

Simulador de algoritmos de sustitución de páginas

Juan José Domínguez[†], Antonia Estero[†], Félix Velázquez

[†]Dept. de Lenguajes y Sistemas Informáticos
Universidad de Cádiz
11003 Cádiz

e-mail: juanjose.dominguez@uca.es, antonia.estero@uca.es

Resumen

Se presenta una herramienta que puede ser empleada en la docencia de Sistemas Operativos, concretamente en el campo de los algoritmos de sustitución de páginas, elaborada como una aplicación Web en Java.

1. Introducción

La enseñanza en cualquier materia tiene como fin el aprendizaje eficaz por parte del alumnado. Hoy en día el empleo de nuevas tecnologías, como multimedia, Internet, etc., nos permite elaborar nuevas técnicas de aprendizaje, en beneficio de una educación más extendida y de mayor calidad.

Podemos considerar tres tipos básicos de software educativo, según [1]:

- β *Programas tutoriales* en los que la estrategia pedagógica empleada es del tipo ejercicios.
- β *Tutores inteligentes*, similares a los anteriores, pero adaptan el nivel de enseñanza a la capacidad de aprendizaje del alumno.
- β *Simuladores* que crean un entorno simulado, que permite al alumno, mediante la exploración y la experimentación, adquirir y reafirmar sus conocimientos.

El empleo de estos últimos permite la sustitución de los métodos clásicos de enseñanza, basados en la explicación con ayuda de la pizarra y las transparencias, por otros en los que el alumno puede tener mayor participación permitiendo el desarrollo de las habilidades cognitivas de éste.

2. El simulador de algoritmos de sustitución de páginas

El *simulador de algoritmos de sustitución de páginas* [2] tiene como objetivo mostrar el funcionamiento de los aspectos de la gestión de memoria que intervienen en la ejecución de los procesos en un sistema de multiprogramación. En concreto, se describe el funcionamiento de diversos algoritmos de sustitución, el esquema de traducción de direcciones lógicas a físicas, así como las estructuras de datos implicadas en un sistema de paginación.

Esta herramienta está elaborada en Java y se ha desarrollado como una aplicación Web, lo que permite ejecutarla desde cualquier ordenador que posea un navegador de Internet, de forma independiente del sistema operativo que ejecute.

El simulador posee diversos modos de ejecución, permitiendo la simulación de un algoritmo de sustitución, la realización de un ejercicio o bien la visualización de gráficas comparativas entre distintos algoritmos.

3. Simulación de un algoritmo

Si se desea analizar el comportamiento de un algoritmo es necesario proporcionar al simulador los distintos parámetros de configuración del sistema. Éstos son:

- β *Tamaño del bloque*: especifica el tamaño de la página, que puede seleccionarse entre 1, 2 y 4 KiB¹.

¹ Las unidades empleadas siguen las recomendaciones de la IEC [3].

- β *Número de marcos* en el sistema, que puede escogerse entre un rango de valores de 3 a 6.
- β *Número de procesos* a simular, entre 2 y 4.
- β *Algoritmo de sustitución*, que puede ser el Óptimo, LRU, Reloj, Reloj mejorado, LFU, MFU, Conjunto de trabajo o Frecuencia de fallos de página.
- β *Esquema de Asignación y Alcance*, pudiendo escogerse entre alcance local o global y asignación fija o variable.
- β *Política de carga*, que puede ser paginación por demanda o prepaginación. En el caso de escoger ésta, puede realizarse cada vez que el proceso toma el control de la CPU, o bien cada vez que se produce un fallo de página. También puede elegirse el porcentaje del tamaño del proceso que hay que prepaginar.

Una vez especificados los parámetros del sistema se procede a generar la cadena de referencias que se desea simular; cuya longitud puede oscilar entre 25 y 40. Con el botón de *Generar ref.* se procede a la generación aleatoria de dichas referencias a páginas. El formato de éstas es una dirección lógica junto con el proceso que la realiza, y el modo de acceso. Así, *4550PIE* significa la referencia 4550 del proceso P1 y modo escritura.

Para iniciar la simulación se pulsa el botón *Empezar*. A partir de este momento se puede observar como evoluciona ésta a través de diferentes zonas de la pantalla. La figura 1 muestra una instantánea de la simulación de un algoritmo LRU.

En la parte superior se muestra la cadena de referencias, señalándose en color verde la referencia que se va a procesar.

