算法课第 2 次作业

作业得分：

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

题目 1:

答：不能保证总存在稳定时间表。以下选择任务(b)来求解。

设：当前共有 n=2 个时段，A 网的节目列表 Pa={a0, a1}，B 网的时间表 Pb={b0, b1}，而对于这些电视节目而言，收视率关系有 a0 > b0 > a1 > b1。

以下分两种时间表的情况进行讨论：

（1）若：给出时间表(S, T) = {<a0,b0>, <a1,b1>}，则对于 B 网，目前能赢得的时段有 0 个，若 B 网单方面改变时间表为(S, T') = {<a0, b1>, <a1, b0>}，则可赢得 1 个时段，是一种更优的策略。

（2）若：给出时间表(S, T) = {<a0,b1>, <a1,b0>}，则对于 A 网，目前能赢得的时段有 1 个，若 B 网单方面改变时间表为(S, T') = {<a0, b1>, <a1, b0>}，则可赢得 2 个时段，是一种更优的策略。

综上所述，对每组节目和收视率，不能保证总存在稳定时间表。

题目 2:

答：1 小时=60 分钟=3600 秒=$3600 \times 10^{10} = 3.6 \times 10^{13}$次运算。

（a） $n^2 = 3.6 \times 10^{13} \rightarrow n = \sqrt{3.6 \times 10^{13}} \rightarrow n = 6 \times 10^6$

（b） $n^3 = 3.6 \times 10^{13} \rightarrow n = \sqrt[3]{3.6 \times 10^{13}} \rightarrow n = 33015$

（c） $100n^2 = 3.6 \times 10^{13} \rightarrow n^2 = 3.6 \times 10^{11} \rightarrow n = \sqrt{3.6 \times 10^{11}} \rightarrow n = 6 \times 10^5$

（d） $nlogn = 3.6 \times 10^{13} \rightarrow 2^{nlogn} = 3.6 \times 10^{13} \rightarrow n \times 2^n = 2^{3.6 \times 10^{13}}$

（e） $2^n = 3.6 \times 10^{13} \rightarrow n = \log(3.6 \times 10^{13}) \rightarrow n = 45$

（f） 由于2^{2^n}增长速度过快，以下采用枚举法进行计算。最终得到 n 不应超过 5。

表 1 运算结果

n	2^n	2^{2^n}
1	2	4
2	4	16
3	8	256
4	16	65536
5	32	4294967296
6	64	18446744073709551616L

以下展示部分关键运算结果截图（本计算由 python2.7 实现）：

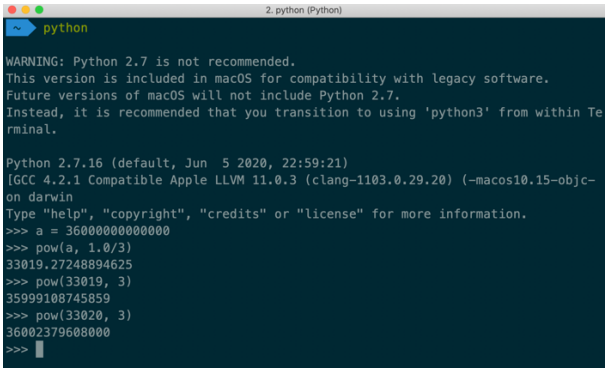

图 1 （b）运算结果

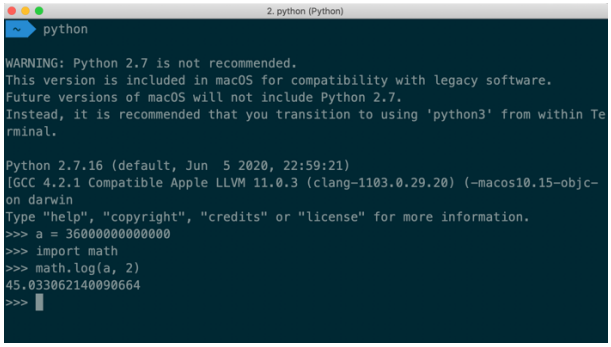

图 2 （e）运算结果

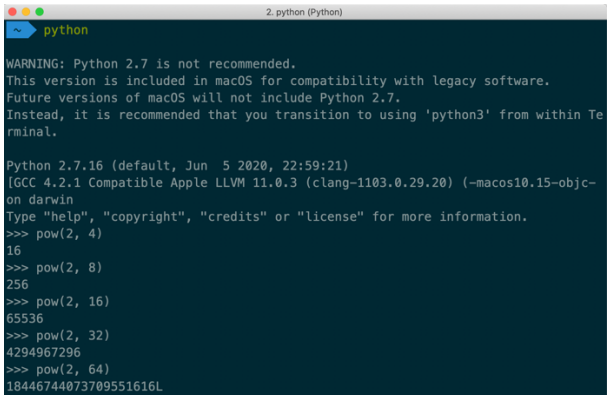

图 3 （f）运算结果

题目 3:

