

REPUBLICA BOLIVARIANA DE VENEZUELA
MINISTERIO DEL PODER POPULAR PARA LA EDUCACION CIENCIA Y
TECNOLOGIA
UNIVERSIDAD POLITÉCNICA TERRITORIAL ALONSO GAMERO
PROGRAMA NACIONAL DE FORMACIÓN EN INGENIERÍA INFORMÁTICA
TRAYECTO III TRIMESTRE I
SECCIÓN 01

COORDINACIÓN Y SINCRONIZACIÓN DE PROCESOS

TÉCNICO SUPERIOR UNIVERSITARIO:

KATHERINE HERNÁNDEZ

C.I V-26.656.307

SANTA ANA DE CORO, ABRIL DE 2024

Coordinación de procesos

La coordinación y sincronización de procesos se refiere a la manera en que los procesos en un sistema operativo se comunican entre sí y se organizan para trabajar de manera eficiente y sin conflictos. Esto es importante para garantizar que los procesos se ejecuten de manera ordenada y que no interfieran entre sí.

Algunas técnicas comunes para la coordinación y sincronización de procesos incluyen el uso de semáforos, mutex, monitores y variables de condición. Estas herramientas permiten a los procesos comunicarse entre sí, compartir recursos de manera segura y evitar condiciones de carrera y otros problemas de concurrencia.

Sincronización

Es la coordinación y cooperación de un conjunto de procesos para asegurar la comparación de recursos de cómputo. Los diferentes sistemas operativos proporcionan mecanismos para la sincronización de procesos. Algunos ejemplos comunes incluyen:

- Subprocesos POSIX: proporciona primitivas de sincronización de subprocesos, como mutex, variables de condición y barreras. Estas primitivas permiten que los subprocesos sincronicen sus acciones y coordinen el acceso a recursos compartidos.
- Java: ofrece funciones de sincronización integradas, incluidos bloques sincronizados y el paquete `java.util.concurrent`. Estas características permiten a los desarrolladores sincronizar el acceso a recursos compartidos en programas Java.
- Windows: admite la sincronización a través de primitivas como secciones críticas, eventos y semáforos. Estos mecanismos de sincronización permiten que los procesos y subprocesos de los sistemas basados en Windows coordinen sus acciones y accedan a recursos compartidos de forma segura.

Concurrencia

Dos o más procesos decimos que son concurrentes, paralelos, o que se ejecutan concurrentemente, cuando son procesados al mismo tiempo, es decir, que para ejecutar uno de ellos, no hace falta que se haya ejecutado otro.

Condiciones de Concurrencia

- Debe darse un conjunto de condiciones para que se puedan ejecutar varios procesos a la vez.
- Un conjunto de lectura $R(S_i)$ de la sentencia S_i es aquel formado por todas las variables que son referenciadas por la sentencia S_i durante su ejecución sin sufrir cambios.
- Un conjunto de escritura $W(S_i)$ de la sentencia S_i es aquel formado por todas las variables cuyos valores son modificados durante su ejecución.

- Existen diversas notaciones para especificar actividades concurrentes. Entre ellas, las instrucciones fork-join (no estructurados) y cobegin-coend (estructurados).
- Un proceso es independiente si no puede afectar o ser afectado por otros procesos corriendo en el sistema. Un proceso es interactuante si puede afectar o ser afectado por otros procesos

Semáforos

Los semáforos son objetos de sincronización que mantienen un recuento y permiten o restringen el acceso a los recursos según el valor del recuento. La gestión exitosa de recursos compartidos y la ejecución coordinada de procesos a menudo requieren herramientas confiables como los semáforos. Estos mecanismos flexibles permiten controlar diversos requisitos de recursos, ya sean binarios (0 o 1) o no binarios (mayor que 1). Por último. Este enfoque garantiza el uso eficiente de los recursos disponibles sin comprometer otros procesos vitales.

Monitores

Los monitores son construcciones de sincronización de nivel superior que encapsulan datos compartidos y los procedimientos que operan en ellos. Garantizan que sólo un proceso pueda ejecutar un procedimiento dentro del monitor en un momento dado, evitando el acceso simultáneo a datos compartidos. Los monitores proporcionan una forma estructurada y controlada de sincronizar procesos concurrentes, a menudo utilizando variables de condición para gestionar la coordinación de procesos.

Características

- Las variables de datos locales están sólo accesibles para los procedimientos del monitor y no para procedimientos externos.
- Un proceso entra en el monitor invocando a uno de sus procedimientos.
- Sólo un proceso puede estar ejecutando en el monitor en un instante dado; cualquier otro proceso que haya invocado al monitor quedará suspendido mientras espera que el monitor esté disponible.

Sección crítica y bloqueos

Identificar y priorizar procesos críticos es esencial para garantizar un rendimiento óptimo del sistema. Los procesos críticos pueden ser aquellos que requieren una respuesta rápida o que realizan tareas esenciales para el funcionamiento del sistema. Al asignar prioridades adecuadas, el sistema operativo puede asegurarse de que los procesos críticos se ejecuten de manera oportuna.

Bloqueos

- Condición de exclusión mutua: Existencia al menos de un recurso compartido por los procesos, al cual sólo puede acceder uno simultáneamente.
- Condición de posesión y espera: Al menos un proceso P_i ha adquirido un recurso R_i , y lo mantiene mientras espera al menos un recurso R_j que ya ha sido asignado a otro proceso.
- Condición de no expropiación: Los recursos no pueden ser apropiados por los procesos, es decir, los recursos sólo podrán ser liberados voluntariamente por sus propietarios.
- Condición de espera circular: Dado el conjunto de procesos $P_0 \dots P_n$, P_0 está esperando un recurso adquirido por P_1 , que está esperando un recurso adquirido por P_2, \dots , que está esperando un recurso adquirido por P_n , que está esperando un recurso adquirido por P_0