Analysis of Recursive Algorithms
Solutions to Exercises

1. Steps 3-5 iterate 8 times and so take constant time.

   Recurrence relation: $T(n) = c + T(3n/4)$.

$$T(n) = c + T(3n/4) = c + c + T((3/4)^2 n) = c + c + c + T((3/4)^3 n)$$
$$= \underbrace{c + c + \ldots + c + T(1)}_{\log_{4/3}(n)} = \underbrace{c + c + \ldots + c + c}_{\log_{4/3}(n)} = \log_{4/3}(n)c \in \Theta(\log_2(n)).$$

   Therefore, $T(n) \in \Theta(\log_2(n))$.

2. Steps 2-8 take $\sum_{i=1}^{\lfloor n/2 \rfloor} \sum_{j=1}^{\lfloor n/3 \rfloor} c'n = cn^3$ time.

   Recurrence relation: $T(n) = cn^3 + T(n-5)$.

$$T(n) = cn^3 + T(n-5) = cn^3 + c(n-5)^3 + T(n-10)$$
$$= \underbrace{cn^3 + c(n-5)^3 + c(n-10)^3 + c(n-15)^3 + \ldots + T(1)}_{n/5}$$
$$= \underbrace{cn^3 + c(n-5)^3 + c(n-10)^3 + c(n-15)^3 + \ldots + c}_{n/5}$$
$$\leq \underbrace{cn^3 + cn^3 + cn^3 + cn^3 + \ldots + cn^3}_{n/5} = cn^3(n/5) = cn^4 \in O(n^4).$$

$$T(n) = cn^3 + T(n-5) = cn^3 + c(n-5)^3 + T(n-10)$$
$$\geq \underbrace{cn^3 + c(n-5)^3 + c(n-10)^3 + c(n-15)^3 + \ldots + c(n/2)^3}_{n/10}$$
$$\geq \underbrace{c(n/2)^3 + c(n/2)^3 + c(n/2)^3 + c(n/2)^3 + \ldots + c(n/2)^3}_{n/10}$$
$$= c(n^3/8)(n/10) = cn^4/80 \in \Omega(n^4).$$

   Therefore, $T(n) \in \Theta(n^4)$.

3. Steps 2-4 take cn time.

   Recurrence relation:

$$T(n) = cn + 4T(n/4)$$
$$= cn + 4(cn/4 + 4T(n/4^2)) = cn + cn + 4^2 T(n/4^2)$$
$$= cn + cn + 4^2(cn/4^2 + 4T(n/4^3)) = cn + cn + cn + 4^3 T(n/4^3)$$
$$= \underbrace{cn + cn + \ldots + cn + 4^{\log_4(n)} T(1)}_{\log_4(n)} = \underbrace{cn + cn + \ldots + cn + nc}_{\log_4(n)}$$
$$= cn \log_4(n) \in \Theta(n \log_2(n)).$$

4. Steps 2-6 take $\sum_{i=1}^{n} c'(n/3) = cn^2$ time.

   Recurrence relation:

$$T(n) = cn^2 + T(2n/3)$$
$$= cn^2 + c(2n/3)^2 + T((2/3)^2 n) = cn^2 + c(2n/3)^2 + c((2/3)^2 n)^2 + T((2/3)^3 n)$$
$$= cn^2 + c(2n/3)^2 + c((2/3)^2 n)^2 + c((2/3)^3 n)^2 + \ldots + c((2/3)^{\log_{3/2}(n)-1} n)^2 + T(1)$$
$$= cn^2(1 + (2/3)^2 + (2/3)^4 + (2/4)^6 + \ldots + (2/3)^{\log_{3/2}(n)-1}) + c$$
$$\approx cn^2(1 + (2/3)^2 + (2/3)^4 + (2/4)^6 + \ldots)$$
$$= cn^2(1/(1-(2/3)^2)) = cn^2(9/5) \in \Theta(n^2).$$

5. Steps 3-7 take $\sum_{i=1}^{\lfloor n/2 \rfloor} \lfloor n/2 \rfloor = cn^2$ time.

   Recurrence relation: $T(n) = cn^2 + T(n-5) + T(n-8)$.

$$\begin{aligned} T(n) &= cn^2 + T(n-5) + T(n-8) \geq T(n-5) + T(n-8) \geq T(n-8) + T(n-8) \\ &= 2T(n-8) \geq 2^2 T(n-16) \geq 2^3 T(n-24) \geq 2^{n/8} T(1) = c2^{n/8} \in \Omega(2^{n/8}). \end{aligned}$$

   Since $T(n) \in \Omega(2^{n/8})$, running time $T(n)$ has an exponential lower bound.

