K06

代码：

```
def recWordEdit(original, target, i, j, oplist):
    m = {(a, b): None for a in range(i) for b in range(j)}
    operations = []
    score = 0
    ops = []
    if j == 0:
        for _ in range(i):
            m[_, 0] = 'delete ' + original[_]
            score += oplist['delete']
    elif i == 0:
        for _ in range(j):
            m[0, _] = 'insert ' + target[_]
            score += oplist['insert']
    else:
        if original[i - 1] == target[j - 1]:
            ops.append(
                [recWordEdit(original, target, i - 1, j - 1, oplist)[0] + oplist['copy'],
                 str(recWordEdit(original, target, i - 1, j - 1, oplist)[1]) + 'copy ' + original[i - 1]])
        ops.append([recWordEdit(original, target, i, j - 1, oplist)[0] + oplist['insert'],
                    str(recWordEdit(original, target, i, j - 1, oplist)[1]) + 'insert ' + target[j - 1]])
        ops.append(
            [recWordEdit(original, target, i - 1, j, oplist)[0] + oplist['delete'],
             str(recWordEdit(original, target, i - 1, j, oplist)[1]) + 'delete ' + original[i - 1]])
        op = min(ops, key=lambda x: x[0])
        score += op[0]
        operations = op[1]
    return score, operations


# 检验
print("========== 2 单词最小编辑距离问题 ==========")
oplist = {'copy': 5, 'delete': 20, 'insert': 20}
# oplist = {'copy':5, 'delete':10, 'insert':15}
# oplist = {'copy':10, 'delete':25, 'insert':20}
```

```
originalWords = [
    "cane", "sheep", "algorithm", "debug", "difficult", "directory",
    "wonderful"
]
targetWords = [
    "new", "sleep", "alligator", "release", "sniffing", "framework", "terrific"
]
for i in range(len(originalWords)):
    score, operations = dpWordEdit(originalWords[i], targetWords[i], oplist)

    print(score)
    print(operations)
```

运行结果：

```
========= 2 单词最小编辑距离问题 =========
70
['delete e', 'delete c', 'delete a', 'copy n', 'copy e']
60
['delete p', 'copy s', 'delete h', 'copy e', 'copy e', 'copy p']
185
['delete m', 'insert l', 'insert l', 'insert i', 'copy a', 'copy l', 'copy g', 'delete o', 'delete r', 'delete i', 'copy t', 'delete h', 'copy m']
205
['delete g', 'insert e', 'insert l', 'insert e', 'insert a', 'insert s', 'delete d', 'copy e', 'delete b', 'delete u', 'delete g']
200
['delete t', 'insert n', 'insert i', 'insert f', 'delete d', 'copy i', 'copy f', 'copy f', 'copy i', 'delete c', 'delete u', 'delete l', 'delete t']
220
['delete y', 'insert r', 'insert a', 'insert m', 'insert e', 'delete d', 'delete i', 'copy r', 'copy e', 'delete c', 'delete t', 'copy o', 'copy r', 'delete y']
235
['delete l', 'insert e', 'insert r', 'insert r', 'insert i', 'delete w', 'delete o', 'delete n', 'delete d', 'copy e', 'copy r', 'copy f', 'delete u', 'delete l']


Process finished with exit code 0
```

========= 2 单词最小编辑距离问题 =========

70

['delete e', 'delete c', 'delete a', 'copy n', 'copy e']

60

['delete p', 'copy s', 'delete h', 'copy e', 'copy e', 'copy p']

185

['delete m', 'insert l', 'insert l', 'insert i', 'copy a', 'copy l', 'copy g', 'delete o', 'delete r', 'delete i', 'copy t', 'delete h', 'copy m']

205

['delete g', 'insert e', 'insert l', 'insert e', 'insert a', 'insert s', 'delete d', 'copy e', 'delete b', 'delete u', 'delete g']

200

['delete t', 'insert n', 'insert i', 'insert f', 'delete d', 'copy i', 'copy f', 'copy f', 'copy i', 'delete c', 'delete u', 'delete l', 'delete t']

220

['delete y', 'insert r', 'insert a', 'insert m', 'insert e', 'delete d', 'delete i', 'copy r', 'copy e', 'delete c', 'delete t', 'copy o', 'copy r', 'delete y']

```
235
['delete l', 'insert e', 'insert r', 'insert r', 'insert i',
'delete w', 'delete o', 'delete n', 'delete d', 'copy e',
'copy r', 'copy f', 'delete u', 'delete l']

Process finished with exit code 0
```