**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №5

«Расстояние Левенштейна»

|  |  |  |
| --- | --- | --- |
| Выполнила: |  | Проверил: |
| студент группы ИУ5-32Б |  | преподаватель каф. ИУ5 |
| Перова Анна Евгеньевна |  | Гапанюк Юрий  Евгеньевич |
| Подпись и дата: |  | Подпись и дата:  23.11.20 |

Москва, 2020 г.

Задание

Разработать программу, реализующую вычисление расстояния Левенштейна с использованием алгоритма Вагнера-Фишера.

1. Программа должна быть разработана в виде библиотеки классов на языке C#.

2. Использовать самый простой вариант алгоритма без оптимизации.

3. Дополнительно возможно реализовать вычисление расстояния Дамерау-Левенштейна (с учетом перестановок соседних символов).

4. Модифицировать предыдущую лабораторную работу, вместо поиска подстроки используется вычисление расстояния Левенштейна.

5. 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.

Текст программы

**Class1.cs**

using System;

namespace laba5

{

public class Livenshtein

{

static int Min(int a, int b, int c)

{

if (a > b)

a = b;

if (a > c)

a = c;

return a;

}

static int Min(int a, int b) => a < b ? a : b;

// Расстояние Дамерау-Левенштейна

public int DamerauLivenshteinDist(string word1, string word2, bool limht)

{

var n = word1.Length + 1;

var m = word2.Length + 1;

var mD = new int[n, m];

for (var i = 0; i < n; i++)

mD[i, 0] = i;

for (var j = 0; j < m; j++)

mD[0, j] = j;

for (var i = 1; i < n; i++)

{

for (var j = 1; j < m; j++)

{

int Word = word1[i - 1] == word2[j - 1] ? 0 : 1;

mD[i, j] = Min(mD[i - 1, j] + 1, mD[i, j - 1] + 1 , mD[i - 1, j - 1] + Word );

if (i > 1 && j > 1 && word1[i - 1] == word2[j - 2] && word1[i - 2] == word2[j - 1] && !limht)

mD[i, j] = Min(mD[i, j], mD[i - 2, j - 2] + Word);

}

}

return mD[n - 1, m - 1];

}

}

}

**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;

using laba5;

namespace laba4\_5

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

bool b = true;

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

{

OpenFileDialog fd = new OpenFileDialog();

fd.Filter = "Текстовые файлы| \*.txt\*";

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

{

Stopwatch timer1 = new Stopwatch();

timer1.Start();

string text = File.ReadAllText(fd.FileName);

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

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

foreach (string strTemp in textArray)

{

string str = strTemp.Trim();

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

}

timer1.Stop();

this.textBox1.Text = timer1.Elapsed.ToString();

}

else

{

MessageBox.Show("Нужно выбрать файл");

}

}

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

{

Stopwatch timer = new Stopwatch();

timer.Start();

string word = textBox3.Text.ToLower();

if (!int.TryParse(textBox4.Text, out int distlim)|| distlim <0) distlim = 2;

Livenshtein lim = new Livenshtein();

int distance;

foreach (string str in list)

if ((distance = lim.DamerauLivenshteinDist( str, word, b)) <= distlim)

{

listBox1.BeginUpdate();

listBox1.Items.Add($"{word}, {str}, {distance}");

listBox1.EndUpdate();

timer.Stop();

textBox2.Text = timer.Elapsed.ToString();

break;

}

}

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

{

listBox1.Items.Clear();

}

private void radioButton1\_CheckedChanged(object sender, EventArgs e)

{

radioButton2.Checked = !radioButton1.Checked;

b = true;

label4.Text = "Максимальное расстояние Левенштейна";

}

private void radioButton2\_CheckedChanged(object sender, EventArgs e)

{

radioButton1.Checked = !radioButton2.Checked;

b = false;

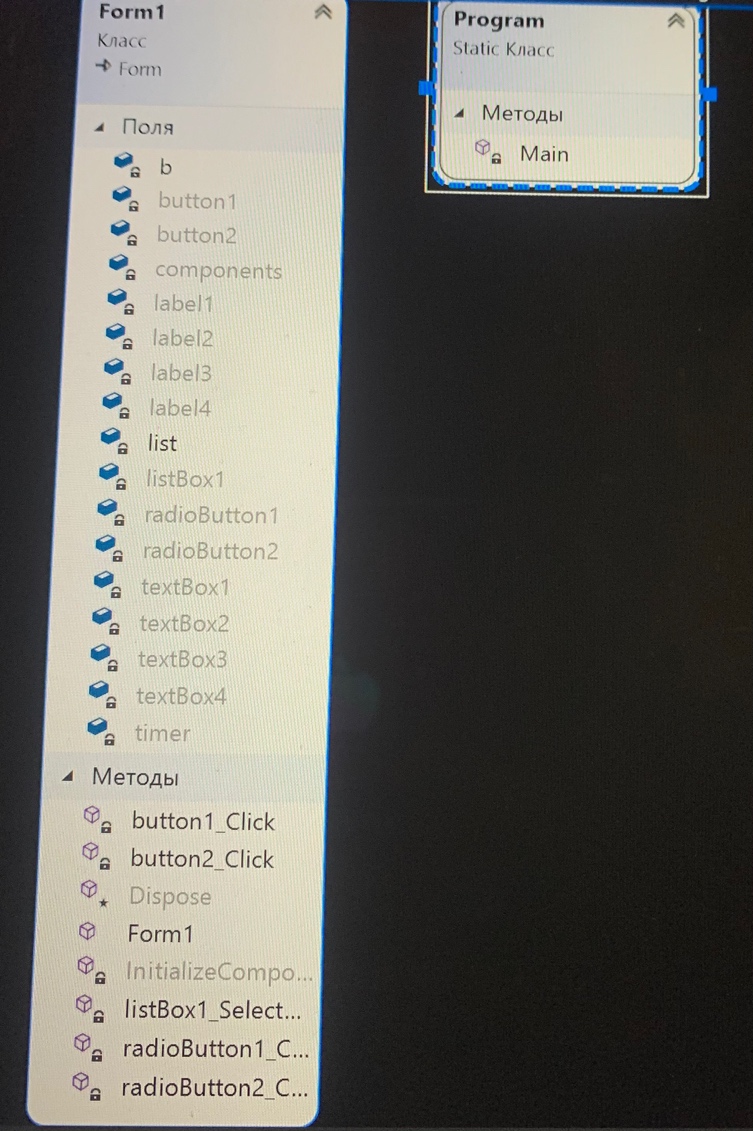
label4.Text = "Максимальное расстояние Дамерау-Левенштейна";

}

}

}

Диаграмма классов



Скриншоты работы программы

