算法课第一次书面作业

作业得分（打印时请保留此项）：

	题目 1	题目 2	题目 3	题目 4	题目 5	总分
分数						
阅卷人						

注意事项：

1. 算法课作业均采用 A4 纸打印，**左上角装订**；
2. 不需要复制题目内容，直接在每一题的标题下填写解题过程即可；
3. 对于复杂公式或图形，可以留空白后手写完成，**文字部分应该打印**；
4. 注意填写页眉中的姓名、学号；
5. 打印时请保留第一页上方的“作业得分”表格。

题目 1：

船对于港口的偏好按照时间顺序越前的越偏好，港口对于船的偏好按照时间顺序越后的越偏好，在船和港口中间做一个稳定匹配算法。
由于有n条船和n个港口，且这个月每条船访问每个港口恰好一次，所以是一个完全连边的二分图。所以稳定匹配一定存在。下面说明找到的匹配是符合题意的，如果违反题意，那么港口对于后到港口的船的偏好应当高于已经停在港口的船，那么他不是稳定匹配。

题目 2：

（1）　第i次归并的代价为$(i+1)n/k$，一共需要进行$k-1$次归并。所以最坏的时间复杂度为：

$$(2+3+\cdots+k)*\frac{n}{k}=\frac{n(k+2)(k-1)}{k}=O(nk)$$

（2）　利用k个序列的第一个元素构建一个小根堆，每次取出堆中最小的元素，若它是来自表L_i且并不是表L_i的最后一个元素，那么把表的下一个元素压入堆中，直至堆为空。每得到一个元素的时间复杂度为$O(logk)$，一共有n个元素，所以总的时间复杂度为$O(nlogk)$。

题目 3：

（a）　对于$k=2$的情况，考虑如果能够丢m次，最多可以测试多高的阶梯。如果第一

次摔碎，之后一个瓶子还可以尝试$m-1$次，所以第一次应当在m阶楼梯丢下去，如果摔碎且$m-1$层为最高安全格挡的话，可以用一个瓶子测试出来。所以对于第一个杯子如果没有碎的话应当依次扔在$m, m+(m-1), m+(m-1)+(m-2), \ldots, m+\cdots+1$层上。能够测试的最高阶梯为$g(m)=\frac{m(m+1)}{2}$。所以$f$实际上应当为$g$的反函数，考虑到取整的问题$f(n)=\lceil\frac{\sqrt{1+8n}-1}{2}\rceil$，有$\lim_{n\to\infty} f(n)/n=0$。

放松一些可以第一个瓶子每一次扔$\sqrt{n}$节点处，第二个瓶子最多扔$\sqrt{n}$次，$f(n)=n^{1/2}$。

（b）　定义g_2为之前得到的g，那么应当有

$$g_{k+1}(m)=\sum_{i=1}^{m} g_k(m-i)+1$$

对任意层数n，一定有$g_k(m-1)<n\le g_k(m)$，按照k个瓶子丢m次的策略来丢。放松一些可以第一个瓶子扔$n^{(k-1)/k}$处，第i个瓶子以$n^{(k-i)/k}$在测出的区间上扔，此时$f(n)=n^{1/k}$。

题目 4：

利用并查集，最开始有$2n$个元素（从 0 开始标号），如果i,j是一类，那么对$(2i,2j),(2i+1,2j+1)$进行 merge 操作。如果i,j不是同一类，那么对于$(2i,2j+1),(2j,2i+1)$进行 merge 操作。之后对于$i=1,\ldots,n$检查$2i,2i+1$的 root 节点是否相同。如果存在相同则说明判断不矛盾，否则有矛盾。

初始化并查集$O(n)$，每次 merge 操作$O(1)$，总的 merge 操作复杂度$O(m)$，最后检查的复杂度$O(n)$，总的时间复杂度是$O(n+m)$的。

题目 5：

不成立，考虑一个n个节点的链，之后接m个顶点都与且仅与链的一个端点连边，且这$m+1$个顶点自己可以成团。那么可以知道无论m为多少，都有$diam(G)=n$。

$$apd(G)\le\frac{C_{n+m}^2+mn^2+n^3}{C_{n+m}^2}=1+\frac{(m+n)n^2}{C_{n+m}^2}=1+\frac{2n^2}{n+m-1}$$

当$m>2n^2-n+1$时，有$apd(G)<2$，所以不能存在正的自然数可以对所有连通图使得不等式成立。