Московский государственный технический университет

им. Н. Э. Баумана

**Отчет**

**по лабораторной работе №4**

Выполнил:

Студент группы ИУ5-35Б

Богданова В.В.

Проверил:

Преподаватель

Гапанюк Ю.Е.

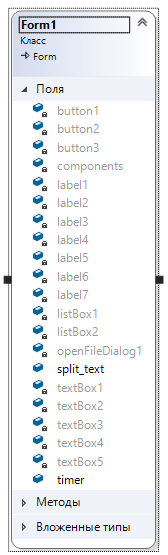
2020г.

**1). Описание задания**

Разработать программу, реализующую работу с файлами.

1. Программа должна быть разработана в виде приложения Windows Forms на языке C#. По желанию вместо Windows Forms возможно использование WPF.
2. Добавить кнопку, реализующую функцию чтения файла в список слов List<string>.
3. Для выбора имени файла используется класс OpenFileDialog, который открывает диалоговое окно с выбором файла. Ограничить выбор только файлами с расширением «.txt».
4. Для чтения из файла рекомендуется использовать статический метод ReadAllText() класса File (пространство имен System.IO). Содержимое файла считывается методом ReadAllText() в виде одной строки, далее делится на слова с использованием метода Split() класса string. Слова сохраняются в список List<string>.
5. При сохранении слов в список List<string> дубликаты слов не записываются. Для проверки наличия слова в списке используется метод Contains().
6. Вычислить время загрузки и сохранения в список с использованием класса Stopwatch (пространство имен System.Diagnostics). Вычисленное время вывести на форму в поле ввода (TextBox) или надпись (Label).
7. Добавить на форму поле ввода для поиска слова и кнопку поиска. При нажатии на кнопку поиска осуществлять поиск введенного слова в списке. Слово считается найденным, если оно входит в элемент списка как подстрока (метод Contains() класса string).
8. Добавить на форму список (ListBox). Найденные слова выводить в список с использованием метода «название\_списка.Items.Add()». Вызовы метода «название\_списка.Items.Add()» должны находится между вызовами методов «название\_списка.BeginUpdate()» и «название\_списка. EndUpdate()».
9. Вычислить время поиска с использованием класса Stopwatch. Вычисленное время вывести на форму в поле ввода (TextBox) или надпись (Label).

**2). Диаграмма классов**



**3). Текст программы**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Diagnostics;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Threading;

using System.IO;

namespace lab4\_сишарп

{

public partial class Form1 : Form

{

public static List<string> split\_text = new List<string>();

public static Stopwatch timer = new Stopwatch();

public Form1()

{

InitializeComponent();

}

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

{

OpenFileDialog ofd1 = new OpenFileDialog();

ofd1.Filter = "Text files(\*.txt)|\*.txt";

if (ofd1.ShowDialog() == DialogResult.Cancel)

{

MessageBox.Show("Необходимо выбрать файл");

return;

}

Stopwatch stopWatch = new Stopwatch();

stopWatch.Start();

// читаем файл в строку

string fileText = System.IO.File.ReadAllText(ofd1.FileName);

//добавляем разделители

char[] separators = new char[] { ' ', '.', ',', '!', '?', '/', '\t', '\n' };

//получаем список слов

string[] split\_text\_temp = fileText.Split(separators);

//убираем дубликаты

foreach (string str in split\_text\_temp)

{

if (!split\_text.Contains(str))

split\_text.Add(str);

}

stopWatch.Stop();

this.textBox1.Text = stopWatch.Elapsed.ToString();

//this.textBoxFileReadCount.Text = list.Count.ToString();

}

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

{

//определяем слово для поиска в нижнем регистре

string word = this.textBox2.Text.Trim().ToLower();

int potok\_kol = int.Parse(this.textBox5.Text);

if (string.IsNullOrWhiteSpace(word) || split\_text.Count == 0)

{

MessageBox.Show("Необходимо выбрать файл и ввести слово для поиска");

return;

}

List<string> find\_list = new List<string>();

//Временные результаты поиска

Stopwatch stopWatch = new Stopwatch();

stopWatch.Start();

foreach (string str in split\_text)

{

if (str.Contains(word))

find\_list.Add(str);

}

stopWatch.Stop();

this.textBox3.Text = stopWatch.Elapsed.ToString();

this.listBox1.BeginUpdate();

//Очистка списка

this.listBox1.Items.Clear();

//Вывод результатов поиска

foreach (string str in find\_list)

{

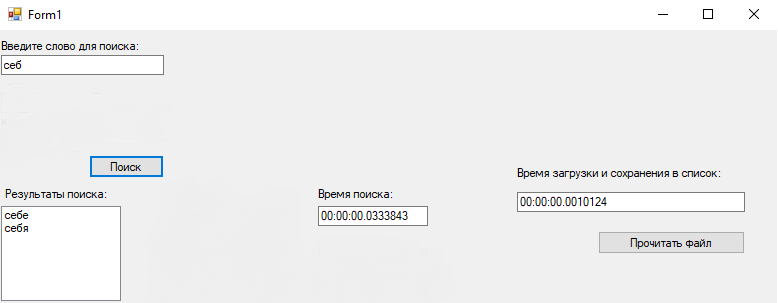
this.listBox1.Items.Add(str);

}

this.listBox1.EndUpdate();

}

**4). Пример выполнения**

****