

Arquitectura de Computadores Paralelos y Distribuidos

Tema 3: Apartado 2 - Paquete 3

Realizado por:
Jesús Gálvez Guerrero
Alejandro González Cardeñosa

Índice

1. Exclusión Mútua
 - a. Herramientas para la exclusión mutua
 - b. Problemas a la hora de la exclusión mutua
2. Instrucciones MRW
 - a. LL/CS
 - b. TT/TTS

Exclusión Mútua

Son algoritmos usados en programación concurrente para evitar problemas de incongruencia de los recursos a la hora de que más de una tarea deseen acceder a un mismo recurso.

Estas zonas de acceso a los recursos compartidos en la programación concurrente se le llama la sección crítica.


Algunos de estos recursos pueden ser:

- Colas
- Variables
- Datos vinculados a interrupciones

Herramientas para la exclusión mutua

Algunas de las herramientas y/o algoritmos más básicos para el control de la exclusión mutua son:

- Algoritmo de Dekker: Se basa en un sistema de turnos en el cual cuando ambos procesos quieren acceder a la sección crítica se decide cual entra en función de ese turno.
- Algoritmo de Peterson: La versión básica es una solución simplificada del algoritmo de Dekker, también usa un sistema de turnos, pero a diferencia del anterior no se asigna de antemano, sino que se van dando el turno entre ellos.

- 
- **Semáforos:** Son un mecanismo de sincronización de procesos que permiten al programador asistir al planificador del sistema operativo en su toma de decisiones de manera que permite sincronizar dos o más procesos, también permiten limitar el número de accesos simultáneos a un recurso compartido.
 - **Monitores:** Son objetos que encapsulan los recursos o variables compartidas y ofrecen una serie de métodos que permiten trabajar con esos recursos, garantizando la exclusión mutua al trabajar con ellos.
 - **Cerrosos:** Son mecanismos de sincronización los cuales tienen suelen tener 2 estados(bloqueado o desbloqueado) y que permiten bloquear o permitir la ejecución del código de la región crítica de los procesos concurrentes

Problemas a la hora de la exclusión mutua

Si no se controla bien el tema de la sincronización de las tareas y de la exclusión mutua se pueden llegar a ciertos estados del programa que son problemáticos:

- Interbloqueos: Se producen cuando las distintas tareas están esperando a que se les devuelvan el control del programa pero al estar todas así ninguna avanza en su ejecución.
- Inanición: Cuando una tarea no se le permite el acceso a la sección crítica aunque esta misma esté libre.
- Control de prioridades: Cuando una tarea de una prioridad menos es cortada por que un proceso de mayor prioridad requiere de los recursos del equipo, pero la tarea de menor prioridad no libera la exclusión mutua y el proceso de mayor prioridad no puede acceder por ello a la sección crítica

Instrucciones MRW

Son instrucciones de lectura, modificación y escritura atómicas implementadas a nivel de hardware.

A partir de ellas se puede implementar la sincronización a nivel de software.

Algunas de las instrucciones MRW básicas son:

- LL/SC
- TS/TTS

Instrucciones LL y SC

Son un par de instrucciones de carga y almacenamiento que se ejecutan juntas.

LL(Load Linked): Carga de memoria de forma normal y coloca un flag de monitorización de la caché.

SC(Store Conditional): Comprueba el estado del flag, si ha habido alguna interrupción/cambio de contexto que haya podido afectar al valor leído por LL, se cancela la operación, si no, se guarda el valor.

Instrucciones TS y TTS

TS(Test and Set): Comprueba el valor de una posición de memoria(cerrojo), si está a 0(disponible), devuelve el valor(para indicar que ha adquirido el cerrojo)y lo coloca a 1(no disponible).

TTS(Test and Test and Set):Consiste en esperar a que el valor del cerrojo esté a 0 antes de llamar al Test and Set.

Bibliografía

- [https://es.wikipedia.org/wiki/Exclusi%C3%B3n_mutua_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Exclusi%C3%B3n_mutua_(inform%C3%A1tica))
- <https://compas.cs.stonybrook.edu/~nhonarmand/courses/fa15/cse610/slides/05-synch.pdf>
- <https://disco.ethz.ch/courses/fs15/podc/lecture/chapter5.pdf>
- <https://en.wikipedia.org/wiki/Test-and-set>
- https://en.wikipedia.org/wiki/Test_and_test-and-set
- https://es.wikipedia.org/wiki/Algoritmo_de_Dekker
- https://es.wikipedia.org/wiki/Algoritmo_de_Peterson
- <https://1984.lsi.us.es/wiki-ssoo/index.php/Sem%C3%A1foros>
- https://www.ctr.unican.es/asignaturas/procodis_3_II/Doc/Procodis_2_05.pdf
- <https://es.slideshare.net/delapuerta/programacin-concurrente-datos-compartidos-y-exclusin-mutua>