**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №4

«Работа с файлами»

|  |  |  |
| --- | --- | --- |
| Выполнила: |  | Проверил: |
| студентка группы ИУ5-33 |  | преподаватель каф. ИУ5 |
| Беспалова У.А. |  | Гапанюк Ю.Е. |
| Подпись и дата:  16.10.2018 |  | Подпись и дата: |

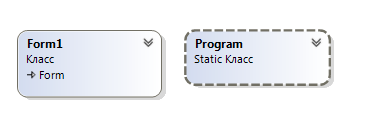
Москва, 2018 г.

1. Содержание 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).

Диаграмма классов



Текст программы

using System;

using System.IO;

using System.Diagnostics;

using System.Collections.Generic;

//using EditDistanceProject;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace лаба4

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent(); //просто не будем это трогать.

}

/// <summary>

/// Список слов

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

List<string> list = new List<string> ();//создали пустой лист

private void Form1\_Load(object sender, EventArgs e)

{

}

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

{

OpenFileDialog fileD = new OpenFileDialog();

fileD.Filter = "текстовые файлы|\*.txt"; //ограничиваемся текстовыми файлами

if (fileD.ShowDialog() == DialogResult.OK)

{

Stopwatch t = new Stopwatch(); //штучка для измерения времени

t.Start(); //начали отсчет

string text = File.ReadAllText(fileD.FileName); //чтение файла в виде строки

string[] textArray = text.Split(new char[] { ' ', '.', ',', '!', '?', '-', ';', ':', '/', '\t', '\n' }, StringSplitOptions.RemoveEmptyEntries); //разделяем строку на слова

foreach (string i in textArray)

{

string str = i.Trim(); //поудаляли пробелы перед и после слова

if (!list.Contains(str)) list.Add(str); //если в списке такого слова нет, добавляем его

}

t.Stop(); //остановили таймер

this.label4.Text = (t.ElapsedTicks\*0.0001).ToString(); //посчитали время

this.label1.Text = list.Count.ToString(); //посчитали количество записанных слов

}

else

{

MessageBox.Show("Нужно выбрать файл");

}

}

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

{

string word = this.textBoxFind.Text.Trim(); //введенное слово для поиска

if (!string.IsNullOrWhiteSpace(word) && list.Count > 0) //если слово не пустое

{

string wordUp = word.ToUpper();//перевели в верхний регистр

List<string> tempList = new List<string>();//пустой массив для найденных слов

Stopwatch t = new Stopwatch();

t.Start(); //начали отсчет

foreach (string i in list)

{

if (i.ToUpper().Contains(wordUp))

{

tempList.Add(i); //если нашлось слово из тех же букв, записываем во временный список

}

}

t.Stop();

this.labelTime2.Text = (t.ElapsedTicks \* 0.0001).ToString();

this.listBoxResults.Items.Clear(); //очистили список (чтоб не показывал предыдущие результаты)

this.listBoxResults.BeginUpdate();

foreach (string i in tempList)

{

this.listBoxResults.Items.Add(i);

}

this.listBoxResults.EndUpdate();

}

else

{

MessageBox.Show("Нужно ввести слово для поиска");

}

}

private void label5\_Click(object sender, EventArgs e)

{

}

}

}

Экранные формы с примерами выполнения программы

