Each problem is restated and followed by my solution to the problem, indicated by the "$\hookrightarrow$" symbol. For multi-part problems, my solutions are placed directly following the statement of the relevant part and preceeding the statement of the next part.

1. Consider the function $T(n)$ defined by the recurrence formula

$$T(n) = \begin{cases} 6 & 1 \leq n < 3 \\ 2T(\lfloor n/3 \rfloor) + n & n \geq 3 \end{cases}$$

   a. Use the iteration method to write a summation formula for $T(n)$.
   $\hookrightarrow$ We begin the iteration method by attempting to compute $T(n)$ for some n that satisfies the recurrence condition. In this case, we assume $n \geq 3$, and in particular, that n is large enough greater than 3 so that we may invoke the recurrence relation as many times as we like. Then

$$\begin{aligned}
T(n) &= n + 2T\left(\left\lfloor \frac{n}{3} \right\rfloor\right) && \text{by the recurrence formula} \\
&= n + 2\left[\left\lfloor \frac{n}{3} \right\rfloor + 2T\left(\left\lfloor \frac{\lfloor \frac{n}{3} \rfloor}{3} \right\rfloor\right)\right] && \text{by the recurrence formula} \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2 T\left(\left\lfloor \frac{n}{3^2} \right\rfloor\right) && \text{by distribution and laws of floors} \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2\left[\left\lfloor \frac{n}{3^2} \right\rfloor + 2T\left(\left\lfloor \frac{\lfloor \frac{n}{3^2} \rfloor}{3} \right\rfloor\right)\right] && \vdots \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2\left\lfloor \frac{n}{3^2} \right\rfloor + 2^3 T\left(\left\lfloor \frac{n}{3^3} \right\rfloor\right) \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2\left\lfloor \frac{n}{3^2} \right\rfloor + 2^3\left[\left\lfloor \frac{n}{3^3} \right\rfloor + 2T\left(\left\lfloor \frac{\lfloor \frac{n}{3^3} \rfloor}{3} \right\rfloor\right)\right] \\
&= n + 2\left\lfloor \frac{n}{3} \right\rfloor + 2^2\left\lfloor \frac{n}{3^2} \right\rfloor + 2^3\left\lfloor \frac{n}{3^3} \right\rfloor + 2^4 T\left(\left\lfloor \frac{n}{3^4} \right\rfloor\right)
\end{aligned}$$

   From this analysis it becomes apparent what the effect each iteration of the recurrence relation has on the overall expression for $T(n)$. We now generalize the expression to depth k:

$$T(n) = n + \sum_{i=1}^{k-1} 2^i \left\lfloor \frac{n}{3^i} \right\rfloor + 2^k T\left(\left\lfloor \frac{n}{3^k} \right\rfloor\right)$$

   The recursion bottoms out when the base case is satisfied; that is, when $1 \leq \lfloor \frac{n}{3^k} \rfloor < 3$. Since these bounds for $\lfloor \frac{n}{3^k} \rfloor$ are integers, we can disregard the floors and just deal with the expression in terms of $\frac{n}{3^k}$. With this, we solve for k as a function of n:

$$\begin{aligned}
&1 \leq \frac{n}{3^k} < 3 \\
&3^k \leq n < 3^{k+1} && \text{by multiplication by } 3^k \\
&k \leq \log_3 n < k+1 && \text{by logarithmization of base 3} \\
&k = \lfloor \log_3 n \rfloor && \text{by the inequality definition of floor}
\end{aligned}$$

For this value of k, we know that $T\left(\left\lfloor \frac{n}{3^k} \right\rfloor\right) = 6$. Hence the iterative formula for $T(n)$—written entirely as a function of n—is

$$T(n) = n \sum_{i=0}^{\lfloor \log_3 n \rfloor - 1} 2^i \left\lfloor \frac{n}{3^i} \right\rfloor + 6 \cdot 2^{\lfloor \log_3 n \rfloor}$$

b. Use the summation in (a) to show that $T(n) = O(n)$.

