

**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
«Расстояние Левенштейна»

Выполнил:
Студент группы ИУ5-33Б
Абрамов А. В.
Подпись и дата:

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

Москва, 2023 г.

Описание задания

Разработать программу для нахождения расстояния Левенштейна между двумя словами.

Текст программы Python

```
def levenshtein_distance(str1, str2):
    len1, len2 = len(str1), len(str2)
    dp = [[0] * (len2 + 1) for _ in range(len1 + 1)]

    for i in range(len1 + 1):
        dp[i][0] = i
    for j in range(len2 + 1):
        dp[0][j] = j

    for i in range(1, len1 + 1):
        for j in range(1, len2 + 1):
            if str1[i - 1] == str2[j - 1]:
                cost = 0
            else:
                cost = 1

            dp[i][j] = min(
                dp[i - 1][j] + 1, # Удаление
                dp[i][j - 1] + 1, # Вставка
                dp[i - 1][j - 1] + cost, # Замена
            )

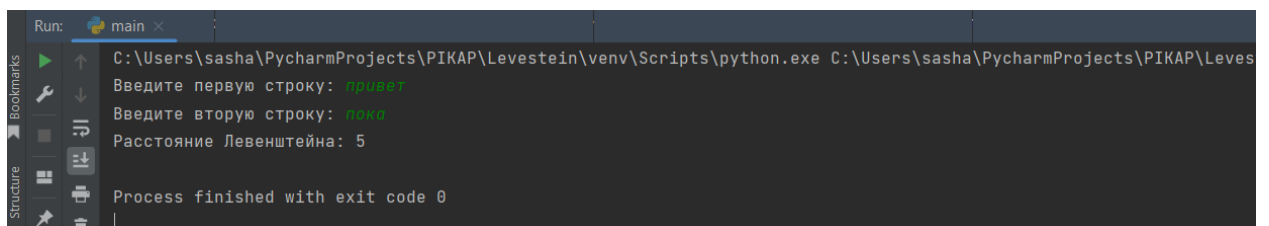
    return dp[len1][len2]

def main():
    print("Введите первую строку: ", end='')
    a = input()
    print("Введите вторую строку: ", end='')
    b = input()

    print("Расстояние Левенштейна: {}".format(levenshtein_distance(a, b)))

if __name__ == "__main__":
    main()
```

Примеры выполнения программы



```
Run: main x
C:\Users\sasha\PycharmProjects\PIKAP\Levestein\venv\Scripts\python.exe C:\Users\sasha\PycharmProjects\PIKAP\Leves
Введите первую строку: привет
Введите вторую строку: пока
Расстояние Левенштейна: 5
Process finished with exit code 0
```

Тест программы C#

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static int levenshtein_distance(string str1, string str2)
        {
            int len1 = str1.Length;
            int len2 = str2.Length;

            int[,] dp = new int[len1 + 1, len2 + 1];

            for (int i = 0; i <= len1; i++)
                dp[i, 0] = i;
            for (int j = 0; j <= len2; j++)
                dp[0, j] = j;

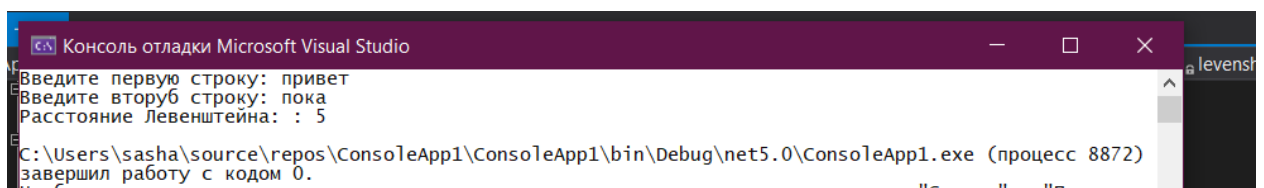
            for (int i = 1; i <= len1; i++)
            {
                for (int j = 1; j <= len2; j++)
                {
                    int cost = (str1[i - 1] == str2[j - 1]) ? 0 : 1;
                    dp[i, j] = Math.Min(
                        dp[i - 1, j] + 1,
                        Math.Min(
                            dp[i, j - 1] + 1,
                            dp[i - 1, j - 1] + cost
                        )
                    );
                }
            }

            return dp[len1, len2];
        }

        static void Main(string[] args)
        {
            Console.WriteLine("Введите первую строку: ");
            string str1 = Console.ReadLine();
            Console.WriteLine("Введите вторую строку: ");
            string str2 = Console.ReadLine();

            int distance = levenshtein_distance(str1, str2);
            Console.WriteLine($"Расстояние Левенштейна: {distance}");
        }
    }
}
```

Пример выполнения программы



```
Консоль отладки Microsoft Visual Studio
Введите первую строку: привет
Введите вторую строку: пока
Расстояние Левенштейна: 5
C:\Users\sasha\source\repos\ConsoleApp1\ConsoleApp1\bin\Debug\net5.0\ConsoleApp1.exe (процесс 8872)
завершил работу с кодом 0.
```

Текст программы тестов на BDD

```
from main import levenshtein_distance
import pytest

def test_equal():
    assert levenshtein_distance("Валера", "Валера") == 0

def test_diff():
    assert levenshtein_distance("Разведка", "Разводка") == 1

def test_empty():
    assert levenshtein_distance("", "Привет") == 6

if __name__ == '__main__':
    pytest.main()
```

Пример прохождения тестов

```
===== test session starts =====
collecting ... collected 3 items

BDDtests.py::test_equal PASSED [ 33%]
BDDtests.py::test_diff PASSED [ 66%]
BDDtests.py::test_empty PASSED [100%]

===== 3 passed in 0.05s =====
```

Текст программы тестов на TDD

```
import unittest
from main import levenshtein_distance

class TestLevenshteinDistance(unittest.TestCase):
    def test_equal(self):
        self.assertEqual(levenshtein_distance("Валера", "Валера"), 0)

    def test_diff(self):
        self.assertEqual(levenshtein_distance("Разведка", "Разводка"), 1)

    def test_empty(self):
        self.assertEqual(levenshtein_distance("", "Привет"), 6)

if __name__ == '__main__':
    unittest.main()
```

Пример прохождения тестов

```
===== test session starts =====  
collecting ... collected 3 items  
  
TDDtests.py::TestLevenshteinDistance::test_diff PASSED [ 33%]  
TDDtests.py::TestLevenshteinDistance::test_empty PASSED [ 66%]  
TDDtests.py::TestLevenshteinDistance::test_equal PASSED [100%]  
  
===== 3 passed in 0.03s =====
```