算法作业答案

作业一：

1（3.1-1） 假设$f(n)$与$g(n)$都是渐近非负函数。使用θ记号的基本定义来证明$\max\big(f(n), g(n)\big) = \theta\big(f(n)+g(n)\big)$。

证明：因为$f(n)$与$g(n)$都是渐近非负函数

根据定义有：存在N_f，N_g使得：当$n \ge N_f$时，$f(n) \ge 0$，同时，当$n \ge N_g$时，$g(n) \ge 0$。

所以，我们取$N_0 = max\big(N_f, N_g\big)$，此时，当$n \ge N_0$时，同时有$f(n) \ge 0$，$g(n) \ge 0$。

取$C_1 = 1/2$，$C_2 = 1$，则当$n \ge N_0$时，有：

$\big(f(n)+g(n)\big)/2 \le \max\big(f(n), g(n)\big) \le f(n)+g(n)$

得证

2（3.1-4） $2^{n+1} = O(2^n)$成立吗？$2^{2n} = O(2^n)$成立吗？

解：1）$2^{n+1} = O(2^n)$成立

取$\mathrm{c} \ge 2$，当$\mathrm{n} \ge 1$时，

有$2^{n+1} = 2 * 2^n \le c * 2^n$

因此，成立

2）$2^{2n} = O(2^n)$不成立

如果$2^{2n} = O(2^n)$成立，根据定义有：$2^{2n} \le c2^n$（c为常数）

得，$n \le \lg c$，则$2^{2n} \le c2^n$对于任意大的n不成立，与假设矛盾

因此，$2^{2n} = O(2^n)$不成立

3 求下列函数的渐近表达式：

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

解：$3n^2 + 10n = O(n^2)$

$n^2/10 + 2^n = O(2^n)$

$21 + 1/n = O(1)$

$\log n^3 = O(\log n)$

$10\log 3^n = O(n)$

4 试讨论 $O(1)$与$O(2)$的区别

解：根据符号O的定义易知$O(1) = O(2)$。用$O(1)$或$O(2)$表示用一个函数时，差别仅在于其中的常数因子。

5 （1）假设某算法在输入规模上为n时的计算时间为$T(n) = 3 * 2^n$。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 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？

解：（1）设新机器用同一算法在t秒内能解输入规模为n_1的问题。因此有，$t = 3 * 2^n = 3 *$

$2^{n_1}/64$，解得$n_1 = n + 6$。

（2）同（1），$t = n^2 = n_1^2/64$，解得，$n_1 = 8n$。

（3）由于$T(n) = 8$为常数阶，因此算法可解任意规模的问题。

6 对于下列各组函数$f(n)$和$g(n)$，确定$f(n) = O\big(g(n)\big)$或$f(n) = \Omega\big(g(n)\big)$或$f(n) = \theta\big(g(n)\big)$，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）$f(n) = 2^{\mathrm{n}}$；$g(n) = 100\mathrm{n}^2$

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

解：（1）$\log n^2 = \theta(\log n + 5)$ （5）$10 = \theta(\log 10)$

（2）$\log n^2 = O\big(\sqrt{n}\big)$ （6）$\log^2 n = \Omega(\log n)$

（3）$n = \Omega(\log^2 n)$ （7）$2^{\mathrm{n}} = \Omega(100\mathrm{n}^2)$

（4）$n\log n + n = \Omega(\log n)$ （8）$2^{\mathrm{n}} = O(3^{\mathrm{n}})$

7 证明：$n! = \mathrm{o}(n^n)$

证明：由 stirling 公式得，$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^{\mathrm{n}}\left[1 + \theta\left(\frac{1}{n}\right)\right]$

$$\lim_{n\to\infty} n!/n^n = \lim_{n\to\infty}\frac{\sqrt{2\pi n}\left[1+\theta\left(\frac{1}{n}\right)\right]}{e^n} = 0$$

