

Caso 2

Ángela Liliana Jiménez Flechas 201912941

Miguel Ángel Acosta Walteros 201914976

Estructuras de datos:

Manejamos una estructura de datos que representa la tabla de páginas, otra estructura que representa las páginas que se encuentran en memoria y finalmente un arreglo que contiene las referencias:

Tabla_paginas: Es un arreglo de enteros.

Cada entero (int) de este arreglo contiene a nivel de bits información que va a ser útil para el funcionamiento del programa. Es importante resaltar que no estamos usando el primer bit para evitar confusiones, porque en java un int es un entero de 32 de bits complemento a 2 (por lo que el primer bit representa el signo) dejándonos 31 bits disponibles para representar enteros sin signo.

La aclaración anterior explica que para nuestro desarrollo de la solución usemos las siguientes referencias representar los estados de una página:

Para decir que la página está llena y acaba de ser referenciada el 2do y 3er bit del entero van a ser 1's

(ejemplo: 01100000000000000000000000000000)

Para decir que la página está llena pero no acaba de ser referenciada el 2do bit será 1 y el 3er bit será 0

(ejemplo: 01000000000000000000000000000000)

Y para decir que la página está vacía el 2do y 3er bit van a ser 0's

(ejemplo: 00000000000000000000000000000000)

Por otro lado, el resto de los bits a la derecha del tercer bit nos van a representar el registro de referencias que se han hecho a esa página (registro que va a ser actualizado por el thread 2).

Por ejemplo si quisiéramos saber la información de la página 2 en determinado momento, preguntamos por la página en la tabla de páginas "Tabla_paginas[2]":

Ahora, Tabla_paginas[2] podría devolvernos este valor:

1614938240

Este valor representado en bits es:

01100000010000100000000010000000

Con 11 sabemos que la página está llena y acaba de ser referenciada, adicional a eso, la parte de la izquierda nos indica que la página fue referenciada 1 vez hace 8 pulsos, 1 vez hace 13 pulsos y 1 vez hace 23 pulsos, lo que en total nos dice que la página ha sido referenciada 3 veces.

Marcos de referencia: arreglo de enteros

Tenemos en el Thread 1 un arreglo de enteros que nos guarda las páginas que se encuentran actualmente en memoria, al ocurrir un fallo de página este arreglo se actualiza.

Referencias: arreglo de enteros

En la clase principal tenemos el arreglo de las referencias que entran por parámetro, se leen al comienzo del programa.

Esquema de sincronización:

Los 2 threads tienen como recurso compartido a la tabla de páginas, por lo que para hacer acceso correcto a esta tabla planteamos el siguiente esquema:

Para el thread 2, vamos a ejecutar un ciclo que termina cuando todas las referencias han sido revisadas, en cada ciclo lo primero que hacemos es obtener control exclusivo de la tabla de páginas para que nadie modifique la tabla mientras trabajamos en ella (tomar cada entero de la tabla de páginas, agregarle un bit si fue referenciado en el último pulso y realizar el corrimiento), luego liberamos la tabla y mandamos a dormido al thread por un segundo para simular el ciclo de reloj.

Para el thread 1, vamos a ejecutar un ciclo que termina cuando todas las referencias han sido revisadas. Inicialmente en cada ciclo obtenemos control exclusivo de la tabla de páginas para hacer modificaciones en ella, luego revisamos si ocurre un fallo de página (revisando si la página referencia está en nuestro arreglo marcos), de no ser así, se libera la tabla de páginas en ese momento y de ser así, tenemos que leer los valores del entero que nos dicen cuál es la página menos referenciada, vaciar esta página y llenar la página que originalmente fue referenciada, para ahí sí liberar la tabla de páginas, posterior a eso se manda a dormido el thread por 2 segundos para simular el acceso a la tabla de páginas cada 2 milisegundos.

