

Tarea 1.6 PSP UT1 Isaac Solano Delgado

Escribe alguna característica de un programa concurrente.

Un programa concurrente es capaz de **realizar varias tareas al mismo tiempo o alternando entre ellas**. Aunque no siempre se ejecutan simultáneamente (especialmente en sistemas con un solo núcleo), el sistema gestiona las tareas de manera que puedan compartir los recursos, lo que ayuda a aprovechar mejor la CPU y a conseguir una mayor velocidad de ejecución, al subdividir un programa en procesos, éstos se pueden "repartir" entre procesadores o gestionar en un único procesador según la importancia.

¿Cuál es la ventaja de la concurrencia en los sistemas monoprocesador?

En un sistema con un solo procesador, la concurrencia permite que el procesador se **aproveche mejor** al cambiar entre diferentes tareas, en lugar de esperar a que una tarea termine completamente. Esto permite que el sistema parezca **más eficiente y responsivo**, ya que, por ejemplo, mientras un programa espera por una operación de disco o red, el procesador puede ejecutar otra tarea.

¿Cuáles son las diferencias entre multiprogramación y multiproceso?

- **Multiprogramación:** Es una técnica en la que el sistema operativo mantiene varios programas en la memoria y va alternando su ejecución, para que cuando uno esté esperando, otro pueda ejecutarse. No se ejecutan a la vez, sino que el sistema cambia entre ellos rápidamente para **optimizar el uso de la CPU**.
- **Multiproceso:** En sistemas multiprocesador (con varios núcleos o CPUs), varios procesos pueden realmente **ejecutarse de manera simultánea**. Esto permite un verdadero paralelismo, donde diferentes programas o partes de un programa (hilos) se ejecutan al mismo tiempo, uno en cada núcleo.

¿Cuáles son los dos problemas principales inherentes a la programación concurrente?

- **Exclusión mutua:** Cuando varios procesos acceden simultáneamente a una variable compartida, puede generar inconsistencias en los datos si uno intenta leer mientras otro la actualiza. Para evitar esto, se usa la región crítica, que es una sección del código donde solo un proceso puede acceder a la variable compartida. Los demás procesos deben esperar hasta que el primero termine su operación, garantizando que no haya conflictos. El tiempo que cada proceso permanece en la región crítica es limitado.
- **Condición de sincronización:** Los procesos concurrentes a menudo necesitan coordinarse entre sí para funcionar correctamente. Esto sucede cuando un proceso debe esperar a que otro alcance cierto punto en su ejecución antes de continuar. Para gestionar esto, la programación concurrente proporciona mecanismos que permiten bloquear procesos hasta que ocurra un evento específico y luego desbloquearlos cuando ese evento suceda, asegurando la correcta sincronización entre ellos.