《算法设计与分析》第4次答案

姓名：XXX　　　　　学号：XXXXXXXX

题目1：从哈尔滨到上海的高速公路上有若干个加油站。如果汽车从一个加油站出发时油箱是满的，则汽车可以顺利到达下一个加油站，若余油可以到下一个加油站，可以选择不加油。试设计一个汽车加油方案使得汽车在整个行驶路程中加油的次数最少，证明所给方案的正确性（需给出最优子结构和贪心选择性两方面的证明）。

答：

贪心思想：当汽车到达一个加油站时，若所剩汽油可以顺利到达下一格加油站，则不加油，否则加油。

贪心选择性：设在加满油后可行驶的N千米这段路程上任取两个加油站X和Y，且X距离起始点为m，Y距离起始点为n，且$m < n$。那么，若在Y加油站不能到达终点，那么在X加油一定不能到达终点，因为$m + N < n + N$，即在Y点加油可行驶的路程比在X点加油可行驶的路程要长$n - m$千米，根据贪心选择，为使加油次数最少就会选择距离加满油的点远一些的加油站去加油，因此，加油次数最少的这一问题具有贪心选择性质。

最优子结构：设公路沿途的加油站距起始位置的距离分别为$S[1 : n]$，需要加油的加油站编号$A[1 : k]$为加油问题的一个最优解，则$A[2 : k]$是剩余路程$S' = S[1 : n] - S[1 : A[1]]$的子问题的最优解。若不然，存在一个$S'$的加油问题的最优解$B'$，$|B'| < |A'|$，由于第一次加油不影响后面的加油策略，因此$B = B' \cup A[1]$是$S$的一个最优解，且$|B| = |B'| + 1 < |A'| + 1 = |A|$，与$A$最小矛盾，因此该问题具有最优子结构。

题目2：假设你是一位很棒的家长，想要给你的孩子们一些小饼干。但是，每个孩子最多只能给一块饼干。对每个孩子 i，都有一个胃口值 g_i，这是能让孩子们满足胃口的饼干的最小尺寸；并且每块饼干 j，都有一个尺寸 s_j。如果 $s_j \geq g_i$，我们可以将这个饼干 j 分配给孩子 i ，这个孩子会得到满足。你的目标是尽可能满足越多数量的孩子，并输出这个最大数值。注意：你可以假设胃口值为正。一个小朋友最多只能拥有一块饼干。试设计一个算法求解该问题，答案需包含以下内容：证明该问题的贪心选择性，描述算法思想并给出伪代码。示例如下：

输入: [1,2], [1,2,3]

输出: 2

解释: 你有两个孩子和三块小饼干，2个孩子的胃口值分别是1,2。你拥有的饼干数量和尺寸都足以让所有孩子满足。所以你应该输出2。

答：

算法思想：因为胃口值最小的孩子最容易吃饱，所以先满足了这个孩子之后，我们再采取同样的策略，考虑剩下孩子里胃口值最小的孩子，直到没有满足条件的饼干存在。因此这里的贪心策略是，给剩余孩子里最小胃口值的孩子分配最小的能饱腹的饼干。我们首先将孩子和饼干分别按由小到大进行排序，这样我们就可以从胃口值最小的孩子和大小最小的

饼干出发，计算有多少个饼干-孩子对可以满足条件。

贪心选择性：给剩余孩子里最小胃口值的孩子分配最小的能饱腹的饼干。

证明：假设每个孩子的胃口值和饼干的大小都已经按照从小到大的顺序排好了序。先证明最优解中包括孩子 1，然后假设最优解中包括孩子$1, 2..., k$, 且饼干还有,再证明最优解中包括孩子$k + 1$。具体如下：

1、如果最优解中不包括孩子 1，则可以用孩子 1 替换其他任意一个孩子，而且使得剩余的大份的饼干数量会更多。这种情况下，新解的价值一定大于等于最优解。因此最优解里面包含孩子1。

2、假设前k个小孩已经选择好饼干，第$k + 1$个小孩去选饼干，他如果没有选择饼干，而第$k + 2$个小孩拿到了饼干，那么把$k + 2$的饼干给$k + 1$，一样可以满足$k + 1$这个小孩，且拿到饼干的小孩总数没有变，所以$k + 1$小孩去选饼干必然会得出一个最优解。

