

Taller: Método Maestro

Parte 1: Resolución Matemática

Ejercicio 1:

Resuelve la siguiente ecuación de recurrencia utilizando el Método Maestro:

$$T(n) = 4 \cdot T\left(\frac{n}{2}\right) + O(n^2)$$

- Identifica los valores de a , b y $f(n)$.
- Compara $f(n)$ con $n^{\log_b a}$.
- Determina la complejidad final $T(n)$.

Ejercicio 2:

Resuelve la siguiente ecuación de recurrencia utilizando el Método Maestro:

$$T(n) = 3 \cdot T\left(\frac{n}{3}\right) + O(n)$$

- Identifica los valores de a , b y $f(n)$.
- Compara $f(n)$ con $n^{\log_b a}$.
- Determina la complejidad final $T(n)$.

Ejercicio 3:

Resuelve la siguiente ecuación de recurrencia utilizando el Método Maestro:

$$T(n) = 5 \cdot T\left(\frac{n}{2}\right) + O(n \log n)$$

- Identifica los valores de a , b y $f(n)$.
- Compara $f(n)$ con $n^{\log_b a}$.
- Determina la complejidad final $T(n)$.

Parte 2: Implementación Algorítmica

Desarrolla los siguientes algoritmos, implementando una solución para cada ecuación de recurrencia utilizando el **Método Maestro** para encontrar la complejidad.

1. **Algoritmo 1: Merge Sort** Implementa el algoritmo **Merge Sort** y analiza su complejidad utilizando el Método Maestro.
2. **Algoritmo 2: Búsqueda Binaria** Implementa el algoritmo de **Búsqueda Binaria** y analiza su complejidad utilizando el Método Maestro.
3. **Algoritmo 3: Quick Sort** Implementa el algoritmo **Quick Sort** y analiza su complejidad utilizando el Método Maestro.

Parte 4:

Análisis de Complejidad con el Método Maestro

Contexto:

Tienes una base de datos que contiene una tabla llamada personas, con la siguiente estructura:

```
CREATE TABLE IF NOT EXISTS personas (  
  id INT PRIMARY KEY,  
  nombre VARCHAR(255),  
  direccion VARCHAR(255),  
  telefono VARCHAR(255),  
  email VARCHAR(255),  
  fecha_nacimiento DATE  
);
```

Además, la base de datos está poblada con 1,000 registros de personas desordenadas, como el archivo **personas_desordenadas.sql** que te proporcioné previamente.

Tarea:

1. Diseño de Algoritmo:

Supón que necesitas diseñar un algoritmo que realice una búsqueda en la tabla personas para encontrar a todas las personas que cumplan con una condición específica, como por ejemplo:

- Buscar todas las personas cuyo nombre comience con una letra específica.
- Buscar todas las personas nacidas antes de una fecha específica.

¿Cuál sería el tiempo de ejecución de este algoritmo, dependiendo de cómo se realice la búsqueda en la base de datos?

2. Formulación de la Recurrencia:

Al pensar en el algoritmo de búsqueda, ¿cómo podrías formular la recurrencia para describir el comportamiento temporal de la búsqueda? Considera:

- Si el algoritmo recursivo divide la búsqueda en partes más pequeñas (por ejemplo, divide los registros en subconjuntos más pequeños para buscar eficientemente).
- El costo de procesar los datos en cada nivel de la recursión.

Supón que el algoritmo realiza la búsqueda dividiendo los registros en dos partes (subconjunto superior e inferior) y luego recursivamente busca en ambas partes. Si además, el costo de dividir y combinar los resultados es $O(n)$, ¿cómo se ve la ecuación de recurrencia?

3. Aplicación del Método Maestro:

Usa el **Método Maestro** para analizar la complejidad de la ecuación de recurrencia y determinar el tiempo total de ejecución del algoritmo.

La ecuación de recurrencia es:

$$T(n) = 2 \cdot T\left(\frac{n}{2}\right) + O(n)$$

Resuelve la recurrencia utilizando el **Método Maestro** y determina la complejidad del algoritmo.

Interpretación:

Una vez resuelto el problema, explica cómo la complejidad afecta el rendimiento de tu algoritmo en función del tamaño de la tabla personas. Si el número de registros aumenta considerablemente, ¿cómo cambiaría la complejidad del algoritmo?

Pistas para los Estudiantes:

- **Recurrencia de Búsqueda Binaria:** Piensa en el caso donde divides los datos en dos mitades (como en la búsqueda binaria). El costo de cada nivel de búsqueda es proporcional al número de registros, pero se va reduciendo a la mitad en cada recursión.
- **Método Maestro:** Recuerda que el **Método Maestro** se usa para resolver recursiones de la forma:

$$T(n) = a \cdot T\left(\frac{n}{b}\right) + O(n^d)$$

Identifica los valores de a , b y d , y compara n^d con $n^{\log_b a}$ para determinar en qué caso cae la recurrencia.