Datos recopilados (archivos de prueba):

requerimientos1.txt:

Los datos son: 40 páginas, 8 marcos, 25,00 porcentaje de localidad
se detectaron:597 fallos de pagina

requerimientos2.txt:

Los datos son: 60 páginas, 8 marcos, 75,00 porcentaje de localidad
se detectaron:601 fallos de pagina

requerimientos3.txt:

Los datos son: 100 páginas, 8 marcos, 75,00 porcentaje de localidad
se detectaron:755 fallos de pagina

requerimientos4.txt:

Los datos son: 40 páginas, 16 marcos, 25,00 porcentaje de localidad
se detectaron:337 fallos de pagina

Tabla de datos conjuntos:

	R1	R2	R3	R4
Fallos de Página Obtenidos	597	601	755	337
Marcos de Página asignados	8	8	8	16
Páginas totales	40	60	100	40
Porcentaje de Localidad	25%	75%	75%	25%

Interpretación:

Primero comparamos Fallas vs número de marcos

Para esto usamos los archivos de prueba (creados por nosotros) referencias10, referencias11, referencias12, y referencias 13, en estos archivos dejamos fijo porcentaje de localidad en 50% y número de páginas en 100

Para referencias 10:

Los datos son: 100 páginas, 10 marcos, 0,50 porcentaje de localidad se detectaron:759 fallos de pagina

Para referencias 11:

Los datos son: 100 páginas, 15 marcos, 0,50 porcentaje de localidad se detectaron:674 fallos de pagina

Para referencias 12:

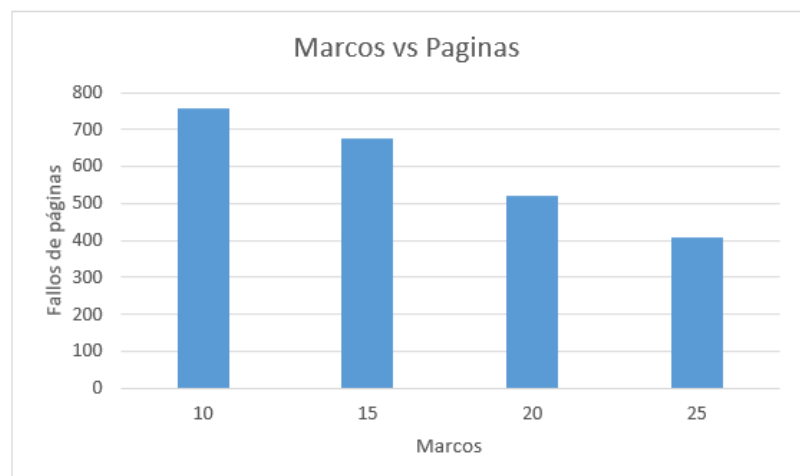
Los datos son: 100 páginas, 20 marcos, 0,50 porcentaje de localidad se detectaron:522 fallos de pagina

Para referencias 13:

Los datos son: 100 páginas, 25 marcos, 0,50 porcentaje de localidad se detectaron:408 fallos de pagina

Lo que nos da:

Marcos	Fallas
10	759
15	674
20	522
25	408



Esta gráfica nos indica que al aumentar el número de marcos vamos a encontrarnos con menor número de fallos de página, y esto tiene sentido porque al aumentar el número de marcos podemos tener más páginas con información en memoria llevando a que sea menos probable que llegue una referencia que no esté en memoria.

Luego comparamos Porcentaje de Localidad vs Fallos:

Para esto usamos los archivos de prueba (creados por nosotros) referencias16, referencias17, referencias18, y referencias 19, en estos archivos dejamos fijo porcentaje de localidad en marcos en 25 y número de páginas en 100

Para referencias 16:

Los datos son: 100 páginas, 25 marcos, 0,80 porcentaje de localidad se detectaron:204 fallos de pagina

Para referencias 17:

Los datos son: 100 páginas, 25 marcos, 0,60 porcentaje de localidad se detectaron:359 fallos de pagina

Para referencias 18:

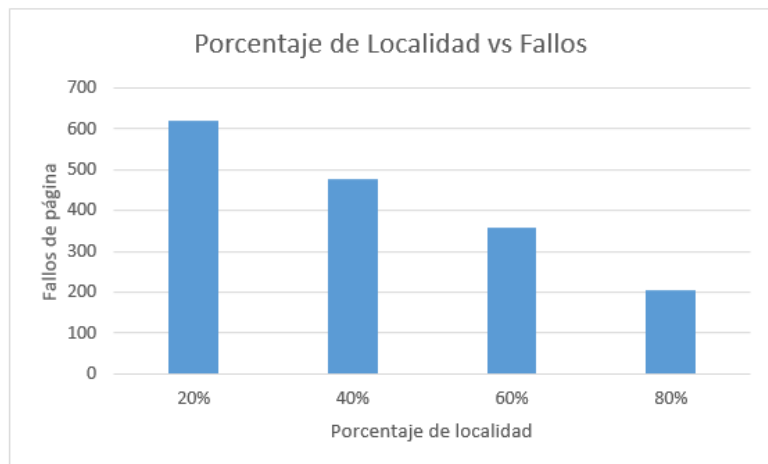
Los datos son: 100 páginas, 25 marcos, 0,40 porcentaje de localidad se detectaron:476 fallos de pagina

Para referencias 19:

Los datos son: 100 páginas, 25 marcos, 0,20 porcentaje de localidad se detectaron:620 fallos de pagina

Lo que nos da:

Porcentaje	Fallas
20%	620
40%	476
60%	359
80%	204



Esta gráfica nos indica que al aumentar el porcentaje de localidad nos vamos a encontrar con menos fallos de página.

Lo que tiene sentido porque estamos centrando los valores esperados de referencia en una zona en específico, lo que reduce la posibilidad de que salte una referencia fuera de este rango y se cause un fallo de página.

Por último, comparamos marcos vs porcentaje vs marcos

Para esto usamos los siguientes archivos (que creamos nosotros) con los siguientes valores asignados:

Referencias 00:

Los datos son: 100 páginas, 10 marcos, 0,20 porcentaje de localidad se detectaron:834 fallos de pagina

Referencias 01:

Los datos son: 100 páginas, 10 marcos, 0,40 porcentaje de localidad se detectaron:784 fallos de pagina

Referencias 02:

Los datos son: 100 páginas, 10 marcos, 0,60 porcentaje de localidad se detectaron:734 fallos de pagina

Referencias 03:

Los datos son: 100 páginas, 10 marcos, 0,80 porcentaje de localidad se detectaron:665 fallos de pagina

Referencias 10:

Los datos son: 100 páginas, 15 marcos, 0,20 porcentaje de localidad se detectaron:773 fallos de pagina

Referencias 11:

Los datos son: 100 páginas, 15 marcos, 0,40 porcentaje de localidad se detectaron:680 fallos de pagina

Referencias 12:

Los datos son: 100 páginas, 15 marcos, 0,60 porcentaje de localidad se detectaron:601 fallos de pagina

Referencias 13:

Los datos son: 100 páginas, 15 marcos, 0,80 porcentaje de localidad se detectaron:510 fallos de pagina

Referencias 20:

Los datos son: 100 páginas, 20 marcos, 0,20 porcentaje de localidad se detectaron:705 fallos de pagina

Referencias 21:

Los datos son: 100 páginas, 20 marcos, 0,40 porcentaje de localidad se detectaron:603 fallos de pagina

Referencias 22:

Los datos son: 100 páginas, 20 marcos, 0,60 porcentaje de localidad se detectaron:470 fallos de pagina

Referencias 23:

Los datos son: 100 páginas, 20 marcos, 0,80 porcentaje de localidad se detectaron:352 fallos de pagina

Referencias 30:

Los datos son: 100 páginas, 25 marcos, 0,20 porcentaje de localidad se detectaron:631 fallos de pagina