答：

1.显然$n < 2^n$故有$2^n < 2^{2^n}$。

2. $\lim\limits_{n\to\infty}\frac{2^n}{n^{logn}} = \lim\limits_{n\to\infty} 2^{log\frac{2^n}{n^{logn}}} = \lim\limits_{n\to\infty} 2^{log2^n - logn^{logn}} = \lim\limits_{n\to\infty} 2^{log2^n - logn^{logn}} = \lim\limits_{n\to\infty} 2^{n-(logn)^2} \infty$。故有$n^{logn} < 2^n$。

3.显然常数小于$logn$，故有$n^{\frac{4}{3}} < n^{logn}$。

4. $\lim\limits_{n\to\infty}\frac{n(logn)^3}{n^{\frac{4}{3}}} = \lim\limits_{n\to\infty}\frac{(logn)^3}{n^{\frac{1}{3}}} = \lim\limits_{n\to\infty}\frac{logn}{n^{\frac{1}{9}}} = 0$，故有$n(logn)^3 < n^{\frac{4}{3}}$。

5.由于$2^{\sqrt{logn}} < 2^{logn} = n < n(logn)^3$，故有$2^{\sqrt{logn}} < n(logn)^3$。

综上所述：按照增长率递增的顺序排列函数如下：

$$2^{\sqrt{logn}} < n(logn)^3 < n^{\frac{4}{3}} < n^{logn} < 2^n < 2^{2^n}$$

题目 4:

(a) 由于要求在小于线性复杂度的时间内完成计算，因此朴素的二分查找+顺序遍历的思想就不符要求，因为这种方法仅仅是将$O(n)$变成了$O\left(\frac{1}{2^k}n\right)$其中$\frac{1}{2^k}$仍是一个常数，即还是一个$O(n)$复杂度的计算问题。

为了降低计算复杂度，可以尝试采用开方的形式进行求解，即第一个瓶子的尝试位置序列可以表示为：$i\lfloor\sqrt{n}\rfloor, i \in [1, \lfloor\sqrt{n}\rfloor]$，假设第一个瓶子在掷出第$k$次后碎掉了，那么第二个瓶子将从第$(k-1)\lfloor\sqrt{n}\rfloor + 1$个位置开始尝试。

对于上述方法，第一个瓶子最多尝试$\lfloor\sqrt{n}\rfloor$次，第二个瓶子同样最多尝试$\lfloor\sqrt{n}\rfloor$次，整体时间复杂度为$O(2\lfloor\sqrt{n}\rfloor) = O(\sqrt{n}) < O(n)$。

(b) 受到(a)的启发，我们在掷第k个瓶子时，可以将下次查找的区间缩小至$n^{\frac{k-1}{k}}$的范围内，即对于第k个瓶子，我们的尝试位置序列可以表示为：$i\left\lfloor n^{\frac{k-1}{k}}\right\rfloor, i \in \left[1, \left\lfloor n^{\frac{1}{k}}\right\rfloor\right]$，该步最多尝试了$\left\lfloor n^{\frac{1}{k}}\right\rfloor$次。

接着，对于剩下的$k-1$个瓶子，我们的尝试位置序列可以表示为：$i\left\lfloor n^{\frac{k-2}{k-1}}\right\rfloor, i \in \left[1, \left\lfloor n^{\frac{1}{k-1}}\right\rfloor\right]$。

递归进行上述步骤直至剩下 2 个瓶子，该问题回到了问题(a)，对k个瓶子的尝试次数进行累加得总尝试次数为$(k-1)\lfloor\sqrt[k]{n}\rfloor$，即使用该方法的计算复杂度为$O\left(k\sqrt[k]{n}\right)$。

对于$f_k(n) = k\sqrt[k]{n}$而言，有：$\lim_{n\to\infty}\frac{f_k(n)}{f_{k-1}(n)} = 0$。符合题目要求。

题目 5:

答：命题成立。

证明：采用反证法，假设满足要求的图G不是联通的，则图G必然有n个$(n \geq 2)$联通子图。对于这n个子图，必有其中一个子图S的节点数$|S| \leq \frac{n}{2}$。而对于S中的任一节点v，其度数之最大值$d_{max} \leq \frac{n}{2} - 1 < \frac{n}{2}$。这与我们的前提图$G$的每个节点度数至少为$\frac{n}{2}$是矛盾的。

故假设不成立，原命题成立。

此命题证毕。

郑重声明

本作业由作者独立完成。抄袭行为在任何情况下都是不能容忍的(COPY is strictly prohibited under any circumstances)！由抄袭所产生的一切后果由抄袭者承担，勿谓言之不预也。

陈麒先