【H3】动态规划作业

```
#uuid_share#  63c7b311-0fb4-4ede-ac43-c549b0f1114a  #
# DSA'21 课程上机作业
#  【H3】动态规划作业
#
#  说明：为方便批改作业，请同学们在完成作业时注意并遵守下面规则：
#  （1）直接在本文件中的*函数体内*编写代码，每个题目的函数后有调用语句用于检验
#  （2）如果作业中对相关类有明确命名/参数/返回值要求的，请严格按照要求执行
#  （3）有些习题会对代码的编写进行特殊限制，请注意这些限制并遵守
#  （4）作业在 4 月 28 日 18:00 之前提交到 Canvas 系统



# ========== 1  博物馆大盗问题  ==========
#  给定一个宝物列表 treasureList = [{'w': 2,'v': 3}, {'w': 3,'v': 4}, ...]
#  注意：每样宝物只有 1 个。
#  这样 treasureList[0]['w']就是第一件宝物的重量，等等
#  给定包裹最多承重 maxWeight > 0
#  实现一个函数，根据以上条件得到最高总价值以及对应的宝物
#  参数：宝物列表 treasureList，背包最大承重 maxWeight
#  返回值：最大总价值 maxValue，选取的宝物列表 choosenList(格式同 treasureList)
def dpMuseumThief(treasureList, maxWeight):
    treasureList = list(treasureList)
    treasureList.insert(0, None)
    #print(treasureList)
    maxValue = {(i, w): 0 for i in range(len(treasureList))
                for w in range(maxWeight + 1)}
    choosenList = []
    for i in range(1, len(treasureList)):
        for w in range(1, maxWeight + 1):
            if treasureList[i]['w'] > w:
                maxValue[(i, w)] = maxValue[(i - 1, w)]
            else:
                maxValue[(i, w)] = max(
                    maxValue[(i - 1, w)],
                    maxValue[(i - 1, w - treasureList[i]['w'])] + treasureList[i]['v']
                )
                #print(i,w)
                #choosenList.append(maxValue[(i, w)])
    item = [0]*len(treasureList)
    #print(treasureList)
    #print(maxValue)
    def findWhat(i,j):
```

```
            if i-1>=0:
                if maxValue[(i,j)] == maxValue[(i-1,j)]:
                    item[i]=0
                    return findWhat(i-1,j)
                elif j - treasureList[i]['w'] >=0 and \
                    maxValue[(i,j)] == maxValue[(i-1, j - treasureList[i]['w'])]+treasureList[i]['v']:
                    item[i]=1
                    return findWhat(i-1,j-treasureList[i]['w'])
        findWhat(len(treasureList) - 1,maxWeight)
        #print(item)
        for k in range(len(item)):
            if item[k] !=0:
                choosenList.append(treasureList[k])

        return maxValue[(len(treasureList) - 1, maxWeight)],choosenList



# 检验
print("========== 1 博物馆大盗问题 ==========")
treasureList = [[{'w':2, 'v':3}, {'w':3, 'v':4}, {'w':4, 'v':8}, {'w':5, 'v':8}, {'w':9, 'v':10}]]
treasureList.append([{'w':1, 'v':2}, {'w':2, 'v':2}, {'w':2, 'v':3}, {'w':4, 'v':5}, {'w':4, 'v':6}, {'w':4, 'v':7}, {'w':5, 'v':7},
                     {'w':5, 'v':8}, {'w':6, 'v':8}, {'w':6, 'v':10}, {'w':7, 'v':10}, {'w':7, 'v':12}, {'w':8, 'v':12}, {'w':8, 'v':13}, {'w':9, 'v':14}, {'w':9, 'v':16}])
treasureList.append([{'w':1, 'v':2}, {'w':2, 'v':2}, {'w':2, 'v':3}, {'w':3, 'v':4}, {'w':3, 'v':5}, {'w':4, 'v':6}, {'w':4, 'v':7},
                     {'w':5, 'v':7}, {'w':5, 'v':8}, {'w':6, 'v':8}, {'w':6, 'v':10}, {'w':7, 'v':11}, {'w':7, 'v':12}, {'w':8, 'v':13},