Referencias 31:

Los datos son: 100 páginas, 25 marcos, 0,40 porcentaje de localidad se detectaron:485 fallos de pagina

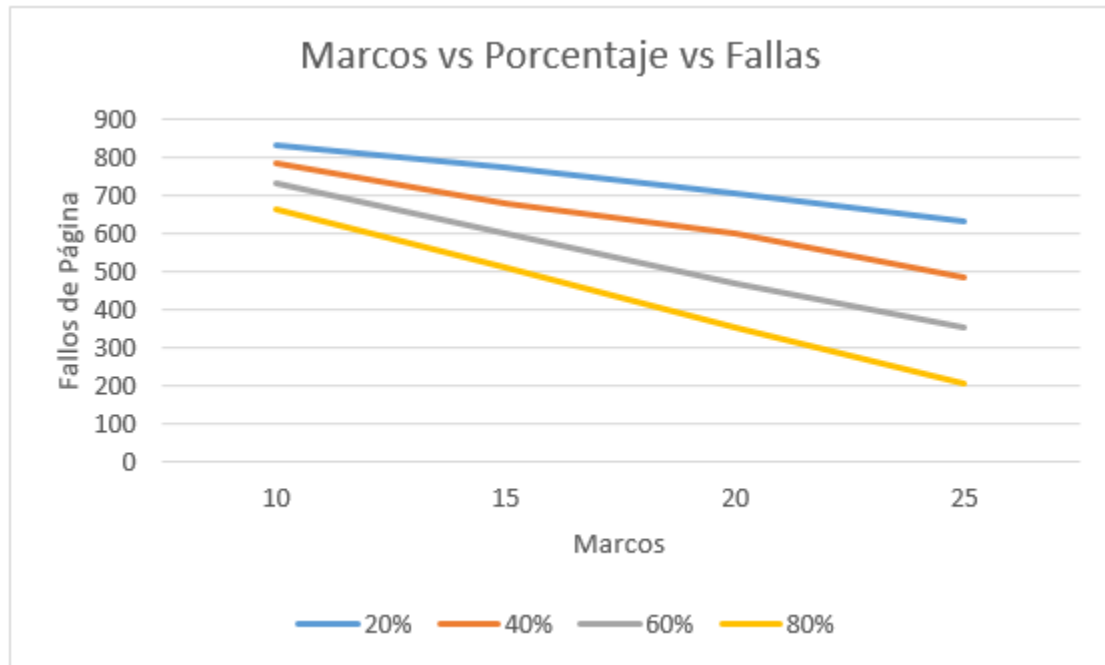
Referencias 32:

Los datos son: 100 páginas, 25 marcos, 0,60 porcentaje de localidad se detectaron:354 fallos de pagina

Referencias 33:

Los datos son: 100 páginas, 25 marcos, 0,80 porcentaje de localidad se detectaron:205 fallos de pagina

Marcos	Porcentaje de Localidad				
		20%	40%	60%	80%
	10	834	784	734	665
	15	773	680	601	510
	20	705	603	470	352
	25	631	485	354	205



Lo que finalmente nos indica que los fallos de página son dependientes tanto de los marcos como del porcentaje de localidad, entre mayor sea el porcentaje de localidad y mayor sea el número de marcos, menos fallos ocurrirán.

En síntesis, para un sistema sería óptimo tener la mayor cantidad de marcos de página posible y se esperaría que el porcentaje de localidad sea lo mayor posible.

Por otro lado, para los datos de prueba que nos entregaron,

Fallos de Página Esperados vs Fallos de Página Obtenidos:

	R1	R2	R3	R4
Fallos de Página Obtenidos	597	601	755	337
Fallos de Página Esperados	587	577	758	340
Porcentaje de aproximación	>10%	>10%	>10%	>10%

Dado que todos los fallos de Página obtenidos se acercan a los esperados y tienen menos de 10% de diferencia, se puede decir que estos datos se encuentran en el margen esperado.