Московский Государственный Университет им. Н.Э. Баумана

Отчет по лабораторной работе №4

по курсу БКИТ

# Выполнил:

## Громов Вадим

## ИУ5-31

# Условие задачи

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).

Код

## Файл: MainWindow.Xaml.cs

using System;

using System.IO;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using Microsoft.Win32;

using System.Diagnostics;

namespace Lab4

{

public partial class MainWindow : Window

{

List< String> list = new List<String>();

public MainWindow()

{

InitializeComponent();

}

private void Read\_File\_Button\_Click(object sender, RoutedEventArgs e)

{

OpenFileDialog Dialog\_one = new OpenFileDialog();

Dialog\_one.Filter = "text\_files|\*.txt";

if (Dialog\_one.ShowDialog()== true)

{

Stopwatch mytimer = new Stopwatch();

mytimer.Start();

string text = File.ReadAllText(Dialog\_one.FileName);

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

string[] textArray = text.Split(separators);

foreach (string strTemp in textArray)

{

string str = strTemp.Trim();

if (!list.Contains(str)) list.Add(str);

}

mytimer.Stop();

this.textbox\_for\_timer.Text = mytimer.Elapsed.ToString();

this.textbox\_for\_list.Text = list.Count.ToString();

}

else

{

MessageBox.Show("Please choose file");

}

}

private void Search\_button\_Click(object sender, RoutedEventArgs e)

{

string word = this.Inputwords.Text.Trim();

if (!string.IsNullOrWhiteSpace(word) && list.Count > 0 && word != "Input word you'd like to find")

{

string wordUpper = word.ToUpper();

List<string> tempList = new List<string>();

Stopwatch t = new Stopwatch();

t.Start();

foreach (string str in list)

{

if (str.ToUpper().Contains(wordUpper))

{

tempList.Add(str);

}

}

t.Stop();

this.Anothertimer.Text = t.Elapsed.ToString();

this.found\_words.Items.Clear();

foreach (string str in tempList)

{

this.found\_words.Items.Add(str);

}

}

else

{

MessageBox.Show("Input the word you'd like to find, please");

}

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

}

}

# Примеры работающей программы