                     {'w':8, 'v':14}, {'w':9, 'v':15}, {'w':9, 'v':16}, {'w':9, 'v':17}, {'w':10, 'v':17}, {'w':10, 'v':18}, {'w':11, 'v':18}])
treasureList.append([{'w':1, 'v':2}, {'w':2, 'v':2}, {'w':2, 'v':3}, {'w':3, 'v':4}, {'w':3, 'v':5}, {'w':4, 'v':5}, {'w':4, 'v':6},
                     {'w':5, 'v':6}, {'w':5, 'v':7}, {'w':6, 'v':8}, {'w':6, 'v':9}, {'w':7, 'v':10}, {'w':7, 'v':11}, {'w':8, 'v':12},
                     {'w':8, 'v':13}, {'w':9, 'v':14}, {'w':9, 'v':15}, {'w':9, 'v':16}, {'w':10, 'v':16}, {'w':10, 'v':17}, {'w':11, 'v':18},
                     {'w': 12, 'v': 18}, {'w': 12, 'v': 19}, {'w': 13, 'v': 20}, {'w': 13, 'v': 21}, {'w': 14, 'v': 21}, {'w': 14, 'v': 22}])
treasureList.append([{'w':1, 'v':2}, {'w':2, 'v':2}, {'w':2, 'v':3}, {'w':3, 'v':4}, {'w':3, 'v':5}, {'w':4, 'v':5}, {'w':4, 'v':6},
```

```
                              {'w':5, 'v':6}, {'w':5, 'v':7}, {'w':6, 'v':8}, {'w':6, 'v':9}, {'w':7, 'v':9},
{'w':7, 'v':10}, {'w':8, 'v':11},
                              {'w':8, 'v':12}, {'w':9, 'v':13}, {'w':9, 'v':14}, {'w':9, 'v':15}, {'w':10,
'v':16}, {'w':10, 'v':17}, {'w':11, 'v':18},
                              {'w': 11, 'v': 19}, {'w': 12, 'v': 20}, {'w': 13, 'v': 20}, {'w': 13, 'v': 21},
{'w': 14, 'v': 21}, {'w': 14, 'v': 22},
                              {'w': 14, 'v': 23}, {'w': 15, 'v': 24},{'w': 15, 'v': 25}, {'w': 16, 'v':
26},{'w': 17, 'v': 27}, {'w': 18, 'v': 28}])

maxWeightList = [20, 50, 80, 100, 150]
for i in range(len(treasureList)):
    maxValue, choosenList = dpMuseumThief(treasureList[i], maxWeightList[i])
    print(maxValue)
    print(choosenList)

# 可有多种取法，以下只给出一种符合条件的宝物列表
# 29
# [{'w':2, 'v':3}, {'w':4, 'v':8}, {'w':5, 'v':8}, {'w':9, 'v':10}]
# 83
# [{'w': 1, 'v': 2}, {'w': 2, 'v': 3}, {'w': 4, 'v': 7}, {'w': 5, 'v': 8}, {'w': 6, 'v': 10}, {'w': 7, 'v': 12},
{'w': 8, 'v': 12}, {'w': 8, 'v': 13}, {'w': 9, 'v': 16}]
# 139
# [{'w': 1, 'v': 2}, {'w': 3, 'v': 5}, {'w': 4, 'v': 6}, {'w': 4, 'v': 7}, {'w': 6, 'v': 10}, {'w': 7, 'v': 12},
{'w': 8, 'v': 14}, {'w': 9, 'v': 15}, {'w': 9, 'v': 16}, {'w': 9, 'v': 17}, {'w': 10, 'v': 17}, {'w': 10, 'v':
18}]
# 164
# [{'w': 1, 'v': 2}, {'w': 3, 'v': 5}, {'w': 8, 'v': 13}, {'w': 9, 'v': 15}, {'w': 9, 'v': 16}, {'w': 10, 'v':
16}, {'w': 10, 'v': 17}, {'w': 11, 'v': 18}, {'w': 12, 'v': 19}, {'w': 13, 'v': 21}, {'w': 14, 'v': 22}]
# 246
# [{'w': 1, 'v': 2}, {'w': 3, 'v': 4}, {'w': 3, 'v': 5}, {'w': 9, 'v': 15}, {'w': 10, 'v': 17}, {'w': 11, 'v':
18}, {'w': 11, 'v': 19}, {'w': 12, 'v': 20}, {'w': 13, 'v': 21}, {'w': 14, 'v': 23}, {'w': 15, 'v': 24},
{'w': 15, 'v': 25}, {'w': 16, 'v': 26}, {'w': 17, 'v': 27}]


# ========== 2 单词最小编辑距离问题 ==========
# 实现一个函数，给定两个单词，得出从源单词变到目标单词所需要的最小编辑距离，返回总得分与编辑操作过程
# 可以进行的操作有：
# 从源单词复制一个字母到目标单词
# 从源单词删除一个字母
# 在目标单词插入一个字母
# 参数：两个字符串，即源单词 original 与目标单词 target，以及不同操作对应的分值，即一个字典
# 返回值：一个整数与一个列表，最低的分数与操作过程，示例见检验
```

```
