**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

«Расстояние Левенштейна»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-31Б |  | преподаватель каф. ИУ5 |
| Сомов Кирилл |  | Гапанюк Ю. Е. |
| Подпись и дата: |  | Подпись и дата: |
|  |  |  |

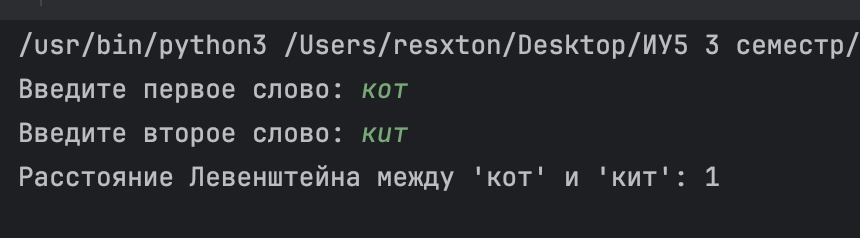
Москва, 2023 г.

Задание

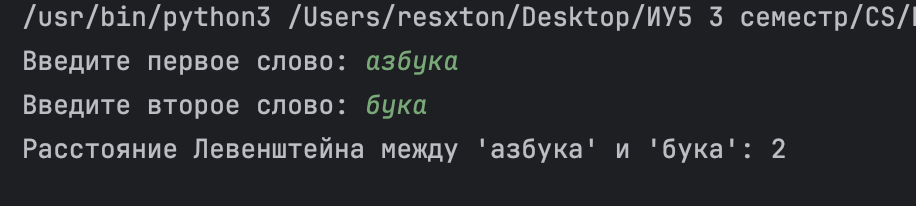
Написать программу, находящую расстояние Левенштейна для двух слов.

Код программы

def levenshtein\_distance(word1, word2):  
 m, n = len(word1), len(word2)  
 # Создаем матрицу размером (m+1) x (n+1)  
 dp = [[0] \* (n + 1) for \_ in range(m + 1)]  
  
 # Инициализируем первую строку и первый столбец  
 for i in range(m + 1):  
 dp[i][0] = i  
 for j in range(n + 1):  
 dp[0][j] = j  
  
 # Заполняем матрицу  
 for i in range(1, m + 1):  
 for j in range(1, n + 1):  
 if word1[i - 1] != word2[j - 1]:  
 dp[i][j] = min(dp[i - 1][j], dp[i][j - 1], dp[i - 1][j - 1]) + 1  
 else:  
 dp[i][j] = dp[i - 1][j - 1]  
  
 return dp[m][n]  
  
  
def main():  
 word1 = input("Введите первое слово: ")  
 word2 = input("Введите второе слово: ")  
  
 # Расстояние Левенштейна между словами  
 distance = levenshtein\_distance(word1, word2)  
 print(f"Расстояние Левенштейна между '{word1}' и '{word2}': {distance}")  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

Вывод

Действительно, расстояние Левенштейна для слов кит и кот равно 1.



Действительно, расстояние Левенштейна для слов азбука и бука равно 2.