$\hookrightarrow$ The summation contains floors instead of ceilings, which are simple to estimate upward to conclude an asymptotic upper-bound. Then

$$\begin{aligned}
T(n) &= n + \sum_{i=0}^{\lfloor \log_3 n \rfloor - 1} 2^i \left\lfloor \frac{n}{3^i} \right\rfloor + 6 \cdot 2^{\lfloor \log_3 n \rfloor} \\
&\leq n + \sum_{i=0}^{\log_3 n - 1} 2^i \left(\frac{n}{3^i}\right) + 2^{\log_3 n} \cdot 6 && \text{since } \lfloor x \rfloor < x \text{ for all } x \\
&\leq n + n \sum_{i=0}^{\infty} \left(\frac{2}{3}\right)^i + 6n^{\log_3 2} && \text{since each } \left(\frac{2}{3}\right)^i > 0 \\
&= n + \frac{n}{1 - 2/3} + 6n^{\log_3 2} \\
&= n + 3n + 6n^{\log_3 2} \\
&= O(n) && \text{since } n^{\log_3 2} < n \text{ for all sufficiently large } n
\end{aligned}$$

c. Use the Master Theorem to show that $T(n) = \Theta(n)$.

$\hookrightarrow$ To use the Master Theorem, we must compare the functions n and $n^{\log_3 2}$. Since these functions are both polynomials, we must simply compare their degrees to compare them asymptotically. Notice that $2 < 3$ so that $\log_3 2 < \log_3 3 = 1$. Hence $n^{\log_3 2} \leq n$ for any $n \geq 1$, so we are dealing with case 3 of the Master Theorem. Let $\varepsilon = 1 - \log_3 2$. Then $n = n^{\log_3 2 + \varepsilon}$, so that $n = \Theta(n^{\log_3 2+\varepsilon})$, and in particular $n = \Omega(n^{\log_3 2+\varepsilon})$. To demonstrate the regularity condition, notice that $2\left(\frac{n}{3}\right) = \frac{2}{3}n \leq \frac{2}{3}n$, and that the inequality $0 < 2/3 < 1$ holds. By case 3 of the Master Theorem, we conclude that $T(n) = \Theta(n)$.

2. Use the Master Theorem to find asymptotic solutions to the following recurrences.

a. $T(n) = 7T(n/4) + n$

$\hookrightarrow$ We are comparing the functions n and $n^{\log_4 7}$. Notice that $4 < 7$ so that $1 = \log_4 4 < \log_4 7$. This means that $n \leq n^{\log_4 7}$, so we are in case 1 of the Master Theorem. Let $\varepsilon = \log_4 7 - 1 > 0$. Then $n = n^{\log_4 7 - \varepsilon}$, so $n = O(n^{\log_4 7 - \varepsilon})$. By case 1 of the Master Theorem, we conclude that $T(n) = \Theta(n^{\log_4(7)})$.

b. $T(n) = 9T(n/3) + n^2$

$\hookrightarrow$ We are comparing the functions n^2 and $n^{log_3 9}$. Since $log_3 9 = 2$, we are dealing with case 2 of the Master Theorem. Since $n^2 = \Theta(n^2) = \Theta(n^{\log_3 9})$, we conclude that by the Master Theorem that $T(n) = \Theta(n^{\log_3 9} \log(n)) = \Theta(n^2 \log(n))$.

c. $T(n) = 6T(n/5) + n^2$

$\hookrightarrow$ We are comparing the functions n^2 and $n^{\log_5 6}$. Notice that $6 < 25$, so that $\log_5 6 < \log_5 25 = 2$. Let $\varepsilon = 2 - \log_5 6$, so that $2 = \log_5 6 + \varepsilon$. Then $n^2 = \Omega(n^2) = \Omega(n^{\log_5 6+\varepsilon})$. To satisfy the regularity condition, notice that

$$6\left(\frac{n}{5}\right)^2 = \frac{6}{25}n^2 \leq \frac{6}{25}n^2$$

and that the inequality $0 < 6/25 < 1$ holds. By case 3 of the Master Theorem, we conclude that $T(n) = \Theta(n^2)$.