所以，$n! = \mathrm{o}(n^n)$。

证毕

8 证明：当$a_k > 0$时，任何多项式$P(n) = a_k n^k + a_{k-1}n^{k-1} + \cdots + a_0$属于集合$\theta(n^k)$。

证明：要证，$P(n)$属于集合$\theta(n^k)$

只需证，存在N_0，C_1，C_2，使得：当$n \geq N_0$时，有$0 \leq C_1 n^k \leq P(n) \leq C_2 n^k$。

1）
$$\begin{aligned} P(n) &= a_k n^k + a_{k-1}n^{k-1} + \cdots + a_0 \\ &\geq a_k n^k - (|a_{k-1}| + \cdots + |a_0|)n^{k-1} \\ &\geq \left(a_k - \frac{(|a_{k-1}|+\cdots+|a_0|)}{n}\right)n^k \end{aligned}$$

令$a_k - \frac{(|a_{k-1}|+\cdots+|a_0|)}{n} \geq 0$，得$n \geq \frac{(|a_{k-1}|+\cdots+|a_0|)}{a_k}$

取$N_1 = \frac{(|a_{k-1}|+\cdots+|a_0|)}{a_k} + 1$，$C_1 = a_k - \frac{(|a_{k-1}|+\cdots+|a_0|)}{N_1} > 0$，当$n \geq N_1$时，有

$$0 \leq C_1 n^k \leq P(n)$$

2）
$$\begin{aligned} P(n) &= a_k n^k + a_{k-1}n^{k-1} + \cdots + a_0 \\ &\leq (|a_k| + |a_{k-1}| + \cdots + |a_0|)n^k \end{aligned}$$

取$N_2 = 1$，$C_2 = |a_k| + |a_{k-1}| + \cdots + |a_0|$，当$n \geq N_2$时，有

$$P(n) \leq C_2 n^k$$

综上 1），2），取$N_0 = \max(N_1, N_2)$，当$n \geq N_0$时，有

$$0 \leq C_1 n^k \leq P(n) \leq C_2 n^k$$

所以，当$a_k > 0$时，任何多项式$P(n) = a_k n^k + a_{k-1}n^{k-1} + \cdots + a_0$属于集合$\theta(n^k)$。

证毕

作业 3：

1（4.3-1）证明：$T(n)=T(n-1)+n$的解为$O(n^2)$。

证明：假设对$\forall m<n$，$\exists c>0$，使得：

$$T(m)\le \mathrm{cm}^2$$

则有：

$$T(n-1)\le c(n-1)^2$$

带入迭代式可得：

$$\begin{aligned}T(n)&\le c(n-1)^2+n\\&=cn^2-2cn+c+n\end{aligned}$$

令$-2cn+c+n\le 0$，可得：

$$c\ge\frac{n}{2n+1}$$

故对于$\forall n>0$，可令$c=1$，使得

$$T(n)\le cn^2-2cn+c+n\le cn^2$$

故$T(n)=O(n^2)$。

2（4.3-6） 证明：$T(n)=2T(\lfloor n/2\rfloor+17)+n$的解为$\mathrm{O}(n\lg n)$。

往证：存在常数N_0，C_0，使得：当$n\ge N_0$时，$T(n)\le C_0 n\lg n$。

假设：$T(\lfloor n/2\rfloor+17)\le C_0(\lfloor n/2\rfloor+17)\lg(\lfloor n/2\rfloor+17)$

有：$T(n)\le 2C_0(\lfloor n/2\rfloor+17)\lg(\lfloor n/2\rfloor+17)+\mathrm{n}$

$\le 2C_0(n/2+17)\lg(n/2+17)+\mathrm{n}$

$=C_0(n+34)[\lg(n+34)-1]+\mathrm{n}$

$=C_0 n\lg(n+34)-C_0 n+34C_0\lg(n+34)-34C_0+\mathrm{n}$ (*1)

