно создана", "",

MessageBoxButtons.OK, MessageBoxIcon.Information);

OpenFileDialog opd = new OpenFileDialog();

opd.InitialDirectory = name;

opd.ShowDialog();

}

}

}

Приложение Г

Листинг программного кода языка C#

Задание 3

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;

using System.IO;

using System.Reflection.Emit;

namespace LW\_8\_\_3

{

public partial class Ex\_3 : UserControl

{

public Ex\_3()

{

InitializeComponent();

}

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

{

File.Create("file1.txt").Close();

button1.Visible = false;

button2.Visible = false;

button4.Visible = false;

button5.Visible = false;

label2.Visible = true;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

button3.Visible = true;

}

private void button2\_Click(object sender, EventArgs e)

{

File.CreateText("file2.txt").Close();

button1.Visible = false;

button2.Visible = false;

button4.Visible = false;

button5.Visible = false;

label2.Visible = true;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

button3.Visible = true;

}

private void button4\_Click(object sender, EventArgs e)

{

FileStream file3 = new FileStream("file3.txt", FileMode.CreateNew);

file3.Close();

button1.Visible = false;

button2.Visible = false;

button4.Visible = false;

button5.Visible = false;

label2.Visible = true;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

button3.Visible = true;

}

private void button3\_Click(object sender, EventArgs e)

{

button1.Visible = true;

button2.Visible = true;

button4.Visible = true;

button5.Visible = true;

label2.Visible = false;

label3.Visible = true;

label4.Visible = true;

label5.Visible = true;

label6.Visible = true;

button3.Visible = false;

}

private void button5\_Click(object sender, EventArgs e)

{

FileStream file4 = new FileStream("file4.txt", FileMode.Append);

file4.Close();

button1.Visible = false;

button2.Visible = false;

button4.Visible = false;

button5.Visible = false;

label2.Visible = true;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

button3.Visible = true;

}

}

}

Приложение Е

Листинг программного кода языка C#

Задание 4

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace LW\_08

{

public partial class Ex\_4 : UserControl

{

public Ex\_4()

{

InitializeComponent();

}

void reading\_file()

{

FileStream file = new FileStream("poemfile.txt", FileMode.Open);

StreamReader reader = new StreamReader(file);

textBox2.Text = reader.ReadLine();

textBox3.Text = reader.ReadLine();

textBox4.Text = reader.ReadLine();

textBox5.Text = reader.ReadLine();

reader.Close();

FileStream file1 = new FileStream("poemfile.txt", FileMode.Open);

StreamReader reader1 = new StreamReader(file1);

textBox1.Text = reader1.ReadToEnd();

reader1.Close();

}

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

{

reading\_file();

}

}

}