d. $T(n) = 6T(n/5) + n\log n$

$\hookrightarrow$ We are comparing the functions $n\log n$ and $n^{\log_5 6}$. Notice that $6 > 5$, so that $\log_5 6 > \log_5 5 = 1$. Let $\varepsilon = \frac{1}{2}(\log_5 6 - 1) > 0$, so that

$$2\varepsilon = \log_5 6 - 1 \qquad \Rightarrow \qquad 1 + \varepsilon = \log_5 6 - \varepsilon$$

This implies that $n\log n = o(n^{1+\varepsilon})$, since

$$\lim_{n\to\infty} \frac{n\log n}{n^{1+\varepsilon}} = \lim_{n\to\infty} \frac{\log n}{n^{\varepsilon}} = \lim_{n\to\infty} \frac{1/n}{\varepsilon n^{\varepsilon-1}} = \lim_{n\to\infty} \frac{1}{\varepsilon n^{\varepsilon}} = 0$$

But since $o(n^{1+\varepsilon}) \subseteq O(n^{1+\varepsilon})$, we know that $n\log n = O(n^{1+\varepsilon}) = O(n^{\log_5 6-\varepsilon})$. By case 1 of the Master Theorem, we conclude that $T(n) = \Theta(n^{\log_5 6})$.

e. $T(n) = 7T(n/2) + n^2$

$\hookrightarrow$ We are comparing the function n^2 and $n^{\log_2 7}$. Notice that $4 < 7$, so that $\log_2 4 = 2 < \log_2 7$. This implies that $n^2 < n^{\log_2 7}$, so we are in case 1 of the Master Theorem. Let $\varepsilon = \log_2 7 - 2 > 0$, so that $2 = \log_2 7 - \varepsilon$. Then $n^2 = O(n^2 = O(n^{\log_2 7-\varepsilon})$. By case 1 of the Master Theorem, we conclude that $T(n) = \Theta(n^{\log_2 7})$.

f. $S(n) = aS(n/4) + n^2$

$\hookrightarrow$ We are comparing the functions n^2 and $n\log_4 a$. Notice that these functions are equal when $\log_4 a = 2$, that is, $a = 16$. If a is any larger than this, then $n^{\log_4 a} > n^2$, and if a is any smaller, then $n^2 > n^{\log_4 a}$. We formalize these three cases to coincide with those of the Master Theorem:

1. Suppose $a > 16$. Then $\log_4 a > \log_4 16 = 2$, so that $n^{\log_4 a} > n^2$. Let $\varepsilon = \log_4 a - 2 > 0$, so that $2 = \log_4 a - \varepsilon$. This implies that $n^2 = O(n^2) = O(n^{\log_4 a-\varepsilon})$. By case 1 of the Master Theorem, we conclude that $S(n) = \Theta(n^{\log_4 a})$.
2. Suppose $a = 16$. Then $\log_4 a = \log_4 16 = 2$, so that $n^{\log_4 a} = n^2$. Thus $n^2 = \Theta(n^2) = \Theta(n^{\log_4 a})$. By case 2 of the Master Theorem, we conclude that $S(n) = \Theta(n^2\log n)$.
3. Suppose $a < 16$. Then $\log_4 a < \log_4 16 = 2$, so that $n^{\log_4 a} < n^2$. Let $\varepsilon = 2 - \log_4 a > 0$, so that $2 = \log_4 a + \varepsilon$. This implies that $n^2 = \Omega(n^2) = \Omega(n^{\log_4 a+\varepsilon})$. To satisfy the regularity condition, notice that

$$a\left(\frac{n}{4}\right)^2 = \frac{a}{16}n^2 \leq \frac{a}{16}n^2$$

and that the inequality $0 < a/16 < 1$ holds by supposition. By case 3 of the Master Theorem, we conclude that $S(n) = \Theta(n^2)$.