取$C_0\ge 2$

则有(*1) $\le C_0 n\lg(n+34)+34C_0\lg(n+34)-34C_0-\mathrm{n}$

$<C_0 n\lg(n+34)+34C_0\lg(n+34)-\mathrm{n}$

$=C_0 n\lg n+C_0 n\lg(n+34)-C_0 n\lg n+34C_0\lg(n+34)-\mathrm{n}$ (*2)

下面我们只需证明存在常数N_0，$C_0\ge 2$，使得当$n\ge N_0$时，

有 $C_0 n\lg(n+34)-C_0 n\lg n+34C_0\lg(n+34)-\mathrm{n}\le 0$

即有： (*2) $\le C_0 n\lg n$

$C_0 n\lg(n+34)-C_0 n\lg n+34C_0\lg(n+34)-\mathrm{n}$

$=C_0 n(\lg(n+34)-\lg n)+34C_0\lg(n+34)-\mathrm{n}$

$=C_0 n(\ln(1+34/n)/\ln 2)+34C_0\lg(n+34)-\mathrm{n}$

$\le C_0 n(34/(n\ln 2))+34C_0\lg(n+34)-\mathrm{n}$

此时只需取$C_0=3$，$N_0=100$，即可使当$n\le N_0$时，有

$C_0 n\lg(n+34)-C_0 n\lg n+34C_0\lg(n+34)-\mathrm{n}\le 0$

原题得证

3（4.4-4）对递归式$T(n)=T(n-1)+1$，利用递归树确定一个好的渐进上界，用代入法进行验证。

解：递归树如下，树高为$n-1$，叶节点数目为 1，整棵树的代价为：

$$T(n)=1+1+1+\cdots 1+\theta(1)=n-1+\theta(1)=O(n)$$

代入法验证：

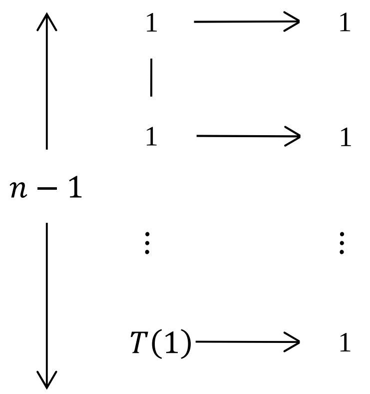
令对于$\forall m<n$，$\exists c>0$，使$T(m)\le cm$，有

$$T(n) = T(n-1) + 1$$
$$\leq c(n-1) + 1$$
$$= cn - c + 1$$

故对于$\forall n > 0$，可令$c = 1$，使得

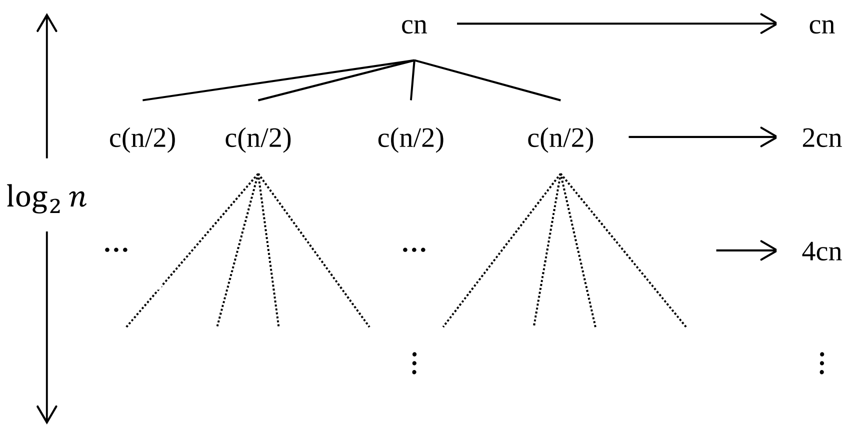
$$T(n) \leq cn - c + 1 \leq cn$$

所以，$T(n) = O(n)$。



4（4.4-7） 对递归式$T(n) = 4T(\lfloor n/2 \rfloor) + cn$（$c$为常数），画出递归树，并给出其解的一个渐近紧缺界。用代入法验证。