## 编辑操作过程不一定唯一，给出一种满足条件的操作过程即可
def dpWordEdit(original, target, oplist):
    operations = []
    # 请在此编写你的代码（可删除 pass 语句）
    originalrow = len(original)+1#原来的词语长度+1 作为行
    targetline = len(target)+1#变过去的词语长度+1 作为列

    scoreList = [[0 for _ in range(targetline)] for _ in range(originalrow)]#二维列表 行是原来的长度 列是目标字符
    for _ in range(targetline):#遍历第一行的每个元素
        scoreList[0][_] = _ * oplist['insert']#假如每一个字符都插入的话 对应到每一个的积分
        operations.append('insert %s' % target[_ - 1])#记录
    for _ in range(originalrow):#删除每一列的第一个元素 依次增加
        scoreList[_][0] = _ * oplist['delete']
        operations.append('delete %s' % original[_ - 1])
    for i in range(1, originalrow):#遍历行
        for j in range(1, targetline):#遍历列
            delete = scoreList[i - 1][j] + oplist['delete']#数字 那一点的积分值 加上对应的分数
            insert = scoreList[i][j - 1] + oplist['insert']
            copy = scoreList[i - 1][j - 1] + oplist['copy']
            if original[i - 1] == target[j - 1] and\
                    ((copy < delete)
                     and (copy < insert)):
                scoreList[i][j] = copy
                operations.append('copy %s' % original[i - 1])
            elif delete < insert:
                scoreList[i][j] = delete
                operations.append('delete %s' % original[i - 1])
            else:
                scoreList[i][j] = insert
                operations.append('insert %s' % target[j - 1])
    result = scoreList[originalrow-1][targetline-1]
    # 代码结束
    return result, operations



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

#  操作所对应的分数可调整
# oplist = {'copy':5, 'delete':20, 'insert':20}
# 70
# ['delete c', 'delete a', 'copy n', 'copy e', 'insert w']
# 60
# ['copy s', 'insert l', 'delete h', 'copy e', 'copy e', 'copy p']
# 185
# ['copy a', 'copy l', 'insert l', 'insert i', 'copy g', 'insert a', 'insert t', 'copy o', 'copy r',
'delete i', 'delete t', 'delete h', 'delete m']
# 205
```

['insert r', 'delete d', 'copy e', 'insert l', 'insert e', 'insert a', 'insert s', 'insert e', 'delete b', 'delete u', 'delete g']
200
['insert s', 'insert n', 'delete d', 'copy i', 'copy f', 'copy f', 'copy i', 'insert n', 'insert g', 'delete c', 'delete u', 'delete l', 'delete t']
220
['insert f', 'delete d', 'delete i', 'copy r', 'insert a', 'insert m', 'copy e', 'insert w', 'delete c', 'delete t', 'copy o', 'copy r', 'insert k', 'delete y']
235
['insert t', 'delete w', 'delete o', 'delete n', 'delete d', 'copy e', 'copy r', 'insert r', 'insert i', 'copy f', 'insert i', 'insert c', 'delete u', 'delete l']
#
45
['delete c', 'delete a', 'copy n', 'copy e', 'insert w']
45
['copy s', 'insert l', 'delete h', 'copy e', 'copy e', 'copy p']
125
['copy a', 'copy l', 'insert l', 'insert i', 'copy g', 'insert a', 'insert t', 'copy o', 'copy r', 'delete i', 'delete t', 'delete h', 'delete m']
135