To summarize, $S(n)$ has the following asymptotic runtimes, as a function of a:

$$S(n) = \begin{cases} \Theta(n^{\log_4 a}) & a > 16 \\ \Theta(n^2\log n) & a = 16 \\ \Theta(n^2) & 1 \leq a < 16 \end{cases}$$

3. The recurrence $T(n) = 7T(n/2) + n^2$ describes the running time of an algorithm A. A competing algorithm B has a running time given by $S(n) = aS(n/4) + n^2$. What is the largest integer value for a such that B is a faster algorithm than A (asymptotically speaking)? In other words, find the largest integer a such that $S(n) = o(T(n))$.

   $\hookrightarrow$ From problem 2e, we know that the runtime of algorithms A is $\Theta(n^{\log_2 7})$, and from problem 2f, we can see the runtime, as a function of a, just above on this page. We wish to find the smallest a such that $S(n) = o(T(n))$—that is, when

   $$\lim_{n\to\infty} \frac{S(n)}{T(n)} = 0$$

   In order for this to happen, the inequality $\Theta(S(n)) < n^{\log_2 7} \approx n^{2.81}$ must hold for all sufficiently large n. Since we need the degree to be greater than about 2.81, we suppose $a > 16$ so we are in the case $S(n) = \Theta(n^{\log_4 a})$. That is, $\log_4 a < \log_2 7$, so that $a < 4^{\log_2 7} = \left(2^{\log_2 7}\right)^2 = 7^2 = 49$. Because a must be an integer, we conclude that $a = 48$ is the largest value for a such that $S(n) = o(T(n))$. So for $a = 48$—and no integer larger—B is a faster algorithm than A.

4. Let $T(n)$ satisfy the recurrence $T(n) = aT(n/b) + f(n)$, where $f(n)$ is a polynomial satisfying $\deg(f) > \log_b a$. Prove that case (3) of the Master Theorem applies, and in particular that the regularity condition necessarily holds.

   $\hookrightarrow$ **Proof.** We are comparing the functions $f(n)$, which we know to be polynomial, and $n^{\log_b a}$. By supposition, $\deg(f) > \log_b a$, so $n^{\deg(f)} > n^{\log_b a}$ for all sufficiently large n. Let $\varepsilon = \deg(f) - \log_b a > 0$, so that $\deg(f) = \log_b a + \varepsilon$. Then $f(n) = \Omega(n^{\deg(f)}) = \Omega(n^{\log_b a+\varepsilon})$. To demonstrate that the regularity condition holds, notice that

   $$a\left(\frac{a}{b}\right)^{\deg(f)} = \frac{a}{b^{\deg(f)}} n^{\deg(f)} \le \frac{a}{b^{\deg(f)}} n^{\deg(f)}$$

   From the supposition that $\deg(f) > \log_b a$, we know that $b^{log(f)} > a$ and so

   $$0 < \frac{a}{b^{\deg(f)}} < 1$$

   where the fraction is positive because $a \ge 1$ and $b > 1$ by supposition. Hence case 3 of the Master Theorem applies, and we conclude that $T(n) = \Theta(n^{\deg(f)})$. $\square$

5. Let G be an acyclic graph with n vertices, m edges, and k connected components. Show that $m = n - k$.

   $\hookrightarrow$ **Proof.** Since G is acyclic, each of its k connected components are trees. Call them $T_1, \ldots, T_k$, and let the number of vertices and edges in T_i be denoted by n_i and m_i, respectively. Since the graph formed by the union of the vertices of the T_i and the union of the edges of the T_i is G, the following equations hold:

   $$\sum_{i=1}^{k} n_i = n \qquad \text{and} \qquad \sum_{i=1}^{k} m_i = m$$

   By a theorem proved in class, on the induction handout, and in the midterm, if a tree has r vertices, then it has $r - 1$ edges. This means that for each i in the range $1 \le i \le k$, we have $m_i = n_i - 1$. Thus

   $$m = \sum_{i=1}^{k} m_i = \sum_{i=1}^{k} (n_i - 1) = \sum_{i=1}^{k} n_i - \sum_{i=1}^{k} 1 = n - k$$

