CSCI3070U: Analysis and Design of Algorithms
Assignment 1

Syed Arham Naqvi

100590852

October 3, 2023

Part 1.

a.

Given: $\mathbf{T(n) = 8T\left(\frac{n}{2}\right) + nlog_2(n)}$.

$\mathbf{a} = 8$, $\mathbf{b} = 2$ and $\forall n > 1, \quad \mathbf{f(n)} = n \cdot \log_2(n) > 0$.

Note that for $c = 1$ and $\forall n > n_0 = 1$:

$$\begin{aligned} f(n) &= n \cdot \log_2(n) \\ &\leq n^2 \\ &\leq n^{\log_2(8)} \\ &= n^3 \end{aligned}$$

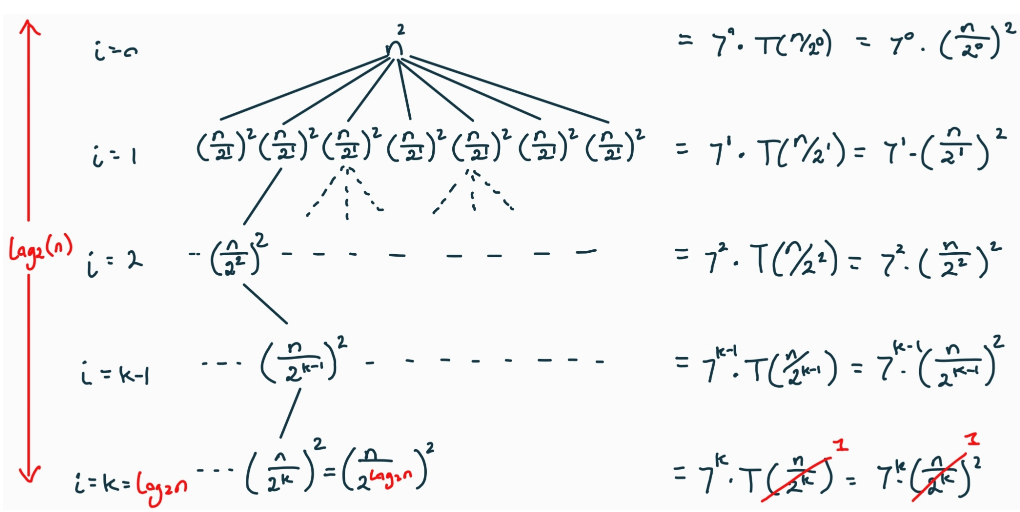
So for $\epsilon = 1 > 0$ we can see that $f(n) = \mathcal{O}(n^{\log_2(8)-1}) = \mathcal{O}(n^{3-1}) = \mathcal{O}(n^2)$.

$\therefore$ by case 1 of the Master Theorem $T(n) = \Theta(n^3)$ which implies $T(n) = \mathcal{O}(n^3)$

b.

Given: $\mathbf{T(n) = 7T\left(\frac{n}{2}\right) + n^2}$.

We can determine the upper bound of this recurrence using a recursion tree:



We find that the complexity of a given recursion level (i), is represented by the expression:

$$7^i \cdot \left(\frac{n}{2^i}\right)^2 \tag{1}$$

Which can be simplified to:

$$n^2 \cdot \left(\frac{7}{4}\right)^i \tag{2}$$

We now sum the cost of each level of recursion for $i \in [0, 1 \ldots, k]$ where $k = \log_2(n)$:

$$\begin{aligned}
T(n) &= \sum_{i=0}^{k} \left(n^2 \cdot \left(\frac{7}{4}\right)^i\right) \\
&= n^2 \cdot \sum_{i=0}^{k} \left(\frac{7}{4}\right)^i \\
&= n^2 \cdot \left(\frac{\left(\frac{7}{4}\right)^{k+1} - 1}{\left(\frac{7}{4}\right) - 1}\right) \\
&= n^2 \cdot \left(\frac{\left(\frac{7}{4}\right)^{\log_2(n)+1} - 1}{\left(\frac{3}{4}\right)}\right) \\
&= n^2 \cdot \left(\frac{\left(\frac{7}{4}\right)^{\log_2(n)} \cdot \left(\frac{7}{4}\right) - 1}{\left(\frac{3}{4}\right)}\right) \\
&= n^2 \cdot \left(\frac{n^{\log_2\left(\frac{7}{4}\right)} \cdot \left(\frac{7}{4}\right) - 1}{\left(\frac{3}{4}\right)}\right) \\
&= n^2 \cdot \left(\frac{n^{\log_2(7) - \log_2(4)} \cdot \left(\frac{7}{4}\right) - 1}{\left(\frac{3}{4}\right)}\right) \\
&= n^2 \cdot \left(\frac{n^{\log_2(7) - 2} \cdot \left(\frac{7}{4}\right) - 1}{\left(\frac{3}{4}\right)}\right) \\
&= \frac{n^{\log_2(7)} \cdot \left(\frac{7}{4}\right) - n^2}{\left(\frac{3}{4}\right)} \\
&= \left(\frac{7}{3}\right) n^{\log_2(7)} - \left(\frac{4}{3}\right) n^2 \\
&\leq c \cdot n^{\log_2(7)} \;\text{ for }\; c = 3 \;\text{ and }\; \forall n > n_0 = 1
\end{aligned}$$

$\therefore$ we have demonstrated using a recursion tree that $T(n) = \mathcal{O}(n^{log_2(7)})$.

c.

Given: $\mathbf{T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + T\left(\frac{n}{8}\right) + 1}$.
Guess: $T(n)$ is $\mathcal{O}(\log_2(n))$.

For the inductive process, we will assume that $n_0 = 2$ and that $T(n_0) = T(2) = 2$ for our inductive base case.

Induction:
Basis: For $n_0 = 2$, $T(2) = 2 \leq c \cdot \log_2(2)$, $\forall c \geq 2$.
Inductive step: Inductive hypothesis is that $\forall k < n$, $T(k) \leq c \cdot \log_2(k)$ where $c \geq 2$.

$$\begin{aligned}
T(n) &= T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + T\left(\frac{n}{8}\right) + 1 \\
&\leq c_1 \cdot \log_2\left(\frac{n}{2}\right) + c_2 \cdot \log_2\left(\frac{n}{4}\right) + c_3 \cdot \log_2\left(\frac{n}{8}\right) + 1 \\
&= c_1 \cdot (\log_2 n - \log_2 2) + c_2 \cdot (\log_2 n - \log_2 4) + c_3 \cdot (\log_2 n - \log_2 8) + 1 \\
&= c_1 \cdot (\log_2 n - 1) + c_2 \cdot (\log_2 n - 2) + c_3 \cdot (\log_2 n - 3) + 1 \\
&\leq c_1 \cdot (\log_2 n) + c_2 \cdot (\log_2 n) + c_3 \cdot (\log_2 n) + c_4 \cdot (\log_2 n) \\
&= \underbrace{(c_1 + c_2 + c_3 + c_4)}_{c} \cdot (\log_2 n) \\
&= c \cdot \log_2 n
\end{aligned}$$

$\therefore$ we have shown using the substitution method and the inductive process that $T(n) = \mathcal{O}(\log_2 n)$.

d.

Given: $\mathbf{T(n) = \sqrt{200} \cdot T\left(\frac{n}{2}\right) + n^{\sqrt{200}}}$.

$\mathbf{a} = \sqrt{200}$, $\mathbf{b} = 2$ and $\forall n > 0$, $\mathbf{f(n)} = n^{\sqrt{200}} > 0$.

