Exercise Session 04

Exercise 1.

Theorem 4.1 (Master theorem)
Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence

$$T(n) = aT(n/b) + f(n) ,$$

where we interpret n/b to mean either $\lfloor n/b \rfloor$ or $\lceil n/b \rceil$. Then $T(n)$ has the following asymptotic bounds:

1. If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
2. If $f(n) = \Theta(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a} \lg n)$.
3. If $f(n) = \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$, and if $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$. ■

You can follow these steps:

1. Identify if the recurrence is in the proper format $T(n) = aT(n/b) + f(n)$ (where $a \geq 1$ and $b > 1$)
2. Simplify $n^{\log_b a}$ by plugging in the values of a and b
3. Check if one of the 3 cases holds:
   1. $f(n) = O(n^{\log_b a - \epsilon}) = O(n^{\log_b a}/n^{\epsilon})$ for some $\epsilon > 0$
   2. $f(n) = \Theta(n^{\log_b a})$
   3. $f(n) = \Omega(n^{\log_b a + \epsilon}) = \Omega(n^{\log_b a} \cdot n^{\epsilon})$ for some $\epsilon > 0$.
      Moreover, check if there exist $n_0 \in \mathbb{N}$ and $0 < c < 1$ such that $af(n/b) \leq cf(n)$ for all $n \geq n_0$.
4. Simplify the corresponding conclusion of the Theorem by plugging in the values of a, b, and $f(n)$

Consider the following recurrence $T(n) = T(2n/3) + \Theta(1)$. Prove that $T(n) = O(\lg n)$.

$$\begin{aligned}
a = 1, b &= \frac{3}{2} \\
f(n) = \Theta(1) &= (n^{log_b(a)}) = (n^{log_{\frac{3}{2}}(1)}) = (n^0) = 1 \\
T(n) &= \Theta(n^{log_b(a)} \cdot log(n)) \\
&= \Theta(n^{log_{\frac{3}{2}}(1)} \cdot log(n)) \\
&= \Theta(1 \cdot log(n)) \\
&= \Theta(log(n))
\end{aligned}$$

Using the seconds case of the master theorem and substituting the values of a and b into the equation we get that $\Theta T(n) = \Theta(log(n))$

Exercise 2.

Consider the following recurrence:

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ T(n-1) + \Theta(n) & \text{if } n > 1 \end{cases}$$

Prove that $T(n) = O(n^2)$ using the substitution method.

$T(n) = O(n^2)$

$$\begin{aligned}
T(n) &= T(n-1) + \Theta(n) \\
T(m) &\leq cn^2, \forall n_0 \leq m < n \\
T(m) &\leq c(n-1)^2 + \Theta(n) \\
&= c(n^2 + 1 - 2n) + \Theta(n) \\
&= cn^2 - 2cn + c + \Theta(n) \\
&\leq cn^2
\end{aligned}$$

In order to complete the proof we need to show that the hypothesis holds for some $n \geq n_0$.

$T(1) = 1$ We can verify our hypothesis by choosing some $c \geq 1$

$T(1) \leq 1 \cdot 1^2 = 1$

This shows that the inequality $T(1) \leq T'(1)$ hold which proves that our hypothesis holds for $c \geq 1$

Exercise 3.

Consider the following recurrence:

$$T(n) = \begin{cases} \Theta(1) & \text{if } n \leq 1 \\ T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil - 1) + \Theta(n) & \text{if } n > 1 \end{cases}$$

Use the substitution method to prove that $T(n) = O(n \lg n)$.

Hint: be careful when you choose the base case because $n = 0$ and $n = 1$ may not work

$$\begin{aligned}
T(m) &\leq cm \cdot log(m) \\
T(n) &= T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil - 1) + \Theta(n) \\
&\leq c(\lfloor n/2 \rfloor) \cdot log(\lfloor n/2 \rfloor) + c(\lceil n/2 \rceil - 1) \cdot log(\lceil n/2 \rceil - 1) + \Theta(n) \\
&\leq c(\lfloor n/2 \rfloor) \cdot log(n/2) + c\lceil n/2 \rceil \cdot log(n/2) + \Theta(n) \\
&\leq c(\lfloor n/2 \rfloor) \cdot log(n) - c(\lfloor n/2 \rfloor) \cdot log(2) + c\lceil n/2 \rceil \cdot log(n) - c\lceil n/2 \rceil \cdot log(2) + \Theta(n) \\
&\leq c(\lfloor n/2 \rfloor) \cdot log(n) - c(\lfloor n/2 \rfloor) + c\lceil n/2 \rceil \cdot log(n) - c\lceil n/2 \rceil + \Theta(n) \\
&= cn \cdot log(n) - c(\lfloor n/2 \rfloor) - c\lceil n/2 \rceil + \Theta(n) \\
&\leq cn \cdot log(n)
\end{aligned}$$

In order to complete the proof we need to show that the hypothesis holds for some $n \geq n_0$.

$$T(1) = 1 \cdot log(1) \neq 1$$

This shows that it doesn't hold for $n = 1$. So we need to try a larger n_0

$$T(2) = 2 \cdot log(2) = 2$$

We can verify our hypothesis by choosing some $c \geq 1$

This shows that the inequality $T(2) \leq 1 \cdot 2 \cdot log(2) = 2 \cdot 1 = 2$ holds for $n \geq 2$. Proving our hypothesis correct

Exercise 4.
The factorial of n, is usually recursively defined as

$$n! = \begin{cases} 1 & \text{if } n = 0, \\ n \cdot (n-1)! & \text{if } n > 0 \end{cases}$$

(a) Prove that $n! = \Omega(2^n)$.

(b) Prove that $n! = O(n^n)$.

(c) Prove that $\lg n! = O(n \lg n)$.

a)

$$\begin{aligned} m! &\geq c2^n \\ n! &= n \cdot (n-1)! \\ &\geq n \cdot c2^{(n-1)} \\ &\geq c2^n \end{aligned}$$

b)

$$\begin{aligned} m! &\leq cn^n \\ n! &= n \cdot (n-1)! \\ &\leq n \cdot c(n-1)^{n-1} \\ &\leq n \cdot cn^{n-1} \\ &= cn^n \end{aligned}$$

c)

$$\begin{aligned} log(m!) &\leq cn \cdot log(n)) \\ log(n!) &= log(n \cdot (n-1)!) \\ &= log(n) + log((n-1)!) \\ &\leq log(n) + c(n-1) \cdot log(n-1) \\ &\leq log(n) + c(n-1) \cdot log(n) \\ &\leq c \cdot log(n) + c(n-1) \cdot log(n) \\ &= cn \cdot log(n) \end{aligned}$$

★ **Exercise 5.**
Consider the recurrence

$$T(n) = \begin{cases} b & \text{if } n = 1 \text{ or } n = 0 \\ T(\lfloor 9n/10 \rfloor) + T(\lfloor n/10 \rfloor) + cn & \text{if } n \geq 1 \end{cases}$$

where b and c are constants such that $b, c > 0$. Prove that $T(n) = O(n \lg n)$.