伪代码：

Algorithm 1 贪心算法求解分饼干问题

Input: $g_1, g_2, ..., g_n$; $s_1, s_2, ..., s_n$;

Output: 能满足的孩子的数量

Sort($g_1, g_2, ..., g_n$);
Sort($s_1, s_2, ..., s_n$);
$child \leftarrow 0$;
$cookie \leftarrow 0$;
while $child < children.length$ and $cookie < cookies.length$ **do**
    if $g_{child} \leq s_{cookie}$ **then**
        $child \leftarrow child + 1$;
    end if
    $cookie \leftarrow cookie + 1$;
end while
return $child$;

题目3：现有一台计算机，在某个时刻同时到达了n个任务。该计算机在同一时间只能处理一个任务，每个任务都必须被不间断地得到处理。该计算机处理这n个任务需要的时间分别为$a_1, a_2, ...a_n$。将第i个任务在调度策略中的结束时间记为e_i，请设计一个贪心算法输出这n个任务的一个调度使得用户的平均响应时间$\frac{\sum e_i}{n}$达到最小。试设计一个算法求解该问题，答案需包含以下内容：证明该问题的贪心选择性，描述算法思想并给出伪代码。

答：

贪心思想：为了使平均响应时间最短，每次选择执行时间最短的任务

贪心选择性：不妨设编号为1的任务是n个任务中执行时间最短的，则只需要证明有一个最优解是优先执行任务1即可。那么，设$i_1, i_2, ..., i_n$是一个最优解，则第i_j个任务的等待时间

为：

$$w_{i_j} = \begin{cases} 0, & j = 1 \\ e_{i_{j-1}} = \sum_{k=1}^{j-1} a_{i_k}, & j \neq 1 \end{cases} \tag{1}$$

则平均响应时间T为：

$$T = \frac{1}{n}\sum_{j=1}^{n} w_{i_j} = \frac{1}{n}\sum_{j=1}^{n-1} e_{i_j} = \frac{1}{n}[(n-1)\cdot a_{i_1} + (n-2)\cdot a_{i_2} + ... + (n-t)\cdot a_{i_t} + ... + a_{i_{n-1}}] \tag{2}$$

若$i_1 = 1$，则显然有一个最优解优先执行任务1，若不然，设$i_t = 1$，交换i_1和i_t，得到一个新的解$i_t, i_2, ..., i_{t-1}, i_1, i_{t+1}, ..., i_n$，这个新解的平均响应时间$T'$为

$$T' = \frac{1}{n}\sum_{j=1}^{n} w_{i_j} = \frac{1}{n}\sum_{j=1}^{n-1} e_{i_j} = \frac{1}{n}[(n-1)\cdot a_{i_t} + (n-2)\cdot a_{i_2} + ... + (n-t)\cdot a_{i_1} + ... + a_{i_{n-1}}] \tag{3}$$

由于$a_{i_1} \geq a_{i_t}$，那么，$T - T' = (t-1)(a_{i_1} - a_{i_t}) \geq 0$。由于$T$是最小的，可知$T' = T$，新解$i_t, i_2, ..., i_{t-1}, i_1, i_{t+1}, ..., i_n$也是一个最优解，且$a_{i_t} = a_{i_1} = \min_{j=0\to n} a_{i_j}$。综上所述，该问题具有贪心选择性。

伪代码：

Algorithm 2 平均响应时间最小问题

Input: n个任务的时间 $a[1:n]$

Output: n个任务的调度 $A[1:n]$

n ← length(a)
create new array $A[1:n]$
for $i = 1 \to n$ **do**
    min ← inf
    for $j = 1 \to n$ **do**
        if $a[j] <$ min **then**
            min ← $a[j]$
            $A[i] = j$
        end if
        $a[A[i]] \leftarrow$ inf
    end for
end for
return A

以上算法是根据贪心思想编写，复杂度为$O(n^2)$，但观察后可以发现该问题等同于将任务按照执行时间大小进行递增排序，排序结果即为最优调度，因此复杂度可降为$O(n \log n)$。