If we can find some $t > 0$ such that $f(n) = n^{\log_b a + t}$, then we can find some ϵ where $0 < \epsilon < t$ so that $f(n) > n^{\log_b a + \epsilon}$:

$$\begin{aligned}
f(n) &= n^{\log_b(a)+t} \\
n^{\sqrt{200}} &= n^{\log_2(\sqrt{200})+t} \\
10\sqrt{2} &= \log_2(10\sqrt{2}) + t \\
t &= 10\sqrt{2} - \log_2(10\sqrt{2}) \\
t &\approx 10.32
\end{aligned}$$

Let $\epsilon = 5$ so that $0 < \epsilon < t$ and $n^{\sqrt{200}} > n^{\log_2(\sqrt{200})+5}$ which means $f(n) = \Omega(n^{\log_b(a)+\epsilon})$.
We now verify the regularity condition:

$$\begin{aligned}
af\left(\frac{n}{2}\right) &\leq cf(n) \\
\sqrt{200}\left(\frac{n}{2}\right)^{\sqrt{200}} &\leq cn^{\sqrt{200}} \\
\frac{(\sqrt{200})\cancel{(n^{\sqrt{200}})}}{\cancel{(n^{\sqrt{200}})}(2^{\sqrt{200}})} &\leq c \\
7.83 \cdot 10^{-4} &\leq c
\end{aligned}$$

Thus if we let $c = \frac{1}{2}$, the regulaity condition holds.

$\therefore$ by case 3 of the master theorem, $T(n) = \Theta(n^{\sqrt{200}})$ which implies that $T(n) = \mathcal{O}(n^{\sqrt{200}})$.

Part 2.

a.

Proposition: For all positive $f(n)$, $g(n)$ and $h(n)$, if $f(n) = \mathcal{O}(g(n))$ and $h(n) = \Omega(f(n))$ then $g(n) + h(n) = \Omega(f(n))$.

This proposition is **true**.

If $f(n) = \mathcal{O}(g(n))$ then $\exists c, n_0 > 0 \mid \forall n \geq n_0,\ f(n) \leq c \cdot g(n)$.
Let $k = \frac{1}{c}$ so $k > 0$ since $c > 0$.
It follows that $k \cdot f(n) \leq g(n)$ meaning that $g(n) = \Omega(f(n))$.

If $h(n) = \Omega(f(n))$ then $\exists t, p_0 > 0 \mid \forall n \geq p_0,\ t \cdot f(n) \leq h(n)$.

Now,

$$\begin{aligned} t \cdot f(n) + k \cdot f(n) &\leq h(n) + g(n) && \text{since } f(n), g(n), h(n), k, t > 0 \\ \underbrace{(t+k)}_{r} \cdot f(n) &\leq h(n) + g(n) && \\ r \cdot f(n) &\leq h(n) + g(n) && \text{where } r > 0 \\ h(n) + g(n) &= \Omega(f(n)) && \end{aligned}$$

$\therefore$ if $f(n) = \mathcal{O}(g(n))$ and $h(n) = \Omega(f(n))$ then $g(n) + h(n) = \Omega(f(n))$.

b.

Proposition: If $f(n) = \mathcal{O}(g(n))$ and $g(n) = \mathcal{O}(f(n))$ then $f(n) = g(n))$.

This proposition is **false**.

Let $f(n) = n^2 + n$ and $g(n) = n^2$ so $f(n) \neq g(n)$.
Take $k = 1$ and $n_0 = 1$, then $\forall n \geq n_0,\ g(n) \leq k \cdot f(n)$ meaning $g(n) = \mathcal{O}(f(n))$.
Now consider,

$$\begin{aligned} f(n) &\leq c \cdot g(n) && \\ n^2 + n &\leq c \cdot n^2 && \\ 1 + \frac{1}{n} &\leq c && \text{since } n > 0 \\ \lim_{n \to \infty} \left(1 + \overset{0}{\cancel{\frac{1}{n}}} \right) &\leq \lim_{n \to \infty} (c) && \\ 1 &\leq c && \end{aligned}$$

If we let $c = 2$ then,

$$\begin{aligned} n^2 + n &= 2 \cdot n^2 \\ 1 + \frac{1}{n} &= 2 \\ n &= 1 \end{aligned}$$

so $\forall n > n_0 = 1$ and for $c = 2$, $f(n) \leq c \cdot g(n)$ which means $f(n) = \mathcal{O}(g(n))$.

$\therefore$ we have shown it to be possible that $g(n) = \mathcal{O}(f(n))$ and $f(n) = \mathcal{O}(g(n))$ and yet $f(n) \neq (g(n))$.