En la ventana central aparecen las estructuras de control del sistema, es decir, la tabla de páginas de cada proceso, así como la tabla de marcos. En las primeras se muestra el bit de presencia así como el marco donde se encuentra la página. En la tabla de marcos, aparece la página que lo ocupa y el proceso al que pertenece, así como los bits de referencia y modificación. Durante la simulación se nos mostrará mediante colores cuando se producen los fallos de páginas, así como las modificaciones que sufren las estructuras de control correspondientes.

En la parte inferior se indica el proceso de traducción de la dirección lógica a física, así como

un sumario de los fallos de páginas que se han cometido y las operaciones de E/S realizadas. Además, se visualizan las estructuras adicionales que necesita cada algoritmo. Así, en el caso del algoritmo LRU, podremos ver en cada instante el estado de la pila de páginas, señalando la página candidata.

Para la realización de la simulación, el usuario dispone de las opciones *Paso Adelante* y *Paso Atrás*, donde puede especificar un valor entre 1 y 5, para activar la simulación. Igualmente con el botón *Terminar* puede dar por finalizada la ejecución de ésta.

4. Realización de ejercicios

Con el simulador también es posible la realización de ejercicios. Éstos consisten en indicar el estado de los distintos marcos, así como las páginas que entran y salen en los fallos de página, según el algoritmo de sustitución elegido, para una cadena de referencias a memoria.

Para la realización del ejercicio es necesario dar los siguientes parámetros:

- β *Algoritmo de sustitución* que se va a emplear: Óptimo, LRU, LFU, MFU, Reloj, y Reloj mejorado.
- β *Número de referencias a memoria*, un valor entre 9 y 14.
- β *Número de marcos* del sistema, entre 3 y 9.

Una vez que se suministran los datos básicos de configuración se pulsa el botón *Empezar*, lo que produce la aparición de la cuadrícula que nos permite realizar el ejercicio. La figura 2 muestra una instantánea de éste.

A continuación se debe proporcionar la cadena de referencias a memoria, cuyo formato es número de página y el modo de acceso (L para lectura y E para escritura). Esta cadena se puede generar aleatoriamente (pulsando el botón *Generar*) o bien especificar una concreta.

Una vez tengamos la cadena de referencias, el usuario debe proceder a indicar en cada paso el estado de la memoria, así como las páginas que entran y salen en cada fallo de página.

Realizado el ejercicio, con el botón *Corregir* el simulador procede a su corrección, indicando si está o no correcto, y en este último caso dónde se encuentra el primer error. Igualmente, con el

botón *Resolver* podemos hacer que el simulador resuelva el ejercicio de forma automática.

5. Gráficas comparativas

Finalmente, el simulador permite realizar comparativas entre los distintos algoritmos de sustitución de páginas mediante gráficas. Se pueden comparar el número de fallos de página que produce un algoritmo determinado a lo largo del tiempo, así como el número de fallos de página en función del número de marcos.

5.1. Fallos de página frente al tiempo

Una comparativa que resulta útil para ver la eficiencia de los distintos algoritmos es establecer los fallos de página que se producen a lo largo del tiempo para una cadena de referencias lo suficientemente grande y un número de marcos fijo.

En este caso, es necesario indicar al simulador la longitud de la cadena de referencias, que puede ser 250, 500 ó 750.

Los distintos algoritmos que se pueden comparar aparecen a la derecha de la ventana, de forma que al seleccionar uno aparece su representación en la gráfica. La figura 3 muestra una instantánea de ésta, en la que se comparan los algoritmos Óptimo, LRU, Reloj y FIFO.

Con el botón *Regenerar referencias* el simulador produce una nueva cadena de referencias de la longitud indicada y actualiza las gráficas de los algoritmos previamente seleccionados.

La utilidad de esta gráfica es que podemos visualizar de forma rápida la evolución en el tiempo de los distintos algoritmos para un número alto de referencias, lo que permite la experimentación y comprobación de resultados teóricos por parte del usuario.

5.2. Fallos de página frente al n° de marcos

El otro tipo de comparativa que permite realizar el simulador es el número de fallos de páginas frente al número de marcos del que dispone el proceso.

En este caso el simulador necesita que el usuario proporcione una cadena de referencias, de

12 a 15 valores, la cuál se puede generar de forma automática mediante el botón *Generar referencias*.