['insert r', 'delete d', 'copy e', 'insert l', 'insert e', 'insert a', 'insert s', 'insert e', 'delete b', 'delete u', 'delete g']
130
['insert s', 'insert n', 'delete d', 'copy i', 'copy f', 'copy f', 'copy i', 'insert n', 'insert g', 'delete c', 'delete u', 'delete l', 'delete t']
145
['insert f', 'delete d', 'delete i', 'copy r', 'insert a', 'insert m', 'copy e', 'insert w', 'delete c', 'delete t', 'copy o', 'copy r', 'insert k', 'delete y']
150
['insert t', 'delete w', 'delete o', 'delete n', 'delete d', 'copy e', 'copy r', 'insert r', 'insert i', 'copy f', 'insert i', 'insert c', 'delete u', 'delete l']
#
90
['delete c', 'delete a', 'copy n', 'copy e', 'insert w']
85
['copy s', 'insert l', 'delete h', 'copy e', 'copy e', 'copy p']
230
['copy a', 'copy l', 'insert l', 'insert i', 'copy g', 'insert a', 'insert t', 'copy o', 'copy r', 'delete i', 'delete t', 'delete h', 'delete m']
230
['insert r', 'delete d', 'copy e', 'insert l', 'insert e', 'insert a', 'insert s', 'insert e', 'delete b', 'delete u', 'delete g']
245
['insert s', 'insert n', 'delete d', 'copy i', 'copy f', 'copy f', 'copy i', 'insert n', 'insert g',

'delete c', 'delete u', 'delete l', 'delete t']
265
['insert f', 'delete d', 'delete i', 'copy r', 'insert a', 'insert m', 'copy e', 'insert w', 'delete c', 'delete t', 'copy o', 'copy r', 'insert k', 'delete y']
280
['insert t', 'delete w', 'delete o', 'delete n', 'delete d', 'copy e', 'copy r', 'insert r', 'insert i', 'copy f', 'insert i', 'insert c', 'delete u', 'delete l']

```
['insert w', 'insert n', 'insert e', 'insert w', 'delete e', 'delete c', 'delete a', 'delete n', 'delete e', 'insert n', 'insert e', 'insert w', 'i
45
['insert p', 'insert s', 'insert l', 'insert e', 'insert e', 'insert p', 'delete p', 'delete s', 'delete h', 'delete e', 'delete e', 'delete p', 'd
125
['insert r', 'insert a', 'insert l', 'insert l', 'insert i', 'insert g', 'insert a', 'insert t', 'insert o', 'insert r', 'delete m', 'delete a', 'd
135
['insert e', 'insert r', 'insert e', 'insert l', 'insert e', 'insert a', 'insert s', 'insert e', 'delete g', 'delete d', 'delete e', 'delete b', 'd
130
['insert g', 'insert s', 'insert n', 'insert i', 'insert f', 'insert f', 'insert i', 'insert n', 'insert g', 'delete t', 'delete d', 'delete i', 'd
145
['insert k', 'insert f', 'insert r', 'insert a', 'insert m', 'insert e', 'insert w', 'insert o', 'insert r', 'insert k', 'delete y', 'delete d', 'd
150
['insert c', 'insert t', 'insert e', 'insert r', 'insert r', 'insert i', 'insert f', 'insert i', 'insert c', 'delete l', 'delete w', 'delete o', 'd
90
['insert w', 'insert n', 'insert e', 'insert w', 'delete e', 'delete c', 'delete a', 'delete n', 'delete e', 'insert n', 'insert e', 'insert w', 'i
85
['insert p', 'insert s', 'insert l', 'insert e', 'insert e', 'insert p', 'delete p', 'delete s', 'delete h', 'delete e', 'delete e', 'delete p', 'd
230