So the graph G has $m = n - k$ edges. $\square$

6. Show that any connected graph G satisfies $|E(G)| \geq |V(G)| - 1$.

   $\hookrightarrow$ **Proof.** The proof is by induction on m, the number of edges in G. Let $P(m)$ be the proposition "If G is a connected graph with n vertices and m edges, then $m \leq n - 1$.".

   I. If $m = 0$, then there must be exactly one vertex in G, since zero vertices is disallowed by the definition of graph and more than one vertex would result in a disconnected graph. Hence $n = 1$, so that $0 \geq 0 = 1 - 1$, that is, $m \geq n - 1$. So $P(0)$ holds.

   IId. Let $m > 0$. Assume that for any connected graph G with fewer than m edges that $|E(H)| \geq |V(G)|$. Remove an edge e from G to form the subgraph G'. This results in two cases, depending on the connectedness of G':

   1 (G' is connected) Notice that G' has n vertices and $m - 1$ edges. By the induction hypothesis, we have $m - 1 \geq n - 1$. Thus $m \geq n - 1$, as desired.

   2 (G' is disconnected) In this case, G' has two connected components. Call them H_1 and H_2, and observe that each component contains fewer than m edges. Suppose that H_i has m_i edges and n_i vertices ($i = 1, 2$). By the induction hypothesis, $m_i \geq n_i - 1$ ($i = 1, 2$). Notice that $n = n_1 + n_2$ since no vertices were removed. Thus

   $$m = m_1 + m_2 + 1 \geq (n_1 - 1) + (n_2 - 1) + 1 = n_1 + n_2 - 1 = n - 1$$

   That is, $m \geq n - 1$.

   By the Second PMI, the proposition $P(m)$ holds for all $m \geq 0$. $\square$

7. Show that the number of vertices of odd degree in any graph must be even.

   $\hookrightarrow$ **Proof.** Let $G = (V, E)$ be a graph on n vertices. Partion the set of vertices V as follows:

   $$\begin{aligned} V_0 &= \{x \in V \mid \deg(x) \equiv 0 \mod 2\} && \text{set of vertices of even degree} \\ V_1 &= \{x \in V \mid \deg(x) \equiv 1 \mod 2\} && \text{set of vertices of odd degree} \end{aligned}$$

   Let the vertices in V_0 be denoted $v_1, \ldots, v_k$, so that for all $i \in \{1 \mathinner{..} k\}$, we have $\deg(v_i) = 2m_i$ for some integer m_i. Let the vertices in V_1 be denoted $u_1, \ldots, u_l$, so that for all $i \in \{1 \mathinner{..} l\}$, we have $\deg(u_i) = 2n_i + 1$ for some integer n_i. Then

   $$\begin{aligned} \sum_{x \in V_0} \deg(x) &= \sum_{i=1}^{k} 2m_i = 2\sum_{i=1}^{k} m_i \\ \sum_{x \in V_1} \deg(x) &= \sum_{i=1}^{l} 2n_i + 1 = l + 2\sum_{i=1}^{l} n_i \end{aligned}$$

   By the Handshake Lemma, we know that

   $$\begin{aligned} 2\,|E| &= \sum_{x \in V} \deg(x) \\ &= \sum_{x \in V_0} \deg(x) + \sum_{x \in V_1} \deg(x) && \text{since } V_0 \text{ and } V_1 \text{ partition } V \\ &= 2\sum_{i=1}^{k} m_i + l + 2\sum_{i=1}^{l} n_i && \text{by the above sum manipulations} \end{aligned}$$

In order for this equality to hold, the integer $l = |V_1|$ must be even. That is, the number of vertices of odd degree must be even. Because G is an arbitrary graph, the theorem is proved. $\square$