

Guía de Estudio: Paralelismo en Sistemas Informáticos

1. Introducción al Paralelismo

El paralelismo es un concepto fundamental en la informática que permite que múltiples tareas o procesos se ejecuten simultáneamente para mejorar el rendimiento. A diferencia de la concurrencia, que se centra en manejar múltiples tareas que pueden ejecutarse en cualquier orden, el paralelismo permite que dichas tareas se realicen al mismo tiempo en sistemas multinúcleo o multiprocesador. Este enfoque es especialmente útil en el procesamiento de grandes volúmenes de datos o en tareas computacionales intensivas, como simulaciones científicas o gráficos.

2. Núcleos de Procesamiento y Seguridad en Hilos

Los núcleos de procesamiento (cores) son unidades dentro de un procesador que ejecutan instrucciones de forma independiente. En sistemas con múltiples núcleos, cada núcleo puede ejecutar un hilo, o una tarea, al mismo tiempo que otros núcleos ejecutan sus propias tareas, lo que permite un verdadero paralelismo. Sin embargo, cuando varios hilos acceden a los mismos recursos, es crucial que esas operaciones sean 'thread-safe' o seguras en hilos. Esto implica que los datos compartidos no se corrompan debido a la concurrencia de múltiples hilos, lo cual se puede lograr a través de técnicas como la exclusión mutua (mutex).

3. Condiciones de Carrera y Sincronización

Una condición de carrera ocurre cuando dos o más hilos intentan acceder y modificar un recurso compartido al mismo tiempo, causando comportamientos inesperados en el programa. Para evitar esto, se utilizan técnicas de sincronización, como los bloqueos y las variables de condición. Estas técnicas garantizan que solo un hilo pueda acceder a un recurso crítico a la vez, eliminando la posibilidad de condiciones de carrera.

4. Memoria Compartida vs. Paso de Mensajes

En programación paralela, existen dos modelos comunes para que los hilos o procesos intercambien información: la memoria compartida y el paso de mensajes. En el modelo de memoria compartida, todos los hilos pueden acceder a una misma área de memoria, lo que permite una comunicación directa y rápida. Sin embargo, esto requiere técnicas de sincronización para evitar problemas de consistencia de datos. El paso de mensajes, en cambio, utiliza canales de comunicación donde los hilos envían y reciben mensajes para intercambiar datos, lo que evita problemas de sincronización, pero puede ser más lento debido a la sobrecarga de comunicación.

5. Deadlocks, Balanceo de Carga y Sobrecarga

Un 'deadlock' o bloqueo mutuo ocurre cuando varios hilos esperan indefinidamente por recursos que están en posesión de otros hilos, resultando en una situación donde ningún hilo puede avanzar. Los deadlocks se evitan generalmente mediante el diseño cuidadoso de

los bloqueos y la asignación ordenada de recursos. El balanceo de carga, por otro lado, se refiere a la distribución equitativa de tareas entre los núcleos disponibles para optimizar el uso de los recursos y evitar la sobrecarga en algunos núcleos. El término 'overhead' se refiere al tiempo adicional y recursos necesarios para gestionar el paralelismo, como la creación y sincronización de hilos.

6. Latencia y Ancho de Banda en el Paralelismo

La latencia y el ancho de banda son factores críticos en el rendimiento de sistemas paralelos. La latencia se refiere al tiempo que tarda en iniciarse una tarea o en transferirse datos, mientras que el ancho de banda es la cantidad de datos que pueden procesarse o transferirse en un período determinado. En aplicaciones paralelas, reducir la latencia y maximizar el ancho de banda son objetivos clave para mejorar la eficiencia general del sistema, especialmente en sistemas distribuidos donde los datos deben transferirse entre varios nodos o máquinas.