Al igual que en la otra gráfica comparativa, en el lado derecho nos aparecen los distintos algoritmos que se pueden comparar, de forma que al seleccionarlos se nos muestra su representación.

La utilidad principal de esta comparación es la de visualizar cuándo un algoritmo presenta la *anomalía de Belady*.

5. Conclusión

Se ha presentado una herramienta elaborada para mejorar la docencia en la asignatura de *Sistemas Operativos*. Ésta permite que el alumno sea un elemento más activo dentro del proceso de aprendizaje de la materia, permitiéndole en todo momento:

- β Realizar ejercicios, adaptando el nivel de dificultad de éstos al grado de aprendizaje adquirido.
- β Realizar estudios estadísticos que permitan evaluar las situaciones donde conviene uno u otro algoritmo.
- β Ampliar el conocimiento proporcionado en la explicación de los algoritmos al disponer de una visión más global del alcance de éstos, al permitir visualizar las estructuras de datos implicadas.

Referencias

- [1] Antonio Vaquero Sánchez. *Informática y Educación*. Panorama Informático. Editado por Federación Española de Sociedades Informáticas. Pág. 65-96. 1996.
- [2] Félix Velázquez Salas. *Herramienta de apoyo a la docencia de Sistemas Operativos: Simulador de algoritmos de sustitución de páginas*. Proyecto Fin de Carrera de la Ingeniería Técnica en Informática de Gestión. Universidad de Cádiz. 1999.
- [3] International Electrotechnical Comisión. *Amendment 2 to IEC International Standard IEC 60027-2*, enero de 1999.

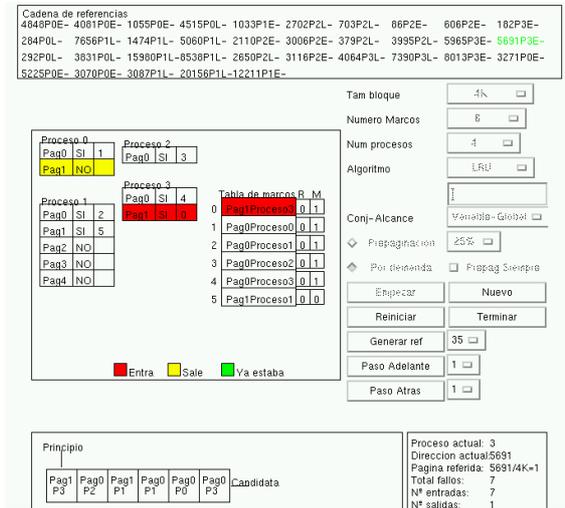


Figura 1. Simulación del algoritmo LRU

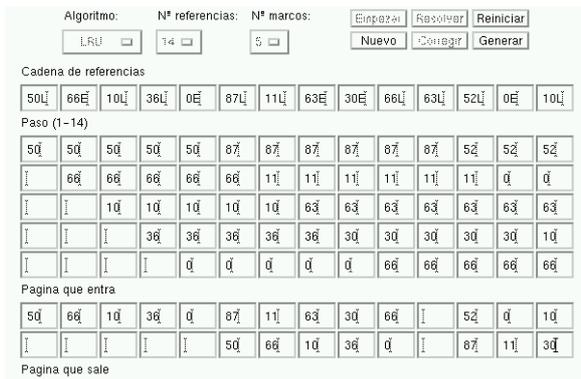


Figura 2. Realización de un ejercicio

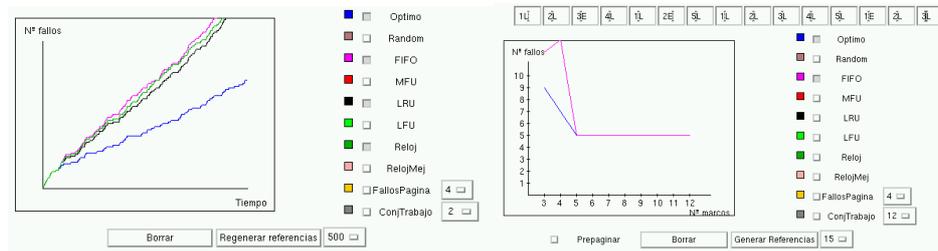


Figura 3. Fallos de página frente al tiempo

Figura 4. Fallos de página frente al n° de marcos