6. While loop (steps 3-6) iterates $n/\sqrt{n} = \sqrt{n}$ times.

   Recurrence relation: $T(n) = c\sqrt{n} + T(n-3)$.

$$\begin{aligned} T(n) &= c\sqrt{n} + T(n-3) = c\sqrt{n} + \sqrt{n-3} + T(n-6) \\ &= \underbrace{c\sqrt{n} + \sqrt{n-3} + \sqrt{n-6} + \ldots + c}_{n/3} \leq \underbrace{c\sqrt{n} + \sqrt{n} + \sqrt{n} + \ldots + \sqrt{n}}_{n/3} \\ &= c\sqrt{n}(n/3) \in O(n^{1.5}). \end{aligned}$$

$$\begin{aligned} T(n) &= c\sqrt{n} + T(n-3) = c\sqrt{n} + \sqrt{n-3} + T(n-6) \\ &= \underbrace{c\sqrt{n} + \sqrt{n-3} + \sqrt{n-6} + \ldots + c}_{n/3} \geq \underbrace{c\sqrt{n} + \sqrt{n-3} + \sqrt{n-6} + \ldots + \sqrt{n/2}}_{n/6} \\ &\geq \underbrace{c\sqrt{n/2} + \sqrt{n/2} + \sqrt{n/2} + \ldots + \sqrt{n/2}}_{n/6} = c\sqrt{n/2}(n/6) = cn^{1.5}/(6\sqrt{2}) \in \Omega(n^{1.5}). \end{aligned}$$

   Therefore $T(n) \in \Theta(n^{1.5})$.

7. Recurrence relation:

$$\begin{aligned} T(n) &= cn + T(n-3) + T(n-7) + T(n-11) + \ldots + T(1) \\ &\geq T(n-3) + T(n-7) \geq T(n-7) + T(n-7) \\ &= 2T(n-7) \geq 2^2 T(n-14) \geq 2^3 T(n-21) \geq 2^{n/7} T(1) \in \Omega(2^{n/7}). \end{aligned}$$

   Since $T(n) \in \Omega(2^{n/7})$, running time $T(n)$ has an exponential lower bound.

8. While loop (steps 3-6) takes $c \log_2(n)$ time.

   Recurrence relation: $T(n) = c \log_2(n) + T(n-6)$.

$$\begin{aligned} T(n) &= c\log_2(n) + T(n-6) = c\log_2(n) + c\log_2(n-6) + T(n-12) \\ &= c\log_2(n) + c\log_2(n-6) + c\log_2(n-12) + \ldots + T(1) \\ &= c\log_2(n) + c\log_2(n-6) + c\log_2(n-12) + \ldots + c \\ &\leq \underbrace{c\log_2(n) + c\log_2(n) + c\log_2(n) + \ldots + c\log_2(n)}_{n/6} \\ &= (n/6)c\log_2(n) \in O(n\log_2(n)). \end{aligned}$$

$$\begin{aligned} T(n) &= c\log_2(n) + c\log_2(n-6) + c\log_2(n-12) + \ldots + c \\ &\geq c\log_2(n) + c\log_2(n-6) + \ldots + c\log_2(n/2) \\ &\geq \underbrace{c\log_2(n/2) + c\log_2(n/2) + \ldots + c\log_2(n/2)}_{n/12} \\ &= c(n/12)\log_2(n/2) \in \Omega(n\log_2(n)). \end{aligned}$$

   Therefore, $T(n) \in \Theta(n\log_2(n))$.

