第三次作业补充（来自大家的作业）

岳镝

Attention!!! 这次作业的 3.3, 3.5, 3.8, 3.10 都是背包问题，请尤其注意其形式（物品个数有没有限制，及其对应的递推方程和伪代码写法）

下面是我在大家作业中发现的一些答案里没有的做法，整理在此以供参考

3.5 解. 用 $f(x)$ 表示付 x 元钱所需的最小硬币重量。则关于 $f(x)$ 的递推方程为

$$f(x) = \min_{\substack{1\le i\le n\\ v_i\le x}} \{f(x-v_i)+w_i\};$$

$$f(0) = 0.$$

用 $t(x)$ 标记付 x 元钱使用的最后一枚硬币编号，则有

$$t(x) \in \operatorname*{argmin}_{\substack{1\le i\le n\\ v_i\le x}} \{f(x-v_i)+w_i\}.$$

算法的伪代码如下

Algorithm 1: COIN

```
Input: 重量 {w_i}_{i=1}^n，价值 {v_i}_{i=1}^n，总钱数 x ∈ ℕ
初始化 f(0) ← 0; f(j) ← +∞, j ≥ 1          // 自然语言描述，不要写一个循环
for j = 1, 2, ..., x do
    for i = 1, 2, ..., n do
        if v_i ≤ j 且 f(j − v_i) + w_i < f(j) then
            f(j) ← f(j − v_i) + w_i          // 习惯使用符号"←"，不要用"="
            t(j) ← i
return {f(j)}_{j=1}^x, {t(j)}_{j=1}^x
```

算法的时间复杂度为 $O(nx)$。 □

3.8 解. 当 N 为奇数时，划分显然不存在，下设 N 为偶数。

用 $f(i,j)$ 表示是否存在子集 $I \subseteq [i]$，使得 $\sum_{p\in I} a_p = j$。若存在这样的 I，则 $f(i,j)=1$，否则 $f(i,j)=0$。则递推方程为

$$
\begin{aligned}
&f(i,j) = \begin{cases} f(i-1,j) \vee f(i-1,j-a_i), & j \geq a_i, \\ f(i-1,j), & j < a_i \end{cases} \\
&f(0,0) = 1; \quad f(0,j) = 0, \forall j \geq 1.
\end{aligned}
$$

求出 $f(n, N/2)$ 即为答案。时间复杂度 $O(nN)$。 □

3.15 解. 对于 $i \in [n]$，考虑前 i 天的情形。用 $f(i,j)$ 表示前 i 天中，最后一次检修发生在第 j 天的情况下所能加工的任务数的最大值，其中 $0 \leq j \leq i$。特别地，$f(i,0)$ 对应着前 i 天从未发生检修，$f(i,i)$ 对应着第 i 天发生检修。

递推方程为

$$
\begin{aligned}
&f(i,j) = \begin{cases} f(i-1,j) + \min\{x_i, s_{i-j}\}, & 0 \leq j \leq i-1, \\ \max_{0\leq k\leq i-1} f(i-1,k), & j = i. \end{cases} \\
&f(0,0) = 0.
\end{aligned}
$$

最终答案为 $\max_{0\leq j\leq n} f(n,j)$。注意到 $j \neq i$ 时递推的时间复杂度为 $O(1)$，$j = i$ 时为 $O(n)$，故总时间复杂度为 $O(n^2)$。 □