

Московский государственный технический
университет имени Н. Э. Баумана.

Факультет “Радиотехнический”
Кафедра ИУ5 “Системы обработки информации и управления”

Отчёт к лабораторной работе №5
по курсу «Парадигмы и конструкции языков
программирования»

Выполнил:

Студент группы РТ5-31Б

Цыгичко А.Н.

Подпись и дата:

Проверил:

Преподаватель кафедры ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Москва, 2024 г.

Постановка задачи.

Лабораторная работа №5

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 Дamerau-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.

Контрольные вопросы:

Текст программы:

Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Lab5;

namespace Lab5ShowCase
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.Write("Первое слово: ");
            string s1 = Console.ReadLine();
            Console.Write("Второе слово: ");
            string s2 = Console.ReadLine();
            //Console.WriteLine(s1);
            //Console.WriteLine(s2);
            //Console.WriteLine("_____");

            Console.WriteLine("Расстояние Левенштейна: {0}",
Lab5.Levenshtine.LevenshteinDistance(s1, s2));
            Console.ReadLine();
        }
    }
}
```

Class1.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab5
{
    public class Levenshtine
    {
        public static int Minimum(int a, int b, int c) => (a = a < b ? a : b) < c ? a : c;

        public static int LevenshteinDistance(string firstWord, string secondWord)
        {
            int n = firstWord.Length + 1;
            int m = secondWord.Length + 1;
            int[,] matrixD = new int[n, m];

            for (var i = 0; i < n; i++)
            {
                matrixD[i, 0] = i;
            }

            for (var j = 0; j < m; j++)
            {
                matrixD[0, j] = j;
            }
            /* for (var i = 0; i < n; i++)
            {
                for (var j = 0; j < m; j++)
                {
                    Console.Write(matrixD[i, j]);
                    Console.Write(" ");
                }
                Console.WriteLine();
            }*/
            for (var i = 1; i < n; i++)
            {
                for (var j = 1; j < m; j++)
                {
                    var substitutionCost = firstWord[i - 1] == secondWord[j - 1] ? 0 : 1;
                    //Console.WriteLine(firstWord[i - 1]);
                    //Console.WriteLine(secondWord[j - 1]);
                    //Console.WriteLine("-");
                    matrixD[i, j] = Minimum(matrixD[i - 1, j] + 1,
                                            matrixD[i, j - 1] + 1,
                                            matrixD[i - 1, j - 1] + substitutionCost);
                }
            }

            return matrixD[n - 1, m - 1];
        }
    }
}
```

Результат:

```
Первое слово: apple  
Второе слово: banana  
Расстояние Левенштейна: 5
```