9. Recurrence relation:

$$\begin{aligned}
T(n) &= cn + 2T(n/4) = cn + 2(c(n/4) + 2T(n/4^2)) = cn + cn(2/4) + 2^2T(n/4^2) \\
&= cn + cn(2/4) + cn(2/4)^2 + cn(2/4)^3 + \ldots + 2^kT(n/4^k) \text{ where } k = \log_4(n) \\
&= cn + cn(2/4) + cn(2/4)^2 + cn(2/4)^3 + \ldots + 2^{\log_4(n)}c \\
&= cn + cn(2/4) + cn(2/4)^2 + cn(2/4)^3 + \ldots + (2/4)^{\log_4(n)}cn \text{ since } n/4^{\log_4(n)} = 1 \\
&= cn(1 + (1/2) + (1/2)^2 + \ldots + (1/2)^{\log_4(n)}) \\
&\approx cn(2) \in \Theta(n).
\end{aligned}$$

10. Steps 3-7 take $\sum_{1=1}^{4} \sum_{j=1}^{n-i} c'(n/2) = cn^2$.
Recurrence relation:

$$\begin{aligned}
T(n) &= cn^2 + 4(T(n/2) = cn^2 + 4(c(n/2)^2 + 4T(n/2^2)) \\
&= cn^2 + cn^2 + 4^2T(n/2^2) = cn^2 + cn^2 + 4^2(c(n/2^2)^2 + 4T(n/2^3)) \\
&= cn^2 + cn^2 + cn^2 + 4^3T(n/2^3) \\
&= \underbrace{cn^2 + cn^2 + cn^2 + \ldots + cn^2 + 4^{\log_2(n)}T(1)}_{\log_2(n)} \\
&= \underbrace{cn^2 + cn^2 + cn^2 + \ldots + cn^2 + cn^2}_{\log_2(n)} \text{ since } 4^{\log_2(n)} = 2^{2\log_2(n)} = n^2 \\
&= cn^2 \log_2(n) \in \Theta(n^2 \log_2(n)).
\end{aligned}$$

11. Recurrence relation:

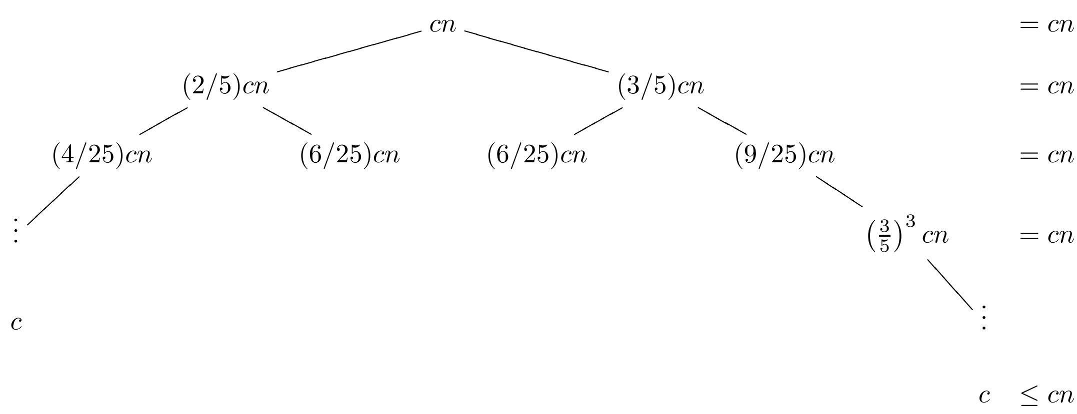
$$\begin{aligned}
T(n) &= T(n-2) + T(n-6) + T(n-18) + T(n-54) + \ldots + T(n - 2 \times 3^k) + \ldots \\
&\geq T(n-2) + T(n-6) \geq T(n-6) + T(n-6) = 2T(n-6). \\
T(n) &\geq 2T(n-6) \geq 2 \times 2 \times T(n-12) \geq 2 \times 2 \times 2 \times T(n-18) \\
&\geq \underbrace{2 \times 2 \times 2 \times \cdots \times 2}_{n/6} \times T(1) = 2^{n/6}c \in \Omega(2^{n/6}).
\end{aligned}$$

Since $T(n) \in \Omega(2^{n/6})$, running time $T(n)$ has an exponential lower bound.

