作业评讲

上机作业5

1.背包问题

问题描述

- 给定n种物品和一背包。物品i的重量是w_i，其价值为v_i，背包的容量为c。问应如何选择装入背包中的物品，使得装入背包中物品的总价值最大？

解题思路

- 运用“三定律”
- 基本结束条件：
  - 当背包容量耗尽，或者物品被装完的时候，结束你的算法。

$$K(w,0) = 0, \quad K(0, j) = 0$$

解题思路

- 运用“三定律”
- 向基本结束条件演进：
  - 考虑问题K(w,j)：
    - 如果问题的解中包含物品j，则拿掉它，剩下的物品构成问题K(w-w_j,j-1)的最优解
    - 如果不包含物品j，则可以直接考虑K(w,j-1)。
  - 也可以从此得出它的时间复杂度O(N*C)

$$\mathrm{K}(w, j) = \max\{K(w - w_j, j - 1) + v_j, K(w, j - 1)\}$$

解题思路

- 直观理解
  - 根据动态规划函数，用一个(n+1)×(C+1)的二维表V，V[i][j]表示把前i个物品装入容量为j的背包中获得的最大价值。

		0	1	2	3	4	…	…	17	18	19	20
	0											
$w_1=2$ $v_1=3$	1											
$w_2=3$ $v_2=4$	2											
$w_3=4$ $v_3=8$	3											
$w_4=5$ $v_4=8$	4											
$w_5=9$ $v_5=10$	5											

解题思路

- 程序伪码，翻译成Python语言就是作业答案。

```
Initialize all K(0,j) = 0 and all K(w,0) = 0
for j = 1 to n:
    for w = 1 to W:
        if w_j > w:  K(w,j) = K(w,j-1)
        else:  K(w,j) = max{K(w,j-1), K(w-w_j,j-1) + v_j}
return K(W,n)
```

扩展阅读

- 其他题目形式
  - 0/1背包问题，完全背包问题，多重背包问题
- 其他解决办法
  - 贪心法，回溯法，分支限界法

2.编辑距离

问题描述

- （与作业题目稍有不同）对于两个字符串a和b，通过插入删除替换的基本操作，我们可以把a变成b或b变成a，那么字符串a变成字符串b需要的最少基本字符操作步数称为字符串a和字符串b的编辑距离。
- 对齐两个单词，比如snowy和sunny。然后才知道如何编辑

S	–	N	O	W	Y
S	U	N	N	–	Y

Cost: 3

–	S	N	O	W	–	Y
S	U	N	–	–	N	Y

Cost: 5

问题描述

- 输入两个字符串：x[0…m]和y[0…n]。令解决x[0…i]和y[0…j]的编辑距离问题入手，记为E(i,j)。

解题思路

- 运用“三定律”
- 基本结束条件：
  - 考虑最后一个字符，有三种对齐方式
    - 第一种情况，产生1个cost（按作业要求是20）
    - 第二种情况，产生1个cost （按作业要求是20）
    - 第三种情况，如果x[i]=y[j]，不产生cost （按作业要求是5），否则产生1个cost（按作业要求是40）

$x[i]$	或者	$-$	或者	$x[i]$
$-$		$y[j]$		$y[j]$

解题思路

$x[i]$	或者	—	或者	$x[i]$
—		$y[j]$		$y[j]$

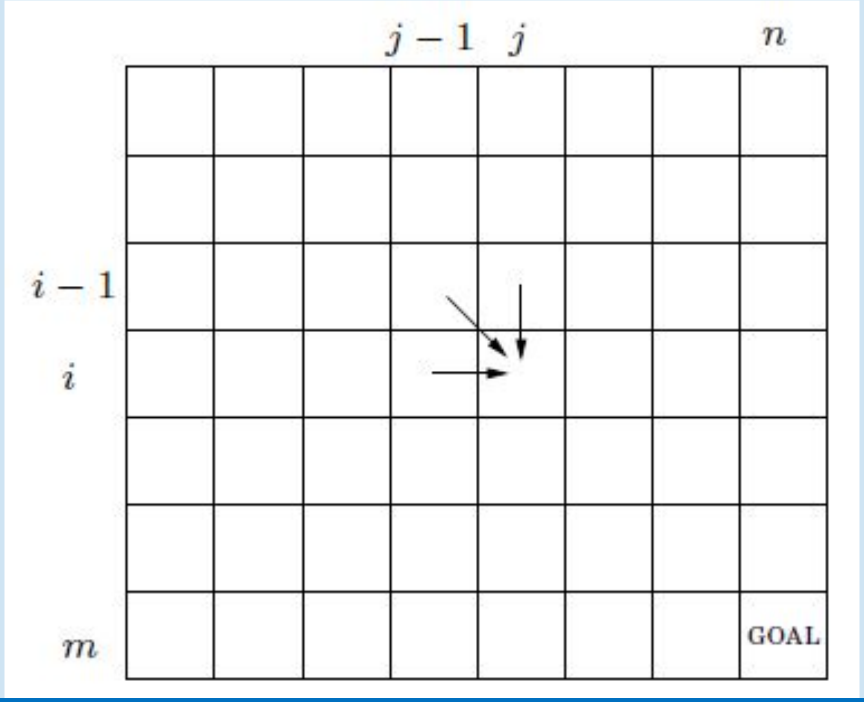
- 运用“三定律”
- 向基本结束条件演进：
  - –第一种情况，将E[i,j]变为E[i-1,j]
  - –第二种情况，将E[i,j]变为E[i,j-1]
  - –第三种情况，将E[i,j]变为E[i-1,j-1]
- 写成公式：

$$E[i,j] = \min\{1 + E(i-1,j), 1 + E(i,j-1),$$

$$diff(i,j) + E(i-1,j-1)\}$$

解题思路

- 直观理解
  - 分解和求解E[m,n]可以通过填一个表格来完成
    - 最终目标是计算E[m,n]，当前试图计算E[I,j]
    - 只要已知E[i-1,j]，E[i,j-1]和E[i-1,j-1]，就可以计算E[i,j]。



解题思路

- 程序伪码，翻译成Python语言就是作业答案。

```
for i = 0,1,2,...,m:
    E(i,0) = i
for j = 1,2,...,n:
    E(0,j) = j
for i = 1,2,...,m:
    for j = 1,2,...,n:
        E(i,j) = min{E(i−1,j)+1, E(i,j−1)+1, E(i−1,j−1)+diff(i,j)}
return E(m,n)
```

DS&A

Thank You

designed by Sun