算法分析第1次作业

小组编号：23

本次作业负责人:黄勖

1-1题 答案:

1-1 求下列函数的渐近表达式：

$3n^2+10n$； $n^2/10+2^n$； $21+1/n$； $\log n^3$； $10\log 3^n$

1-1

(1) $O(n^2)$ (2) $O(2^n)$ (3) $O(1)$ (4) $O(\log n)$ (5) $O(n)$

本题分工：黄勖

1-2题 答案:

1-2 试论 $O(1)$和 $O(2)$的区别。

1-2

渐近上界记号O，若$T(n)=O(g(n))$则$T(n)$的阶小于等于$g(n)$的阶

根据O定义 $O(1)=O(2)$ 均为常数阶

用$O(1)$或$O(2)$表示同函数时，仅是常数因子不同的区别

本题分工：李嘉琪

1-3题 答案:

1-3 按照渐近阶从低到高的顺序排列以下表达式：$4n^2$，$\log n$，3^n，$20n$，2，$n^{2/3}$。又 $n!$ 应该排在哪一位？

1-3

2, $\log n$, $n^{\frac{2}{3}}$, $20n$, $4n^2$, 3^n, $n!$

本题分工：黄勖

1-4题 答案:

1-4 （1）假设某算法在输入规模为 n 时的计算时间为 $T(n)=3\times2^n$。在某台计算机上实现并完成该算法的时间为 t 秒。现有另一台计算机，其运行速度为第一台的 64 倍，那么在这台新机器上用同一算法在 t 秒内能解输入规模为多大的问题？

（2）若上述算法的计算时间改进为 $T(n)=n^2$，其余条件不变，则在新机器上用 t 秒时间能解输入规模为多大的问题？

（3）若上述算法的计算时间进一步改进为 $T(n)=8$，其余条件不变，那么在新机器上用 t 秒时间能解输入规模为多大的问题？

1-4

(1) 设新机器使用同算法在 t 秒内解输入规模为 x 的问题

原机器在输入规模 n 时计算时间 $T(n)=3\times2^n$，用时 t

新机器运行速度为原机器的64倍，故

新机器在输入规模 n 时计算时间 $T(n)=3\times2^n$，用时 $\frac{1}{64}t$

得 $\frac{3\times2^n}{\frac{1}{64}t}=\frac{3\times2^x}{t} \Rightarrow 2^x=2^6\times2^n \Rightarrow x=n+6$

故解决输入规模为 n+6 的问题

(2) 同理可得，设输入规模为 y

$\frac{n^2}{\frac{1}{64}t}=\frac{y^2}{t} \Rightarrow y=8n$

故新机器用 ts 解决输入规模为 8n 的问题

(3) 若算法计算时间 $T(n)=8$，此时 $T(n)=O(1)$，用时与规模无关

无论问题规模多少都可在常数时间解决，

故可解决任意规模的问题

本题分工：李嘉琪

1-5题 答案:

1-5 硬件厂商 XYZ 公司宣称他们最新研制的微处理器运行速度为其竞争对手 ABC 公司同类产品的 100 倍。对于计算复杂性分别为 n、n^2、n^3 和 $n!$ 的各算法，若用 ABC 公司的计算机在 1 小时内能解输入规模为 n 的问题，那么用 XYZ 公司的计算机在 1 小时内分别能解输入规模为多大的问题？

1-5

① $t=n=\frac{n'}{100}$　　$n'=100n$　可解决输入规模为100n的问题

② $t=n^2=\frac{n'^2}{100}$　　$n'=10n$　可解决输入规模为10n的问题

③ $t=n^3=\frac{n'^3}{100}$　　$n'=100^{\frac{1}{3}}n$　可解决输入规模为 $\sqrt[3]{100}n$ 的问题

④ $t=n!=\frac{n'!}{100}$　　由 $n'!=100n!$　　得 $n'-n<\log_2 100\approx 6.64$

$n'(n'-1)(n'-2)\cdots(n+1)=100$

$n^{(n'-n)}<100$　　$n'<n+6.64$

$2^{(n'-n)}<100$　　可解决输入规模小于n+6.64的问题

本题分工：黄勖

1-6题 答案:

1-6　对于下列各组函数 $f(n)$ 和 $g(n)$，确定 $f(n)=O(g(n))$ 或 $f(n)=\Omega(g(n))$ 或 $f(n)=\theta(g(n))$，并简述理由。

（1）$f(n)=\log n^2$；　$g(n)=\log n+5$　　（5）$f(n)=10$；　$g(n)=\log 10$

（2）$f(n)=\log n^2$；　$g(n)=\sqrt{n}$　　（6）$f(n)=\log^2 n$；　$g(n)=\log n$

（3）$f(n)=n$；　$g(n)=\log^2 n$　　（7）$f(n)=2^n$；　$g(n)=100n^2$

（4）$f(n)=n\log n+n$；　$g(n)=\log n$　　（8）$f(n)=2^n$；　$g(n)=3^n$

1-6

(1) $f(n)=\log n^2$, $g(n)=\log n+5$

$$\lim_{n\to\infty}\frac{f(n)}{g(n)}=\lim_{n\to\infty}\frac{\log n^2}{\log n+5}=\lim_{n\to\infty}\frac{2(\log n+5)-10}{\log n+5}=\lim_{n\to\infty}2-\frac{10}{\log n+5}=2$$

