1、证明：对于任意正整数d和任意常数a_d >0，有：$P(n) = \Sigma_{i=0}^{d} a_i n^i = \theta(n^d)$.

证明：对于$0 \le i \le d-1$，可以通过n的增长使得$|a_i n^i| \le \frac{a_d n^d}{d+1}$，这要求$n \ge \sqrt[d-i]{\frac{(d+1)|a_i|}{a_d}}$，此时有$-\frac{a_d n^d}{d+1} \le a_i n^i \le \frac{a_d n^d}{d+1}$。于是，当$n \ge \max_{i=0,1,\dots,d-1}\left\{\sqrt[d-i]{\frac{|a_i|}{a_d}}\right\}$时，$a_d n^d - \frac{d a_d n^d}{d+1} \le \Sigma_{i=0}^{d} a_i n^i \le a_d n^d + \frac{d a_d n^d}{d+1}$。因此，当$c_1 = \frac{a_d}{d+1}$，$c_2 = \frac{(2d+1)a_d}{d+1}$，$n \ge \max_{i=0,1,\dots,d-1}\left\{\sqrt[d-i]{\frac{(d+1)|a_i|}{a_d}}\right\}$时，$c_1 n^d \le P(n) \le c_2 n^d$，得证。

2、证明：（1）$O\big(f(n)\big) + O\big(g(n)\big) = O(f(n) + g(n))$;

（2）$O\big(f(n)\big) \times O\big(g(n)\big) = O(f(n) \times g(n))$.

证明:（1）对于任意$p(n) = O(f(n))$，存在$c_p > 0$，当$n \ge n_p$时，$p(n) \le c_p f(n)$；对于任意$q(n) = O\big(g(n)\big)$，存在$c_q > 0$，当$n \ge n_q$时，$q(n) \le c_q g(n)$。令$c = \max\{c_p, c_q\} > 0$，当 $n \ge n_0 = \max\{n_p, n_q\}$ 时，有 $p(n) + q(n) \le c_p f(n) + c_q g(n) \le c(f(n) + g(n))$，得证。

（2）对于任意$p(n) = O(f(n))$，存在$c_p > 0$，当$n \ge n_p$时，$p(n) \le c_p f(n)$；对于任意$q(n) = O\big(g(n)\big)$，存在$c_q > 0$，当$n \ge n_q$时，$q(n) \le c_q g(n)$。令$c = c_p \times c_q > 0$，当$n \ge n_0 = \max\{n_p, n_q\}$时，有$p(n) \times q(n) \le c_p f(n) \times c_q g(n) = c(f(n) \times g(n))$，得证。

3、证明：$\sum_{j=0}^{\lfloor \log_b n \rfloor} (\log_b n - j)^k = \Omega((\log_b n)^{k+1})$，$k$为大于 0 的常数.

证明：$\sum_{j=0}^{\lfloor \log_b n \rfloor} (\log_b n - j)^k = (\log_b n)^k + (\log_b n - 1)^k + (\log_b n - 2)^k + \cdots + \left(\log_b n - \left\lfloor \frac{1}{2}\log_b n \right\rfloor\right)^k + \left(\log_b n - \left\lfloor \frac{1}{2}\log_b n \right\rfloor - 1\right)^k + \cdots + (\log_b n - \lfloor \log_b n \rfloor)^k \ge (\log_b n)^k + (\log_b n - 1)^k + (\log_b n - 2)^k + \cdots + \left(\log_b n - \left\lfloor \frac{1}{2}\log_b n \right\rfloor\right)^k \ge \left(\log_b n - \left\lfloor \frac{1}{2}\log_b n \right\rfloor\right)^k \times \left(1 + \left\lfloor \frac{1}{2}\log_b n \right\rfloor\right)$

$\ge \left(\frac{1}{2}\log_b n\right)^k \frac{1}{2}\log_b n = (\frac{1}{2})^{k+1} (\log_b n)^{k+1}$。

即当$c = (\frac{1}{2})^{k+1}$为正常数时，有$\sum_{j=0}^{\lfloor \log_b n \rfloor} (\log_b n - j)^k \ge c(\log_b n)^{k+1}$，得证。

4、解递归方程：$T(n)=T(an)+T(bn)+n$，其中$a,b>0$，$a+b=1$.

解：猜测$T(n)=O(n\log n)$，即当n足够大时，存在常数c使得$T(n)\le cn\log n$。

$$\begin{aligned}T(n)&=T(an)+T(bn)+n\\&\le can\log an+cbn\log bn+n\\&=can\log n-can\log\frac{1}{a}+cbn\log n-cbn\log\frac{1}{b}+n\\&=cn\log n-\left(c\left(a\log\frac{1}{a}+b\log\frac{1}{b}\right)-1\right)n\end{aligned}$$

当$c\ge\frac{1}{a\log\frac{1}{a}+b\log\frac{1}{b}}$时，有$T(n)\le cn\log n$，因此$T(n)=O(n\log n)$。

用类似的过程可以证明$T(n)=\Omega(n\log n)$。

5、解递归方程：$T(n)=T(an)+T(bn)+n$，其中$a,b>0$，$a+b<1$.

解：猜测$T(n)=O(n)$，即当n足够大时，存在常数c使得$T(n)\le cn$。

$$\begin{aligned}T(n)&=T(an)+T(bn)+n\\&\le can+cbn+n\\&=(c(a+b)+1)n\end{aligned}$$

当$c\ge\frac{1}{1-a-b}$时，有$T(n)\le cn$，因此$T(n)=O(n)$。

从$T(n)$的定义可知$T(n)=\Omega(n)$。

6、用 Master 方法求解：$T(n)=4T\left(\frac{n}{2}\right)+\theta(n\log n)$.

解：$a=4$，$b=2$，$\log_b a=2$，$n^{\log_b a}=n^2$，$f(n)=n\log n=O(n^{2-\varepsilon})$，其中$\varepsilon<1$即可。因此，$T(n)=\theta\left(n^{\log_b a}\right)=\theta(n^2)$。