证明：



递归树的高度为$\log_2 n$，非叶子节点的度为 4，每一层的贡献为$4^i \lfloor cn/2^i \rfloor$。

则 $$T(n) = 4T(\lfloor n/2 \rfloor) + cn$$

$$= \sum_{i=0}^{\log_2 n} 4^i \lfloor cn/2^i \rfloor$$

1) $$T(n) \leq \sum_{i=0}^{\log_2 n} 4^i cn/2^i$$

$$= cn \sum_{i=0}^{\log_2 n} 2^i$$

$$= cn \frac{2^{\log_2 n + 1} - 1}{2 - 1}$$

$$= O(n^2)$$

2) $$T(n) \ge \sum_{i=0}^{\log_2 n} 4^i\left(cn/2^i - 1\right)$$

$$= cn\sum_{i=0}^{\log_2 n} 2^i - \sum_{i=0}^{\log_2 n} 4^i$$

$$= cn\frac{2^{\log_2 n+1}-1}{2-1} - \frac{4^{\log_2 n+1}-1}{4-1}$$
$$= 2cn^2 - cn - 4/3n^2 + 1/3$$
$$= (2c - 4/3)n^2 - cn + 1/3$$
$$= \Omega(n^2)$$

综上 1)、2)，得$T(n) = \theta(n^2)$

用代入法验证

a) 令$T(\lfloor n/2 \rfloor) \le c\lfloor n/2 \rfloor^2 - c\lfloor n/2 \rfloor$

有
$$\begin{aligned} T(n) &= 4T(\lfloor n/2 \rfloor) + cn \\ &\le 4(c\lfloor n/2 \rfloor^2 - c\lfloor n/2 \rfloor) + cn \\ &< 4c(n/2)^2 - 4c(n/2) + cn \\ &= cn^2 - 2cn + cn \\ &= cn^2 - cn \end{aligned}$$

b) 令$T(\lfloor n/2 \rfloor) \ge c\lfloor n/2 \rfloor^2 + 3c\lfloor n/2 \rfloor + 3c$

有
$$\begin{aligned} T(n) &= 4T(\lfloor n/2 \rfloor) + cn \\ &\ge 4(c\lfloor n/2 \rfloor^2 + 3c\lfloor n/2 \rfloor + c) + cn \\ &> 4c(n/2 - 1)^2 + 12c(n/2 - 1) + 4c + cn \\ &= cn^2 - 4cn + 4c + 6cn - 12c + 12c + cn \\ &= cn^2 + 3cn + 4c \\ &> cn^2 + 3cn + 3c \end{aligned}$$

5（4.2-1） 使用 Strassen 算法计算如下矩阵乘法：

$$\begin{bmatrix} 1 & 3 \\ 7 & 5 \end{bmatrix}\begin{bmatrix} 6 & 8 \\ 4 & 2 \end{bmatrix}$$

给出计算过程。

解： $A_{11} = 1$，$A_{12} = 3$，$A_{21} = 7$，$A_{22} = 5$

$B_{11} = 6$，$B_{12} = 8$，$B_{21} = 4$，$B_{22} = 2$

$S_1 = B_{12} - B_{22} = 8 - 2 = 6$

$S_2 = A_{11} + A_{12} = 1 + 3 = 4$

$S_3 = A_{21} + A_{22} = 7 + 5 = 12$

$S_4 = B_{21} - B_{11} = 4 - 6 = -2$

$S_5 = A_{11} + A_{22} = 1 + 5 = 6$

$S_6 = B_{11} + B_{22} = 6 + 2 = 8$

$S_7 = A_{12} - A_{22} = 3 - 5 = -2$

$S_8 = B_{21} + B_{22} = 4 + 2 = 6$

$S_9 = A_{11} - A_{21} = 1 - 7 = -6$

$S_{10} = B_{11} + B_{12} = 6 + 8 = 14$

$P_1 = A_{11}S_1 = 1 * 6 = 6$

$P_2 = S_2B_{22} = 4 * 2 = 8$