UNIVERSIDAD YACHAY TECH | ESCUELA DE CIENCIAS MATEMÁTICAS Y COMPUTACIONALES

Simulador de Memoria

Sistemas Operativos (TA74) 1-2019

Fecha de Entrega: 12 de Abril 2019

1 Grupos de trabajo

Se organizarán grupos de hasta 3 estudiantes, según la lista anexa. El grupo para cada estudiante vendrá determinado por la fórmula:

$$g_i = int((i + 2)/3)$$

Donde:
i: El número del estudiante en la lista
g_i: Identificación del grupo de trabajo del estudiante i
int: Una función que devuelve la parte entera de un número real.

2 Descripción general

Este programa no requiere llamadas al sistema, por lo que debe ser completamente independiente del SO; además, podrá ser desarrollado en el lenguaje de programación que le resulte más apropiado.

Su programa leerá de un archivo de entrada, o por interfaz gráfica, los parámetros. Esto parámetros serán:

- Tamaño de página (512, 1024, 2048)
- Tamaño de la memoria en frames (16, 32, 64)
- El tamaño del quantum, para la planificación de los procesos usando RR. Suponiendo que el quantum representa el número de instrucciones del proceso que se ejecutan antes de seleccionar el proceso.
- Número de procesos en la simulación
- Una lista de nombres de archivos (un archivo por proceso). Se asumirá que el orden de llegada de los procesos será dado por la posición.

  En cada línea del archivo de un proceso se describen 2 operaciones a memoria, usando 4 (VALORES), separados por un espacio en blanco:

  1. El primero es un **número entero** que representa el ID de proceso.
  2. El segundo es un **número entero** que contiene la dirección de la instrucción para realizar la operación. Esta dirección siempre se refiere a un acceso de lectura.

3. El tercero es un **número entero** que representa la dirección del dato.
4. El cuarto es un **caracter** que representa el tipo de operación (R o W). Si este caracter es una W, la referencia de memoria al dato es una escritura (modificación).

Cada una de las líneas de un archivo de proceso se asumirá como una instrucción, para efectos de contabilizar el quantum.

Todas las palabras son de 16 bits, por lo que todos los procesos tienen un espacio de direcciones lógicas de 64K. Todas las direcciones leídas estarán en el rango 0 .. 65535.

Cada página tiene 2^n bytes ($n \in \{9, 10, 11\}$), por lo que cada proceso tiene una tabla de páginas de tamaño $\frac{2^{16}}{2^n} = 2^{16-n} = 2^k$. En otras palabras, en cada dirección lógica, los primeros (la más significativa) k bits determinan el *frame* de página, y los últimos n bits determinan el desplazamiento de la dirección dentro del *frame* de página. (Sugerencia: para obtener el número de *frame* de página de una dirección lógica particular, divida por 2^n).

La tabla de páginas debe tener un bit de referencia y un bit *dirty* para cada página. Es posible que desee incluir otros datos también.

Debe realizar un seguimiento de dos estadísticas, el número total de fallos de página (page faults) y el número total de referencias de disco. Cada error de página tiene al menos una referencia de disco, pero si la página que se va a reemplazar está *dirty*, habrá dos referencias de disco, una para copiar la página *dirty* de nuevo en el disco y otra para cargar la nueva página.

Cada proceso tiene su propia tabla de páginas, por lo que la dirección 12340 en el proceso 1 es diferente de la dirección 12340 en el proceso 2.

Para que este programa sea fácilmente calificable, los resultados deben ser deterministas.

3 Simulación

Su programa se ejecutará de la siguiente manera:

simulator datafile.txt version [1]

Donde se asume que **simulator** es el nombre del programa, **datafile.txt** contiene los parámetros descritos, **version** es un 1 o 2 para indicar el algoritmo de reemplazo y el último parámetro es opcional para indicar si el programa se ejecuta en modo "debug".

Para el modo debug, se debe tener una variable global **int debug**. Si el programa se ejecuta con algún argumento (si argc $>$ 3), entonces la depuración se establece en 1, de lo contrario, establecerlo en cero. Si la depuración está activada, su programa debería mostrar información sobre cada fallo de página cada vez que ocurra. Debe enumerar la línea que generó el fallo de la página, la página física que se está reemplazando, el número de proceso, la página lógica y si la página que se está reemplazando estaba *dirty* o no.

3.1 Versión 1

En la primera prueba, simulará un algoritmo "no utilizado recientemente". Cuando se hace referencia a una página, su bit de referencia se establece en 1. Después de ejecutar

cada 200 instrucciones, su programa debe establecer todos los bits de referencia en cero.

Cuando se produce un fallo de página, primero busque una página no utilizada (esto solo ocurrirá al principio de su simulación). Luego busque una página sin referencia donde el bit *dirty* esté desactivado y reemplace esta página (0,0). Luego busque una página sin referencia donde esté el bit *dirty* y reemplace esa página (0,1). Si no se encuentra ninguno, busque una página referenciada donde el bit *dirty* esté desactivado y reemplace esa página (1,0). Finalmente, tendrá que reemplazar una página a la que se hace referencia y está *dirty* (1,1).

Para que el resultado sea determinista, siempre se reemplaza la página con el número más bajo en una categoría particular.

3.2 Versión 2

La segunda versión usará un algoritmo de "uso menos reciente"; es decir, hace un seguimiento del ciclo en el que se hizo referencia a cada página por última vez, y reemplaza la página que se usó menos recientemente. Si hay dos páginas con el mismo valor, reemplace una que no esté sucia. Si ambas páginas o ninguna de ellas están sucias, reemplace la página con el número inferior.

4 Entregable

Cada grupo debe entregar un "Reporte Técnico" (archivo PDF). Este reporte debe incluir:

- Diseño. Estructuras de Datos (objetos), Algoritmos usados, etc.
- Construcción. Lenguaje de programación y bibliotecas usadas.
- Pruebas y Funcionamiento. Como se compila y ejecuta el simualdor.
- Localización (links) de los anexos digitales (Códigos y resultados de pruebas).

Este reporte no debe superar las 5 páginas.