1 Recurrences

For this section, we will use $T(n) \simeq f(n) + kg(n) + c$ to mean that $\exists k_1, k_2, c_1, c_2$ fixed such that the following is true for all n:

$$f(n) + k_1 g(n) + c_1 \leq T(n) \leq f(n) + k_2 g(n) + c_2$$

Task 1.1 (5%).
Assuming $T(n) \simeq 3T(\frac{n}{2}) + kn + c$, we know the following information about our work tree:

$$\begin{aligned}
\text{Problem size at level i} &= \frac{n}{2^i} \\
\text{Work of node at level i} &\simeq k\frac{n}{2^i} + c \\
\text{Number of nodes at level i} &= 3^i \\
\text{Total work at level i} &\simeq 3^i k\frac{n}{2^i} + 3^i c \\
\text{Number of levels to tree} &= \log_2 n \\
\text{Total work of tree} &\simeq \sum_{i=0}^{\log_2 n} 3^i k\frac{n}{2^i} + 3^i c \\
&= nk\sum_{i=0}^{\log_2 n} \frac{3}{2}^i + c\sum_{i=0}^{\log_2 n} 3^i \\
&= nk\frac{1 - \frac{3}{2}^{\log_2 n+1}}{1-\frac{3}{2}} + c\frac{1-3^{\log_2 n}}{1-3} \\
&= 3kn^{\log_2 3} - 2kn + \frac{3}{2}cn^{\log_2 3} - 2c
\end{aligned}$$

From this final line, we conclude that $T(n) = \Theta(n^{\log_2 3})$.

Task 1.2 (5%).
Assuming $T(n) \simeq 2T(\frac{n}{4}) + k\sqrt{n} + c$, we know the following information about our work tree:

$$
\begin{aligned}
\text{Problem size at level i} &= \frac{n}{4^i} \\
\text{Work of node at level i} &\simeq k\sqrt{n/4^i} + c \\
\text{Number of nodes at level i} &= 2^i \\
\text{Total work at level i} &\simeq 2^i k\sqrt{n/4^i} + 2^i c \\
\text{Number of levels to tree} &= \log_4 n \\
\text{Total work of tree} &\simeq \sum_{i=0}^{\log_4 n} 2^i k\sqrt{n/4^i} + 2^i c \\
&= k\sum_{i=0}^{\log_4 n} \sqrt{n} + c\sum_{i=0}^{\log_4 n} 2^i \\
&= k\sqrt{n}\log_4(n) + 2c\sqrt{n} - c
\end{aligned}
$$

From this final line, we conclude that $T(n) = \Theta(\sqrt{n}\log n)$.

Task 1.3 (5%).
We proceed by the substitution method. We'll guess that $W(n) \in O(n)$. In fact, we'll make a stronger claim that

> There exist constants $k_1 > 0$ and $k_2 > 0$ such that for all $n \geq 1$, $W(n) \leq k_1 \cdot n - k_2 \cdot \sqrt{n}$.

Proof. Let $k > 0$ be a constant such that $W(n) \leq 4W(n/4) + k\sqrt{n}$. This constant exists by the definition of Θ. Then, set $k_1 = 2k + 1$ and $k_2 = k$. We begin with the base case. Clearly, $W(1) = k \leq k_1(1) - k_2(1)$. For the inductive step, we substitute the inductive hypothesis into the recurrence and obtain

$$
\begin{aligned}
W(n) &\leq 4W(n/4) + k\sqrt{n} \\
&\leq 4(k_1 n/4 - k_2\sqrt{n/4}) + k\sqrt{n} && \textit{(by IH since } n/4 < n\textit{)} \\
&= k_1 n - k_2\sqrt{n} + (k\sqrt{n} - k_2\sqrt{n}) \\
&\leq k_1 n - k_2\sqrt{n}.
\end{aligned}
$$

The final step follows because k_2 is such that $(k\sqrt{n} - k_2\sqrt{n}) \leq 0$ (remember we set $k_2 = k$). □

Thus, $W(n) \in O(n)$. Again, this bound is tight because we can see that the recursion tree of $W(n) = 2W(n/2) + \Theta(\sqrt{n})$ has n leaves, each costing at least a constant c, so $W(n) \geq c \cdot n \in \Omega(n)$.