

Laboratorio de Sistemas Operativos

Parcial Final

Fecha de entrega: jueves 27 de septiembre de 2007

Memoria Virtual

Su tarea consistirá en simular el funcionamiento de un sistema operativo que utiliza paginación bajo demanda con tablas de páginas de un nivel, y que hace uso del algoritmo de reemplazo de páginas FIFO en caso de que ocurra un fallo de página y la memoria RAM se encuentre llena (Ver Tanenbaum, Andrew. Sistemas Operativos Modernos. Segunda Edición).

Al utilizar la paginación, el espacio de memoria del proceso se divide en páginas que se van cargando a medida que se requieran (paginación por demanda). Inicialmente sólo la página que contiene el punto de entrada (*entry*) del proceso se carga en memoria (Por simplicidad asuma que *entry* se encuentra siempre en la primera página del proceso).

A medida que el proceso realiza solicitudes de memoria (en su espacio de direcciones) se verifica que la página correspondiente se encuentre cargada en memoria. Si la página no está cargada en memoria, ocurre un fallo de página, que obliga al sistema operativo a ubicar la página ausente en un marco de página en RAM y asociarle una página en el espacio de direcciones virtual. Muy pronto la memoria RAM se llena con páginas de diferentes procesos, y es necesario elegir algún marco para reemplazarlo con la página ausente. En este momento se debe utilizar el algoritmo de reemplazo de páginas (FIFO) para elegir cual es la página que se va a reemplazar.

Entrada del programa

Los datos del simulador se leerán por entrada estándar, y consistirán en una serie de comandos (TYPE VIRTUAL, PHYSICAL, PAGESIZE y LOAD), que indican el tamaño de la memoria virtual, de la memoria física, el tamaño de cada página y el archivo en el que se encuentran las secuencias de solicitudes de memoria. Estos comandos se explican a continuación.

Comando	Descripción	Ejemplo
VIRTUAL	Define el tamaño de la memoria virtual, en bytes. Se puede suponer que este tamaño es mayor o igual al tamaño de la memoria física	VIRTUAL 65536
PHYSICAL	Define el tamaño de la memoria física, en bytes.	PHYSICAL 32768
PAGESIZE	Define el tamaño de una página en memoria	PAGESIZE 4096
LOAD	Define el nombre del archivo en el cual se encuentran las secuencias de las solicitudes de memoria que se van a simular. Luego de cargar el archivo se deberá iniciar la simulación.	LOAD simulation1.mem

Por simplicidad de la simulación, se puede suponer que el sistema operativo se encuentra almacenado en una ROM, es decir que toda la memoria se encuentra disponible para los procesos.

Solicitudes de memoria

Un archivo de secuencias de solicitudes de memoria consiste en una serie de líneas, cada una de las cuales representa una solicitud de acceso a memoria realizada por un proceso en su espacio de direcciones de memoria. A continuación se presenta un ejemplo de archivo de secuencias, para el cual se ha definido una memoria virtual (VIRTUAL) de 64K, una memoria física de (PHYSICAL) 32K y un tamaño de página (PAGESIZE) de 4K. Para cada línea se presenta una pequeña explicación del estado del sistema.

Se puede asumir que la primera vez que un proceso realiza una solicitud de memoria no genera un fallo de página, siempre y cuando la dirección solicitada se encuentre en su primera página. En este caso sólo se deberá cargar la primera página del proceso en RAM y se deberá asignar una página en el espacio de direcciones virtual. Si no hay espacio para cargar la primera página del proceso, se deberá aplicar FIFO para encontrar la página virtual a reemplazar.

Línea	Significado
A 1200	Primera solicitud de A: el proceso solicita acceso a la dirección de memoria 1200 dentro de su espacio de direcciones. Debido a que 1200 se encuentra en su primera página (de 4K), no ocurre un fallo de página. El SO deberá cargar la primera página del proceso en RAM y asignar una página dentro del espacio de direcciones virtual de 64k (potencialmente la primera página virtual y el primer marco de página, debido a que es el primer proceso en ejecución)
B 800	Igual que la solicitud anterior. No se genera fallo de página porque la dirección se encuentra en la primera página del proceso. La primera página del proceso B se asignará potencialmente al segundo marco de página en RAM y a la segunda página en el espacio de direcciones virtual.
C 5190	Se deben cargar dos páginas del proceso C: Su primera página (que contiene el entry) no genera fallo de página, pero la segunda (la página 1 del proceso, en la cual se encuentra la dirección solicitada) sí genera fallo de página.
A 7200	Fallo de página: se deberá cargar la página correspondiente en el espacio de direcciones virtual.
B 100	No genera fallo de página.
D 17800	Fallo de página. Se deberán cargar las dos páginas (la página 0 y la página 4 del proceso)
...	(Otras solicitudes de acceso a memoria)
N 6780	Eventualmente un proceso "N" realiza su primera solicitud, y la memoria RAM se encontrará llena (todos los marcos de página se encuentran asignados). Al utilizar FIFO, la página que deberá ser reemplazada es la primera página del proceso A, que fue la primera que se cargó en memoria. Como es la primera solicitud del proceso "N" sólo se genera un fallo de página.

Salida del programa

Un ejemplo de salida del programa con los datos de ejemplo será el siguiente:

Solicitud	Página del proceso	Virtual	Física	Total FP	Reemplaza
A (1200)	0*	0	0	0	
B (800)	0*	1	1	0	
C (5190)	0**	2	2	0	
C (5190)	1	3	3	1	
A (7200)	1	4	4	2	
B (100)	0	1	1	2	
D (17800)	0**	5	5	2	
D (17800)	4	6	6	3	
...
...	(#)	
N (6780)	0**	0	0	(#)	0
N (6780)	1	1	1	(#+1)	1
A (600)	0	2	2	(#+2)	2
...

Total de fallos de página: (##)

Fin de la simulación

Como se mencionó anteriormente, se debe observar que cuando un proceso realiza su primera solicitud de memoria existen dos escenarios diferentes:

- La dirección de memoria se encuentra en la primera página del proceso (marcados con *): No ocurre fallo de página. Se carga la primera página del proceso. En caso que la memoria física se encuentre llena, se deberá aplicar FIFO para determinar cual es la página que se va a reemplazar.
- La dirección de memoria no se encuentra en la primera página (marcados con **): Se deberán cargar dos páginas del proceso: La primera página del proceso no genera fallo de página, y la página en la cual se encuentra la dirección solicitada sí genera un fallo de página. Potencialmente habrá la necesidad de reemplazar dos páginas (si la memoria física se encuentra llena)

En el ejemplo, cuando se cargan los procesos C y D ocurre un fallo de página pero no hay necesidad de reemplazar páginas porque aún hay marcos disponibles. En cambio, cuando se carga el proceso N ocurre un solo fallo de página y hay que reemplazar dos páginas debido a que todos los marcos están asignados. Al utilizar FIFO, se deberá reemplazar la primera página de A asignada en la página virtual 0, y la primera página de B asignada en la página virtual 1. Luego cuando A solicita la dirección 600 ocurre otro fallo de página, porque esta página había sido reemplazada anteriormente. Como todos los marcos están asignados, la página 0 de A deberá reemplazar una página (en este caso la página 0 del proceso C que estaba asignada a la página virtual 2).