

**Московский государственный технический  
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»  
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №2  
«Нахождение расстояния Левенштейна»

Выполнила:  
Студентка группы ИУ5-33Б  
Беспалова Виктория  
Подпись и дата:

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Ю. Е.  
Подпись и дата:

2023 год

## Описание задания

Разработать программу для нахождения расстояния Левенштейна

## Текст программы

```
import java.util.Scanner;

public class Main {
    public static int getLevenshteinDistance(String word1, String word2) {
        int m = word1.length();
        int n = word2.length();

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

        for (int i = 1; i <= m; i++) {
            table[i][0] = i;
        }
        for (int j = 1; j <= n; j++) {
            table[0][j] = j;
        }

        printTable(table);

        int value;
        for (int i = 1; i <= m; i++) {
            for (int j = 1; j <= n; j++) {
                value = word1.charAt(i - 1) == word2.charAt(j - 1) ? 0 : 1;
                table[i][j] = Integer.min(Integer.min(table[i][j - 1] + 1,
table[i - 1][j] + 1),
                    table[i - 1][j - 1] + value);
            }
        }

        printTable(table);

        return table[m][n];
    }

    public static String[] getWords() {
        Scanner in = new Scanner(System.in);
        String[] words = new String[2];
        for (int i = 0; i < 2; i++) {
            System.out.printf("Введите %d-е слово для сравнения:%n", i + 1);
            words[i] = in.nextLine();
        }
        return words;
    }

    public static void printTable(int[][] table) {
        System.out.println();
        for (int i = 1; i < table.length; i++) {
            for (int j = 1; j < table[i].length; j++) {
                System.out.print(table[i][j] + "\t");
            }
            System.out.println();
        }
    }

    public static void execute() {
        String[] words = getWords();
        System.out.println("Расстояние Левенштейна = " +
getLevenshteinDistance(words[0], words[1]));
    }
}
```

```

    }

    public static void main(String[] args) {
        execute();
    }
}

```

## Пример выполнения программы

Введите 1-е слово для сравнения:

ГИБРАЛТАР

Введите 2-е слово для сравнения:

ЛАБРАДОР

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

1	2	3	4	5	6	7	8
2	2	3	4	5	6	7	8
3	3	2	3	4	5	6	7
4	4	3	2	3	4	5	6
5	4	4	3	2	3	4	5
5	5	5	4	3	3	4	5
6	6	6	5	4	4	4	5
7	6	7	6	5	5	5	5
8	7	7	7	6	6	6	5

Расстояние Левенштейна = 5

Process finished with exit code 0

|

Введите 1-е слово для сравнения:

ХЛЕБ

Введите 2-е слово для сравнения:

ХЛЕВ

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

0	1	2	3
1	0	1	2
2	1	0	1
3	2	1	1

Расстояние Левенштейна = 1

Process finished with exit code 0

Разбор сравнения слов представлен в [статье](#).