

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №12
дисциплины «Алгоритмизация»

Выполнил:
Кожуховский Виктор Андреевич
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной
техники и автоматизированных систем
», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Порядок выполнения работы:

Написал программу поиска расстояния редактирования с использованием динамического программирования снизу вверх и сверху вниз:

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3
4 def edit_dist_dp(str1, str2):
5     m = len(str1)
6     n = len(str2)
7
8     # Таблицу для хранения результатов подзадач
9     dp = [[0 for x in range(n + 1)] for x in range(m + 1)]
10
11     # Заполняем d[i][j] снизу вверх
12     for i in range(m + 1):
13         for j in range(n + 1):
14
15             # Если первая строка пуста - вставить все символы второй строки
16             if i == 0:
17                 dp[i][j] = j
18
19             # Если вторая строка пуста - удалить все символы первой строки
20             elif j == 0:
21                 dp[i][j] = i
22
23             # Если последние символы одинаковы, игнорируем последний и рекурсивно вызываем для оставшейся строки
24             elif str1[i-1] == str2[j-1]:
25                 dp[i][j] = dp[i-1][j-1]
26
27             # Если последний символ отличается, рассматриваем все возможности и находим минимум
28             else:
29                 dp[i][j] = 1 + min(dp[i][j-1],          # Вставить
30                                   dp[i-1][j],          # Удалить
31                                   dp[i-1][j-1])         # Заменить
32
33     return dp[m][n]
34
35
36 def edit_dist_bu(A, B):
37     n = len(A)
38     m = len(B)
39
40     D = [[0 for x in range(m + 1)] for x in range(n + 1)]
41
42     # Инициализируем первую строку и первый столбец
43     for i in range(n + 1):
44         D[i][0] = i
45     for j in range(m + 1):
46         D[0][j] = j
47
48     # Заполняем остальную матрицу
49     for i in range(1, n + 1):
50         for j in range(1, m + 1):
51             c = 0 if A[i-1] == B[j-1] else 1
52             D[i][j] = min(D[i-1][j] + 1, D[i][j-1] + 1, D[i-1][j-1] + c)
53
54     return D[n][m]
55
56
57
58 if __name__ == "__main__":
59     str1 = input("Введите строку 1: ")
60     str2 = input("Введите строку 2: ")
61
62     result_dp = edit_dist_dp(str1, str2)
63     print(f"Результат edit_dist_dp: {result_dp}")
64
65     # Преобразование строк в списки для edit_dist_bu
66     A = list(str1)
67     B = list(str2)
68
69     result_bu = edit_dist_bu(A, B)
70     print(f"Результат edit_dist_bu: {result_bu}")
71
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS GITLENS SEARCH ERROR

PS C:\Users\viktor> & "C:/Program Files/Python311/python.exe" "c:/Users/viktor/Desktop/соды/алгоритмизация/AlgLab12/Main 1.py"

Введите строку 1: Введенная строка

Введите строку 2: Введенное слово

Результат edit_dist_dp: 6

Результат edit_dist_bu: 6

Рисунок 1. Код неоптимизированного алгоритма heapsort

Вывод: в результате выполнения лабораторной работы были изучены алгоритмы heap sort поиска расстояния редактирования с использованием динамического программирования снизу вверх и сверху вниз.