

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Северо-Кавказский федеральный университет»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №12  
по дисциплине «Алгоритмизация»**

Выполнил студент группы ИВТ-б-о-22-1

Пушкин Н.С. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверил Воронкин Р.А. \_\_\_\_\_  
(подпись)

Ставрополь 2023

## Порядок выполнения работы:

Решения задачи поиска расстояния редактирования и восстановления решения используя динамическое программирование как снизу-вверх, так и сверху вниз

```
1 def main():
2     str1 = "editing"
3     str2 = "distance"
4     result = edit_dist(str1, str2)
5
6     for item in result[1]:
7         print(f"{item[0]} {item[1]}")
8
9 1 usage
10 def edit_dist(a, b):
11     matrix = [[0] * (len(b) + 1) for _ in range(len(a) + 1)]
12
13     def edit_dist_td(i, j):
14         nonlocal matrix
15         if i == 0:
16             matrix[i][j] = j
17         elif j == 0:
18             matrix[i][j] = i
19         else:
20             ins = edit_dist_td(i, j - 1) + 1
21             delete = edit_dist_td(i - 1, j) + 1
22             sub = edit_dist_td(i - 1, j - 1) + (0 if a[i - 1] == b[j - 1] else 1)
23             matrix[i][j] = min(ins, delete, sub)
24     return matrix[i][j]
25
26     def edit_dist_bu():
27         nonlocal matrix
28         for i in range(len(a) + 1):
29             matrix[i][0] = i
30         for j in range(1, len(b) + 1):
31             matrix[0][j] = j
32         for i in range(1, len(a) + 1):
33             for j in range(1, len(b) + 1):
34                 c = 0 if a[i - 1] == b[j - 1] else 1
35                 matrix[i][j] = min(matrix[i - 1][j] + 1, matrix[i][j - 1] + 1, matrix[i - 1][j - 1] + c)
36
37     def restore():
38         result = []
39         i, j = len(a), len(b)
40         while i != 0 or j != 0:
41             if i != 0 and matrix[i][j] == matrix[i - 1][j] + 1:
42                 result.append((a[i - 1], '-'))
43                 i -= 1
44             elif j != 0 and matrix[i][j] == matrix[i][j - 1] + 1:
45                 result.append('-', b[j - 1])
46                 j -= 1
47             elif matrix[i][j] == matrix[i - 1][j - 1] + (0 if a[i - 1] == b[j - 1] else 1):
48                 result.append((a[i - 1], b[j - 1]))
49                 i -= 1
50                 j -= 1
51         result.reverse()
52         return result
53
54     edit_dist_td(len(a), len(b))
55     edit_dist_bu()
56     solution = restore()
57
58     return (matrix[len(a)][len(b)], solution)
59
60 if __name__ == "__main__":
61     main()
```

## Результат

```
C:\Users\НИКИТА\Desktop  
e -  
d d  
i i  
- s  
t t  
i a  
n n  
g c  
- e
```