

Sebastian Amaya Perez

1000442057

Concurrencia e Hilos

¿Qué es la concurrencia?

La concurrencia es la capacidad de un sistema para manejar múltiples tareas a la vez, donde estas tareas pueden comenzar, ejecutarse y completarse en diferentes momentos. En un sistema operativo, esto permite que múltiples procesos o hilos compartan recursos y CPU de manera eficiente.

¿Qué es un hilo (thread)?

Un hilo es la unidad más pequeña de procesamiento que puede ser programada. Es parte de un proceso y comparte con otros hilos del mismo proceso recursos como memoria y archivos abiertos. Los hilos permiten dividir un proceso en múltiples unidades de ejecución que pueden correr en paralelo.

Funcionamiento de los hilos

Los hilos se crean usando APIs como pthread en C. El ciclo básico incluye:

- Creación con pthread_create
- Finalización con pthread_exit
- Espera con pthread_join
- Compartición de recursos dentro del mismo proceso

Puntos clave

- Mejora el rendimiento en programas con tareas concurrentes.
- Reduce la sobrecarga en comparación con procesos.
- Puede provocar errores como condiciones de carrera si no se sincronizan correctamente.

Locks

¿Qué es un lock?

Un lock o cerrojo es una herramienta de sincronización que garantiza que solo un hilo acceda a una sección crítica del código a la vez.

Funcionamiento

1. Un hilo intenta adquirir el lock antes de entrar en la sección crítica.
2. Si el lock está libre, lo adquiere; si no, espera.
3. Tras finalizar, libera el lock.

Tipos

- Mutex (mutual exclusion): el más común.
- Spin-locks: el hilo espera activamente hasta adquirir el lock.

Problemas y soluciones

- **Deadlocks:** dos o más hilos esperan indefinidamente por recursos.
 - **Livelock:** los hilos cambian de estado, pero no progresan.
-

Variables de condición

¿Qué son?

Permiten a los hilos esperar hasta que una condición específica sea verdadera.

Funcionamiento

- Usadas junto con un mutex.
- El hilo libera el mutex y espera (wait).
- Otro hilo modifica el estado y usa signal o broadcast.

Puntos clave

- Son útiles para coordinar productores y consumidores.
 - Se requiere siempre verificar la condición con while antes de continuar.
-

Semáforos

¿Qué es un semáforo?

Un semáforo es una variable entera que controla el acceso a recursos compartidos mediante dos operaciones atómicas:

- wait (P) o down: decrementa el valor; si es menor que cero, el proceso se bloquea.
- signal (V) o up: incrementa el valor; si hay procesos bloqueados, uno se desbloquea.

Tipos

- Binario (valor 0 o 1): similar a un lock.
- Contador: permite acceso a múltiples instancias del recurso.

Aplicaciones

- Problema productor-consumidor.
 - Límite de recursos (como número de conexiones).
-

Problemas de Concurrency

Errores comunes

- **Condiciones de carrera:** cuando múltiples hilos acceden a recursos sin sincronización.
- **Violaciones de atomicidad:** operaciones que deberían ser atómicas pero no lo son.
- **Interferencia de instrucciones:** orden incorrecto de ejecución.

Prevención

- Uso de locks y semáforos.
 - Diseño cuidadoso del orden de adquisición de recursos.
 - Técnicas como lock-ordering, timeout, y detección de deadlocks.
-

Entrada/Salida (I/O)

¿Qué es?

La comunicación entre el sistema y dispositivos externos como teclado, pantalla o disco duro.

Métodos

- **Polling:** la CPU verifica constantemente si el dispositivo está listo.
- **Interrupciones:** el dispositivo interrumpe al CPU cuando está listo.
- **DMA (Direct Memory Access):** transfiere datos sin intervención del CPU.

Importancia

- Permite eficiencia y multitarea.
 - DMA reduce la carga de trabajo del CPU.
-

Discos Duros

Funcionamiento

- Usan platos giratorios y cabezales de lectura/escritura.
- El tiempo de acceso depende de:
 - Tiempo de búsqueda (mover cabezal)
 - Latencia rotacional
 - Tiempo de transferencia

Algoritmos de planificación

- FCFS
 - SSTF (Shortest Seek Time First)
 - SCAN y C-SCAN
-

RAIDs

¿Qué es?

Redundant Array of Independent Disks. Sistema que combina múltiples discos físicos para mejorar desempeño y tolerancia a fallos.

Niveles

- **RAID 0:** striping sin redundancia
- **RAID 1:** duplicación (mirroring)
- **RAID 5:** paridad distribuida
- **RAID 6:** doble paridad

Ventajas

- Mejora velocidad (RAID 0)
 - Redundancia (RAID 1, 5, 6)
-

Implementación de Sistemas de Archivos

Componentes

- **Superbloque:** información del sistema de archivos
- **Inodos:** metadatos de archivos
- **Bloques de datos:** contenido de los archivos

Acciones

- Crear, leer, escribir, borrar archivos
- Uso de directorios e inodos con punteros a bloques

Crash consistency

- Técnicas como journaling o Copy-On-Write aseguran integridad ante fallos.
- Herramientas como fsck reparan inconsistencias.