

Ejercicio 2 de procesos y memoria (septiembre 2000)

Enunciado

Sea un sistema que usa un algoritmo de planificación de procesos basado en prioridades con carácter expulsivo y que proporciona un mecanismo de comunicación asíncrono (envío no bloqueante pero recepción bloqueante). Se pretende analizar la traza de ejecución de dos procesos que usan un semáforo S con un valor inicial igual a 0 y tienen el siguiente perfil de ejecución:

Proceso P1 (alta prioridad)

1. Uso del procesador
2. DOWN(S)
3. Uso del procesador
4. SEND(Proceso 2, mensaje)
5. Uso del procesador
6. Primer fallo de página sobre una determinada página de la región de código. Hay marcos libres.
7. Uso del procesador
8. Primer fallo de página sobre una determinada página de la región de datos con valor inicial. No hay marcos libres y el marco elegido para el reemplazo está modificado.
9. Uso del procesador
10. Primer fallo de página sobre una determinada página de la región de datos sin valor inicial. No hay marcos libres pero el marco elegido para el reemplazo no está modificado.
11. Uso del procesador
12. Fallo de página por escritura en una página marcada como *copy-on-write*. Hay marcos libres.
13. Uso del procesador
14. Fin del proceso

Proceso P2 (baja prioridad)

1. Uso del procesador
2. UP(S)
3. Uso del procesador durante un intervalo de tiempo muy pequeño
4. RECEIVE(mensaje)
5. Uso del procesador durante un tiempo ilimitado

Se pide:

1. Mostrar la traza de ejecución de los dos procesos especificando en qué momentos se activa el sistema operativo, el motivo de dicha activación (llamada al sistema, interrupción de un dispositivo o excepción) y, de manera breve, la labor que realiza el sistema operativo durante esta activación.
2. Analizar qué ocurriría con la ejecución de estos dos procesos si el semáforo estuviera iniciado con un valor igual a 1, el proceso 2 realizara un DOWN en vez de un UP y el proceso 1 tuviera menos prioridad que el 2. ¿Qué podría hacer el sistema operativo ante esta situación?

Solución

1. A continuación, se muestra la traza de ejecución:
 - Empieza ejecutando P1 ya que tiene mayor prioridad.
 - Se produce una llamada DOWN que, al encontrar el semáforo cerrado, bloquea al proceso llamando al

planificador que selecciona P2. Se trata, por tanto, de un cambio de contexto voluntario.

- Ejecuta P2 hasta que realiza la llamada `UP`. Esta llamada desbloquea a P1 que, al tener mayor prioridad, pasa a ejecutar produciéndose un cambio de contexto involuntario.
- Ejecuta P1 y realiza la llamada `SEND`. Puesto que se trata de un envío asíncrono, el proceso no se bloquea, simplemente se copia el mensaje en un buffer del sistema.
- Sigue ejecutando P1 y se produce un fallo de página. El sistema operativo se activa debido a la excepción de fallo de página. La rutina de tratamiento encuentra un marco libre e inicia una operación en el disco para leer la página de código sobre el marco seleccionado. P1 resulta bloqueado y el planificador selecciona a P2. Se trata de un cambio de contexto voluntario, ya que el proceso no puede seguir.
- P2 ejecuta brevemente y realiza una llamada al sistema `RECEIVE`. Dado que hay un mensaje encolado, el proceso no se bloquea continuando su ejecución.
- Mientras está ejecutando P2, se produce una interrupción del disco que indica que ha terminado la operación de lectura. En la rutina de tratamiento se desbloquea a P1 que, al tener mayor prioridad, pasa a ejecutar produciéndose un cambio de contexto involuntario.
- P1 ejecuta y produce un segundo fallo de página. El sistema operativo se activa debido a la excepción de fallo de página. La rutina de tratamiento no encuentra un marco libre, selecciona una página para expulsar y, como está modificada, inicia una operación en el disco para escribir la página en el *swap*. P1 resulta bloqueado (dentro de la rutina del tratamiento del fallo de página) y el planificador selecciona a P2. Se trata de un cambio de contexto voluntario, ya que el proceso no puede seguir.
- Mientras está ejecutando P2, se produce una interrupción del disco que indica que ha terminado la operación de escritura que ha liberado un marco. La rutina de tratamiento de fallo de página asociada a P1 continúa desde el punto en el que se había quedado, iniciando una nueva operación de lectura en el disco de la página de datos con valor inicial requerida. El proceso P1, que ejecuta en modo sistema, vuelve a quedar bloqueado dentro de la rutina de tratamiento del fallo, seleccionándose de nuevo P2 para ejecutar.
- P2 sigue ejecutando hasta que llega la interrupción del disco. Esta indica que ha terminado la operación de lectura. En la rutina de tratamiento se desbloquea a P1 que, al tener mayor prioridad, pasa a ejecutar produciéndose un cambio de contexto involuntario.
- La ejecución de P1 causa un nuevo fallo de página. Como no hay marcos libres, se aplica el algoritmo de reemplazo que encuentra un marco no modificado. No se requiere, por tanto, una escritura al disco. Como se trata de una página de la región de datos sin valor inicial, tampoco se requiere leerla del disco. Simplemente, se rellena con ceros el marco seleccionado. No hay cambios de contexto.
- P1 continúa su ejecución y produce un fallo de página asociado a una página marcada como *copy-on-write*. La rutina encuentra un marco libre y realiza en él una copia de la página original. De nuevo, no hay cambios de contexto.
- P1 sigue ejecutando hasta que realiza una llamada al sistema (`EXIT`) para indicar que ha terminado su ejecución.
- El sistema operativo trata la llamada `EXIT` liberando los recursos del proceso y activando el planificador que selecciona a P2.
- P2 continúa ejecutando indefinidamente.

2. A continuación, se muestra la traza de ejecución siguiendo los nuevos supuestos:

- Empieza ejecutando P2 ya que tiene mayor prioridad.
- Se produce una llamada `DOWN` que, al encontrar el semáforo abierto, simplemente actualiza su valor y termina, no produciéndose ningún cambio de contexto.
- P2 ejecuta brevemente y realiza una llamada al sistema `RECEIVE`. Dado que no hay ningún mensaje, P2 se bloquea llamando al planificador que selecciona P1. Se trata, por tanto, de un cambio de contexto voluntario.
- P1 ejecuta y realiza una llamada `DOWN` que, al encontrar el semáforo cerrado, bloquea al proceso llamando al planificador. Puesto que no hay ningún proceso listo, el sistema se quedaría en ese estado indefinidamente.

Se ha producido una situación de interbloqueo debido al uso *conflictivo* de los recursos por parte de los procesos. Para tratar este tipo de situaciones, el sistema operativo podría usar una de las técnicas clásicas de tratamiento de interbloqueos (detección y recuperación, predicción o prevención= o simplemente ignorarlo).