12. Recurrence relation:

$$T(n) = cn + T(2n/5) + T(3n/5).$$

Recursion tree:

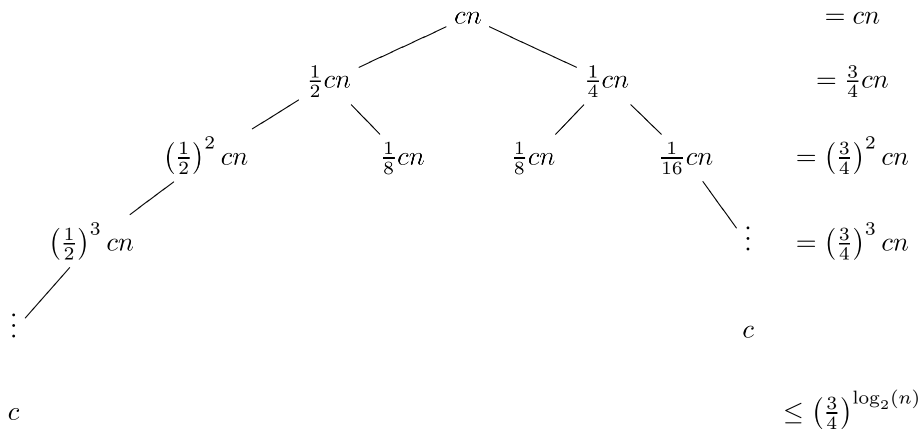


The height of this tree is $\log_{5/3}(n)$ so the total work at all nodes is at most $cn \log_{5/3}(n)$.
The shortest path in this tree is $\log_{5/2}(n)$ so the total work at all nodes is at least $cn \log_{5/2}(n)$.
Thus, $cn \log_{5/2}(n) \leq T(n) \leq cn \log_{5/3}(n)$ and $T(n) \in \Theta(n \log_2(n))$.

13. Recurrence relation:

$$T(n) = cn + T(n/2) + T(n/4).$$

Recursion tree:



Note: At each level, the total number of elements processed goes down by (3/4).
The height of this tree is $\log_2(n)$.

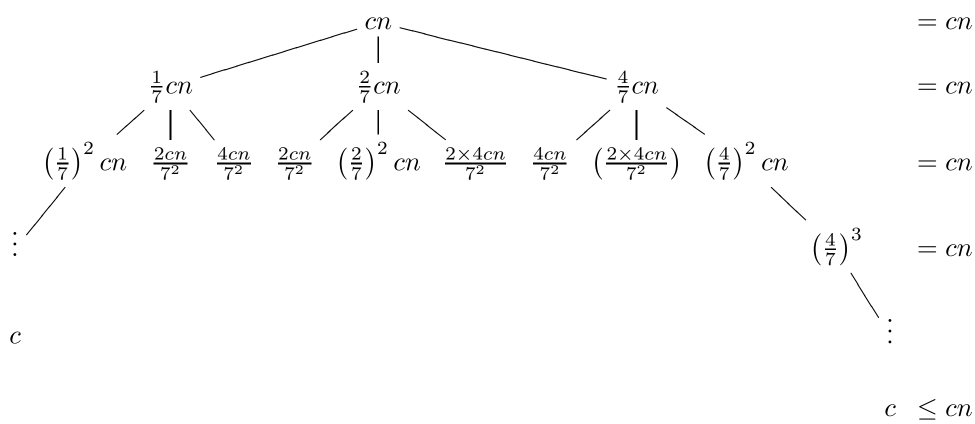
$$\begin{aligned}
T(n) &= cn + (3/4)cn + (3/4)^2cn + (3/4)^3cn + \ldots + (3/4)^{\log_2(n)} \\
&\leq cn + (3/4)cn + (3/4)^2cn + (3/4)^3cn + \ldots \\
&= cn(1 + (3/4) + (3/4)^2 + (3/4)^3 + \ldots) = cn\frac{1}{1-(3/4)} = 4cn. \\
T(n) &= cn + (3/4)cn + (3/4)^2cn + (3/4)^3cn + \ldots + (3/4)^{\log_2(n)} \geq cn.
\end{aligned}$$

Since $cn \leq T(n) \leq 4cn$, running time $T(n) \in \Theta(n)$.

14. Recurrence relation:

$$T(n) = cn + T(n/7) + T(2n/7) + T(4n/7).$$

Recursion tree:



The height of this tree is $\log_{7/4}(n)$ so the total work at all nodes is at most $cn\log_{7/4}(n)$.
The shortest path in this tree is $\log_7(n)$ so the total work at all nodes is at least $cn\log_7(n)$.
Thus, $cn\log_7(n) \leq T(n) \leq cn\log_{7/4}(n)$ and $T(n) \in \Theta(n\log_2(n))$.