['insert r', 'insert a', 'insert l', 'insert l', 'insert i', 'insert g', 'insert a', 'insert t', 'insert o', 'insert r', 'delete m', 'delete a', 'd
230
['insert e', 'insert r', 'insert e', 'insert l', 'insert e', 'insert a', 'insert s', 'insert e', 'delete g', 'delete d', 'delete e', 'delete b', 'd
245
['insert g', 'insert s', 'insert n', 'insert i', 'insert f', 'insert f', 'insert i', 'insert n', 'insert g', 'delete t', 'delete d', 'delete i', 'd
265
['insert k', 'insert f', 'insert r', 'insert a', 'insert m', 'insert e', 'insert w', 'insert o', 'insert r', 'insert k', 'delete y', 'delete d', 'd
280
['insert c', 'insert t', 'insert e', 'insert r', 'insert r', 'insert i', 'insert f', 'insert i', 'insert c', 'delete l', 'delete w', 'delete o', 'd

Process finished with exit code 0
```

```
========== 1 博物馆大盗问题 ============
29
[{'w': 2, 'v': 3}, {'w': 4, 'v': 8}, {'w': 5, 'v': 8}, {'w': 9, 'v': 10}]
83
[{'w': 1, 'v': 2}, {'w': 2, 'v': 3}, {'w': 4, 'v': 7}, {'w': 5, 'v': 8}, {'w': 6, 'v': 10}, {'w': 7, 'v': 12}, {'w': 8, 'v': 12}, {'w': 8, 'v': 13
139
[{'w': 1, 'v': 2}, {'w': 3, 'v': 5}, {'w': 4, 'v': 7}, {'w': 5, 'v': 8}, {'w': 6, 'v': 10}, {'w': 7, 'v': 12}, {'w': 8, 'v': 13}, {'w': 8, 'v': 14
164
[{'w': 1, 'v': 2}, {'w': 3, 'v': 5}, {'w': 8, 'v': 13}, {'w': 9, 'v': 15}, {'w': 9, 'v': 16}, {'w': 10, 'v': 16}, {'w': 10, 'v': 17}, {'w': 11, 'v
246
[{'w': 1, 'v': 2}, {'w': 2, 'v': 3}, {'w': 3, 'v': 5}, {'w': 9, 'v': 14}, {'w': 9, 'v': 15}, {'w': 10, 'v': 16}, {'w': 10, 'v': 17}, {'w': 11, 'v'
========= 2 单词最小编辑距离问题 =========
70
['insert w', 'insert n', 'insert e', 'insert w', 'delete e', 'delete c', 'delete a', 'delete n', 'delete e', 'insert n', 'insert e', 'insert w', '
60
['insert p', 'insert s', 'insert l', 'insert e', 'insert e', 'insert p', 'delete p', 'delete s', 'delete h', 'delete e', 'delete e', 'delete p', '
185
['insert r', 'insert a', 'insert l', 'insert l', 'insert i', 'insert g', 'insert a', 'insert t', 'insert o', 'insert r', 'delete m', 'delete a', '
205
['insert e', 'insert r', 'insert e', 'insert l', 'insert e', 'insert a', 'insert s', 'insert e', 'delete g', 'delete d', 'delete e', 'delete b', '
200
['insert g', 'insert s', 'insert n', 'insert i', 'insert f', 'insert f', 'insert i', 'insert n', 'insert g', 'delete t', 'delete d', 'delete i', '
220
['insert k', 'insert f', 'insert r', 'insert a', 'insert m', 'insert e', 'insert w', 'insert o', 'insert r', 'insert k', 'delete y', 'delete d', '
235
['insert c', 'insert t', 'insert e', 'insert r', 'insert r', 'insert i', 'insert f', 'insert i', 'insert c', 'delete l', 'delete w', 'delete o', '
45
['insert w', 'insert n', 'insert e', 'insert w', 'delete e', 'delete c', 'delete a', 'delete n', 'delete e', 'insert n', 'insert e', 'insert w', '
45
['insert p', 'insert s', 'insert l', 'insert e', 'insert e', 'insert p', 'delete p', 'delete s', 'delete h', 'delete e', 'delete e', 'delete p', '
```