13章习题答案

13章习题

- **1**、用1－优化法求解以下0/1背包问题：

  n = 8, w = [16, 20, 4, 15, 25, 10, 5, 8]，p = [100, 200, 50, 90, 175, 50, 20, 60], c = 70。
- 解

  密度为[6.25, 10 , 12.5 , 6 , 7 , 5 , 4 , 7.5].

  排序后物品顺序为[3 , 2 , 8 , 5 , 1 , 4 , 6 , 7].对应的物品

  重量为w’= [4 , 20 , 8 , 25 , 16 , 15 , 10 , 5], 效益为

  p’= [50, 200, 60, 175, 100, 90, 50, 20].

  k=0时计算结果为: x =(0 , 1 , 1 , 0 , 1 , 1 , 0 , 1),得到的效益值为535。

  k=1时计算结果为:

- 先放物品3、2、8、5，得到的效益值仍为535，贪心解同上；
- 先放物品1，得到效益值520，贪心解为（1，1，1，1，0，0，1，1）；
- 先放物品4，得到的效益值为520，贪心解为（1，1，1，1，0，0，1，1）；
- 先放物品6，得到效益值为535，贪心解为（0，1，1，0，1，1，0，1）；
- 先放物品7，得到效益值为505，贪心解为（0，1，1，0，1，1，1）。
- 所以k优化法得到的解为（0 , 1 , 1 , 0 , 1 , 1 , 0 , 1 ），效益值为535。

习题9．给定n个任务:1,2,…,n,执行任务i要求的时间是t_i. 如果任务按1,…,n的顺序执行则任务i的完成时间为

$$c_i = \sum_{j=1}^{i} t_j$$

求使得平均完成时间ACT=(c_1+c_2+…+c_n)/n最小的任务排序. (如果不是按初始给定的顺序,可重新调整任务的编号)

(c)按任务执行时间t_i值从小到大排序，则所得的调度的平均完成时间ACT最小的。

(e) 设在某任务顺序中，$i>j$而$t_i \le t_j$, 则交换作业i、j的顺序,得到一个新的执行顺序. 设原顺序的平均完成时间为ACT,改变后的平均完成时间为ACT'. 则ACT-ACT' ≤ 0.证明如下：

假设$i>j, t_i<t_j$(逆序)

$nACT=(nt_1+ \text{---}+(n-j+1)t_j+\ldots +(n-i+1)t_i+\ldots)$

$nACT'=(nt_1+ \text{---}+(n-j+1)t_i+\ldots +(n-i+1)t_j+\ldots)$

$nACT-nACT'=(i-j)t_i-(i-j)t_j=(i-j)(t_j-t_i)>0$

所以, ACT’<ACT.

又：每消除一个逆序ACT值减小,所以当无逆序时,也即任务按执行时间从小到大排列时ACT值最小.

习题17.连续背包问题

- 证明: 设$x=(x_1...,x_n)$为贪心法产生的解;则它有形式 $(1,1,..., x_j,0...0)$,其中 $0<x_j<1$;设 $y=(y_1...,y_n)$是优化解;
- 设k是$x_i \neq y_i$ 的最小下标.则$k \leq j$且$y_k<x_k$
- 将y_k增加到x_k,并从$\Sigma_{k<i\leq n}y_iw_i$ 减去$(x_k-y_k)w_k$ (无论什么方式)得到$(z_k,z_{k+1},\ldots,z_n)$, 其中$z_k=x_k$.
- 下面证明$(y_1... y_{k-1}, z_k,z_{k+1},\ldots,z_n)$仍是优化解:

习题17.连续背包问题

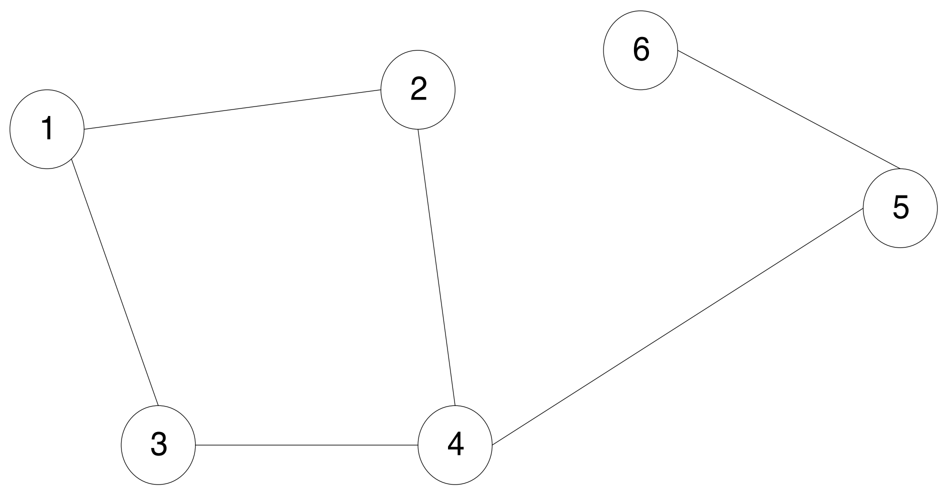
- 证明$\Sigma_{k\le i\le n} y_i p_i \le \Sigma_{k\le i\le n} z_i p_i$
- 将$(y_i-z_i)p_i$ 改写成$(y_i-z_i)w_i p_i/w_i$ 利用 $p_i/w_i \le p_k/w_k$, ($i>k$)和 $\Sigma_{k<i\le n}(y_i-z_i)w_i = (x_k-y_k)w_k$可得到上述不等式

22题

- 无向图的一个最大完全子图(包含意义下)称为一个集团,集团的节点数称为集团的尺寸(size),求图的最大集团.这是NP难度问题.本题要求给出一个贪心算法.
- 可考虑使用以下启发式: 如果一个节点的度数较大则它有可能在一个最大集团中.
- 算法首先将图的节点按度数排序然后从每个节点

- for j←1 to n do
- { S←{v_j}；
- for i←1 to n do
- {
- 检查S∪{v_i}是否构成完全子图；
- 如构成完全子图，将v_i加入到S中，否则舍弃v_i;
- 设该集团的节点数为d_j；
- }/*找出包含v_j的集团*/
- }
- 从d_j中找出最大者，输出对应的集团；
- (b)例如对书中图13.12(a)中的图，上述算法输出最大集团; 反例也能找到。

- 无向图的一种着色方案指将颜色标号赋给图的顶点，使得任意两个有边连接的顶点的颜色标号均不同。求使用最少数目的不同颜色标号的着色方案称为图着色问题，这是一NP难度问题。试按“标号小的颜色优先”的贪心策略设计一图着色问题的启发式算法。要求：
- (1)写出算法的伪代码；
- (2)就下面的图从节点1开始运行你设计的算法并在图上标出所得到的着色方案；
- (3)上述贪心策略能保证得到最优解吗？


1
2
3
4
5
6

- 解，（1）令i为节点号，c为颜色标号；算法的伪代码如下：
- for(i = 1; i≤n; i++)
-     for(c = 1; c≤n; c++)
-     If no vertex adjacent to i has color c then
-     Color i with c；
-     break； // exit for (c)
-     // Continue for (c)
- // Continue for (i) （10分）
- （2）算法对上图各节点的着色如图中各节点旁标记的数字所示，共使用了2种颜色。
- （3）否，该启发式算法不总是产生一个最优的着色方案。 在上图中如果按1,5,4,2,3,6 的顺序运行上述算法,则得到一个使用3种颜色 的着色方案。