

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет «Информатика и вычислительная техника»

Кафедра «Кибербезопасность информационных систем»

Дисциплина «Методы программирования»

Лабораторная работа №3

на тему «Динамическое программирование»

Выполнила студентка группы ВКБ32

Куринский К.И.

г. Ростов-на-Дону

2021 г.

Задание:

Дана текстовая строка. С ней можно выполнять следующие операции:

1. Заменить один символ строки на другой символ.
2. Удалить один произвольный символ.
3. Вставить произвольный символ в произвольное место строки.

Например, при помощи первой операции из строки "СОК" можно получить строку "СУК", при помощи второй операции - строку "ОК", при помощи третьей операции - строку "СТОК.

Минимальное количество таких операций, при помощи которых можно из одной строки получить другую, называется стоимостью редактирования или расстоянием Левенштейна.

Определите расстояние Левенштейна для двух данных строк.

Входные данные:

Программа получает на вход две строки, длина каждой из которых не превосходит 1000 символов, строки состоят только из заглавных латинских букв.

Выходные данные:

Требуется вывести одно число – расстояние Левенштейна для данных строк.

Листинг задания:

import java.io.File;

import java.io.FileWriter;

import java.io.IOException;

import java.util.Scanner;

public class D {

static int minimin(int a, int b, int c) {

return Math.min(Math.min(a, b), c);

}

static int meme(char i, char j) {

if (i == j) return 0;

else return 1;

}

static int HereFishyFishy(String s1, String s2) {

int m = s1.length();

int n = s2.length();

int dp[][] = new int[m + 1][n + 1];

for (int i = 0; i <= m; i++) {

for (int j = 0; j <= n; j++) {

if (i == 0 && j == 0) {

dp[i][j] = 0;

continue;

}

if (i == 0 && j > 0) {

dp[i][j] = j;

continue;

}

if (i > 0 && j == 0) {

dp[i][j] = i;

continue;

}

if (j > 0 && i > 0) {

dp[i][j] = minimin(dp[i][j - 1] + 1, dp[i - 1][j] + 1, dp[i - 1][j - 1] + meme(s1.charAt(i - 1), s2.charAt(j - 1)));

}

}

}

return dp[m][n];

}

public static void main(String[] args) throws IOException {

Scanner scanner = new Scanner(new File("input.txt"));

FileWriter fileWriter = new FileWriter(new File("output.txt"));

String s1 = scanner.nextLine();

String s2 = scanner.nextLine();

fileWriter.write(HereFishyFishy(s1, s2) + "");

fileWriter.flush();

}

}

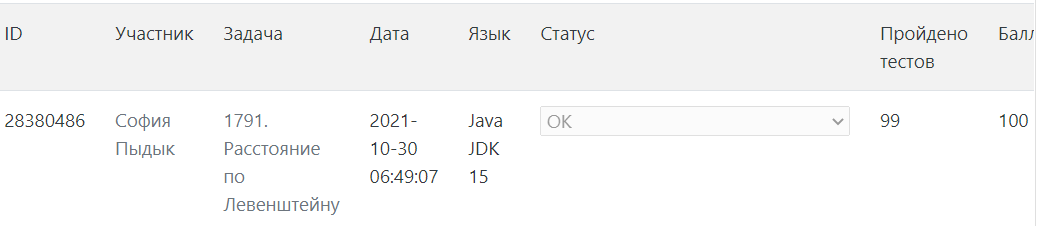


Рисунок 1 – результат работы задания.