$\therefore f(n)=\Theta(g(n))$ 即 $\log n^2=\Theta(\log n+5)$

(2) $f(n)=\log n^2$, $g(n)=\sqrt{n}$

$$\lim_{n\to\infty}\frac{f(n)}{g(n)}=\lim_{n\to\infty}\frac{\log n^2}{\sqrt{n}}=\lim_{n\to\infty}\frac{2\log n}{\sqrt{n}}=\lim_{n\to\infty}\frac{\frac{2}{\ln 2}\cdot\frac{1}{n}}{\frac{1}{2\sqrt{n}}}=\lim_{n\to\infty}\frac{4}{\ln 2\sqrt{n}}=0$$

$\therefore f(n)=O(g(n))$ 即 $\log n^2=O(\sqrt{n})$

(3) $f(n)=n$ $g(n)=\log^2 n$

$$\lim_{n\to\infty}\frac{f(n)}{g(n)}=\lim_{n\to\infty}\frac{n}{\log^2 n}=\lim_{n\to\infty}\frac{1}{2\log n\times\frac{1}{n\ln 2}}=\cancel{\frac{\ln 2}{2}}\lim_{n\to\infty}\frac{\ln 2}{2}\times\frac{n}{\log n}$$

$$=\lim_{n\to\infty}\frac{\ln 2}{2}\times\frac{1}{\frac{1}{\ln 2}\frac{1}{n}}=\lim_{n\to\infty}\frac{\ln^2 2}{2}\cdot n=\infty$$

$\therefore f(n)=\Omega(g(n))$ 即 $n=\Omega(\log^2 n)$

(4) $f(n) = n\log n + n \quad g(n) = \log n$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{n\log n + n}{\log n} = \lim_{n\to\infty} n \cdot \frac{\log n + 1}{\log n} = \infty$$

$\therefore f(n) = \Omega(g(n))$ 即 $n\log n + n = \Omega(\log n)$

(5) $f(n) = 10$, $g(n) = \log 10$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{10}{\log 10} = \frac{10}{\log 10}$$

$\therefore f(n) = \Theta(g(n))$ 即 $10 = \Theta(\log 10)$

(6) $f(n) = \log^2 n \quad g(n) = \log n$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{\log^2 n}{\log n} = \lim_{n\to\infty} \log n = \infty$$

$\therefore f(n) = \Omega(g(n))$ 即 $\log^2 n = \Omega(\log n)$

(7) $f(n) = 2^n \quad g(n) = 100n^2$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{2^n}{100n^2} = \lim_{n\to\infty} \frac{2^n \ln 2}{200n} = \lim_{n\to\infty} \frac{\ln^2 2}{2} \cdot 2^n = \infty$$

$\therefore f(n) = \Omega(g(n))$ 即 ~~f(n)~~ $2^n = \Omega(100n^2)$

(8) $f(n) = 2^n \quad g(n) = 3^n$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{2^n}{3^n} = \lim_{n\to\infty} \left(\frac{2}{3}\right)^n = 0$$

$\therefore f(n) = O(g(n))$ 即 $2^n = O(3^n)$

本题分工：李嘉琪

1-7题 答案:

1-7　证明 $n!=o(n^n)$。

1-7

即求对 $\forall \varepsilon>0$, $\exists N_0$ 当 $N \geq N_0$ 且 $\frac{n!}{n^n} \leq \varepsilon$

由 $\frac{1}{n^n} < \frac{n!}{n^n} < \frac{1}{n}$ ($N_0>1$时), 及 $\lim\limits_{n\to\infty} \frac{1}{n^n}=0$ $\lim\limits_{n\to\infty} \frac{1}{n}=0$

得 $\lim\limits_{n\to\infty} \frac{n!}{n^n}=0$

$\therefore$ $\forall \varepsilon>0$ $\exists N_0$ 当 $N \geq N_0$, $\frac{n!}{n^n} \leq \varepsilon$

本题分工：黄勖

1-8题 答案:

1-8　下面的算法段用于确定 n 的初始值。试分析该算法段所需计算时间的上界和下界。

```
while(n>1)
    if(odd(n))
        n = 3*n+1;
    else
        n = n/2;
```

1-8

当①考虑时间下界，给定n为正(n>0)偶数时，具有渐近下界：

假设经过k次循环 n=1 退出，由else中 n=n/2 得

$$\frac{n}{2^k} = 1$$

$\therefore$ $k = \log n$

$\therefore$ 时间下界为 $\Omega(\log n)$

②考虑时间上界，无法确定一个时间上界对所有正整数均成立

本题分工：李嘉琪

1-9题 答案:

1-9　证明：如果一个算法在平均情况下的计算时间复杂性为 $\theta(f(n))$，则该算法在最坏情况下所需的计算时间为 $\Omega(f(n))$。

1-9

$$T_{avg}(N) = \sum_{Z \in D_N} P(Z) T(N, Z)$$

$$\leq \sum_{Z \in N} P(Z) \max_{Z' \in D_N} T(N, Z')$$

$$= T(N, Z') \sum_{Z \in D_N} P(Z)$$

$$= T(N, Z^*) = T_{max}(N)$$

$$T_{max}(N) = \Omega(T_{avg}(N)) = \Omega(\Theta(f(n))) = \Omega(f(n))$$

本题分工：黄勖