算法分析与设计期末考试答案评分要点

一、填空题(选做5道，10分)

说明：选做的每道题目2分，和本答案等价的描述不扣分。若选做了多于5道的题目，则按照使答卷者得分最高的方式给分。

1 用矩阵幂的方法求斐波那契数，其运行时间为 ($\Theta(\lg(n))$).

2 对于一个可以用动态规划法求解的问题，要求问题既要满足 (最优子结构) 的特性，又要具有大量的 (重叠子问题).

3 对于一个可以用贪心法求解的问题，不仅要求问题满足 (最优子结构) 的特性，还应证明其贪心策略的 (正确性).

4 设有 n 个栈操作 (PUSH、POP、MULTIPOP) 的序列，作用于初始为空的栈S。不区分三种操作，则每个操作的最坏运行时间为 (O(n))，平摊运行时间为 (O(1)).

5 三种平摊分析的方法分别为 (聚集分析)、(记账法)、(势能法)。

6 四后问题的搜索空间为 (四叉) 树；0-1背包问题的搜索空间为 (子集) 树；巡回售货员问题的搜索空间为 (子集) 树。

7 (回溯法) 的求解目标是找出解空间树中满足约束条件的所有解，而 (分支限界法) 的求解目标则是找出满足约束条件的一个解，或是在满足约束条件的解中找出在某种意义下的最优解。

8 回溯法一般以 (深度) 优先的方式搜索解空间树，而分支限界法则一般以 (广度) 优先或以最小耗费优先的方式搜索解空间树。

二、(每小题1分，共10分)单项选择题

答案：CDDCA DDCDB

三、社会名流 (20分)

在n个人中，一个被所有人知道但却不知道别人的人，被定义为社会名流。现在的问题是如果存在，试找出该社会名流。你可以使用的唯一方式是询问：”请问你知道那个人吗？”请给出提问次数为 O(n) 的算法，写出伪代码，分析算法的正确性，并给出算法运行时间的精确分析 (即O(n)中隐藏的系数).

评分标准：

算法思想为，每次询问至少能使目标集合的势减一。

代码可以采多种组织方式，用栈的数据结构，或者采用分治的方法均可。

扣分的说明：

清晰的算法的描述10分。

伪代码描述不正确或者不清晰的酌情扣分，总分5分。

采用构造图的方式解决问题的答案，不满足题目的要求，不予给分。

对于判断过后的候选人未加验证而又未说明情况 (即未说明再有解的情况下) 酌情扣3-4分。

对于O(n) 的系数，一般不扣分。

四、地板覆盖 (20分))

用2*1的地板块覆盖3*n的地面有多少种方案？如下图 (图略) 是一个覆盖的例子，函数 fn 可用于求解这个问题，请说明 fn 算法的正确性，并说明算法运行时间的上界和下界。

```
int fn(int n~)~ {
    if (n % 2 == 1~)~ return 0;
    int [] f = new int[n+1];
    f[0] = 1;
    for (int i = 2; i <= n; i += 2~)~ {
        f[i] = f[i-2]*3;
        for (int j = i-4; j >= 0; j -= 2~)~
            f[i] += f[j]*2;
        }
  return f[n];
  }
```

评分说明：

fn 算法的正确性证明14分，需要按照题目所示的代码说明该算法的原理，并且论证题目中的解答正确。

未说明 n 是奇数的情况、未使用动态规划思想证明的酌情扣分。

fn 算法运行时间的上下界6分，需要对答案进行简要的计算和说明。

五、田忌赛马 (20分)

你一定听过田忌赛马的故事吧？如果3匹马变成n匹，齐王仍然让他的马按从优到劣的顺序出赛，田忌可以按任意顺序选择他的赛马出赛。赢一局，田忌可以得到200两银子，输一局，田忌就要输掉200两银子。已知国王和田忌的所有马的奔跑速度，并且所有马奔跑的速度均不相同，现已经对两人的马分别从快到慢排好序，请设计一个算法，帮助田忌赢得最多的银子。写出伪代码，证明算法的正确性，并分析算法的复杂度。

评分标准：

本题采用采用贪心算法解决，面对国王每匹顺序出场的马，如果田忌的马快，就派最快的出场；否则派最慢的马出场。

扣分说明：

伪代码部分8分，伪代码的算法非线性则酌情扣分。

贪心算法的正确性说明8分，方法不限，但需要严谨地说明贪心方法可以得到最优结果。

复杂度分析4分，没有考虑马已经排序的情况或者进行排序的酌情扣分

六、作业调度 (20分)

给出 n 项作业$J_1, J_2, \cdots, J_n$，对应每项作业有一个运行时间 t_i , 在 m 个处理器上调度这些作业，使完成的时间最小。完成的时间定义为在所有的处理器中运行时间最长的处理器运行的时间。采用如下的近似算法：即，按照原始给定的作业顺序:$J_1, J_2, \cdots, J_n$ ，把每一项作业分配给当前情况下最近可用的那个处理器，使该作业尽可能早被处理 (其它没有任何约束)

1.试证明该算法的近似度为$2 - \frac{1}{m}$。(10分)

2.构造边界情况，说明这个界是紧的。(10分)

参考答案：

易知：

$$OPT \geq \frac{1}{m} \sum t_i$$
$$OPT \geq max t_i$$

假定加载最后一个作业j的机器为M，其运行时间为L，那么在J_j被加载之前M的运行时间为$L - t_j$，且该时刻之前必无可用的机器，于是有:

$$\sum t_i \geq m(L - t_j) + t_j$$

故有：

$$OPT \geq \frac{1}{m}\sum t_i \geq L + \frac{t_j}{m}$$

变换得：

$$L \leq OPT + (1 - \frac{1}{m})t_j \leq (2 - \frac{1}{m})OPT$$

边界的情形如：$m(m-1)$个处理时间为1的作业，1个处理时间为m作业。

扣分说明：

近似度证明10分，证明不严谨的酌情扣分。

边界情况构造10分，要求该作业序列在用近似算法的情况下可以达到题目中的界。其他情况酌情扣分。