**Московский государственный технический университет**

**им. Н.Э. Баумана**

Кафедра ИУ–5

**Отчет по лабораторной работе №4-5 по курсу «Базовые компоненты интернет технологий»**

Выполнил:

студент гр. ИУ5-34

Окопный М.О.

\_\_\_\_\_\_\_\_

Проверил:

к.т.н., доцент

Гапанюк Ю.Е.

\_\_\_\_\_\_\_\_

**Москва – 2017 г**

1. **Цель выполнения лабораторной работы**
2. Разработать программу, реализующую работу с файлами, а также предусмотреть отдельное поле ввода для максимального расстояния. Если расстояние Левенштейна между двумя строками больше максимального, то строки считаются несовпадающими и не выводятся в список результатов.
3. **Текст программы**

**Form1.cs**

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.Diagnostics;

namespace LR4FT

{

public partial class Form1 : Form

{

Stopwatch sw1;

Stopwatch sw2;

string ishod;

string last;

string stroka;

List<string> slova;

string search;

string[] slovamas;

int bo;

public Form1()

{

InitializeComponent();

sw1 = new Stopwatch();

sw2 = new Stopwatch();

slova = new List<string>();

openFileDialog1.Filter = "Text files (\*.txt)|\*.txt";

openFileDialog1.Title = "Open";

openFileDialog1.InitialDirectory = @"C:\Users\KingD\Documents";

openFileDialog1.FileName = "1231.txt";

}

private void openFileDialog1\_FileOk(object sender, CancelEventArgs e)

{

}

private void check()

{

bo = 0;

foreach (string a in slovamas)

{

foreach (string s in slova)

{

if (s.Contains(a))

{

bo += 1;

}

bo += 0;

}

if (bo == 0)

{

slova.Add(a);

sw1.Stop();

label2.Text = sw1.ElapsedMilliseconds.ToString();

}

bo = 0;

}

}

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

{

if (openFileDialog1.ShowDialog() == DialogResult.OK)

{

sw1.Start();

textBox1.Clear();

textBox1.Text = File.ReadAllText(openFileDialog1.FileName);

stroka = File.ReadAllText(openFileDialog1.FileName);

slovamas = stroka.Split();

check();

}

}

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

{

listBox1.Items.Clear();

sw2.Start();

search = textBox2.Text;

foreach (string a in slova)

{

if (a.Contains(search))

listBox1.Items.Add(a);

}

sw2.Stop();

label3.Text = sw2.ElapsedMilliseconds.ToString();

}

private void listBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

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

{

listBox1.Items.Clear();

ishod = textBox3.Text;

int rast = Int32.Parse(textBox4.Text);

foreach (string x in slova)

{

last = x;

if (LevenshteinDistance(ishod, last) <= rast) listBox1.Items.Add(last);

}

}

public static int LevenshteinDistance(string string1, string string2)

{

if (string1 == null) throw new ArgumentNullException("string1");

if (string2 == null) throw new ArgumentNullException("string2");

int diff;

int[,] m = new int[string1.Length + 1, string2.Length + 1];

for (int i = 0; i <= string1.Length; i++) m[i, 0] = i;

for (int j = 0; j <= string2.Length; j++) m[0, j] = j;

for (int i = 1; i <= string1.Length; i++)

for (int j = 1; j <= string2.Length; j++)

{

diff = (string1[i - 1] == string2[j - 1]) ? 0 : 1;

m[i, j] = Math.Min(Math.Min(m[i - 1, j] + 1,

m[i, j - 1] + 1),

m[i - 1, j - 1] + diff);

}

return m[string1.Length, string2.Length];

}

}

}

**3. Результаты работы программы**



