

IIC2343 – Arquitectura de Computadores

AYUDANTÍA 8

Multiprogramación

Constanza Osorio

¿Más de un programa en memoria?

- Si deseamos cargar dos programas en memoria, debemos decidir cómo manejaremos la memoria para lograrlo.
- Una forma simple de hacer esto es darle una parte de la memoria a cada programa. Sin embargo esto genera **3 problemas**:
 1. *El programador debe saber a priori el espacio de memoria correspondiente.*
 2. *No hay protección. Es posible que un programa escriba en los datos del otro.*
 3. *El tamaño de memoria sería fijo para cada programa*

Memoria Virtual

- Tendremos direcciones virtuales, en vez de manejar directamente las direcciones físicas. Además, cada programa tendrá un **espacio virtual de direcciones** del tamaño del espacio de direcciones de la memoria completa.
- Sin embargo, en la práctica cada programa sigue estando mapeado a un lugar distinto en memoria física. Este mapeo es transparente a nivel de programa por lo que el programa sigue pensando que tiene toda la memoria a su disposición.
- Con este sistema, solucionamos los 3 problemas descritos anteriormente

(◉>~<)◉

Memory Management Unit (MMU)

- Para implementar un sistema de memoria virtual, necesitaremos agregar a la CPU un nuevo componente de hardware. Este componente se encargará de hacer el mapeo de las direcciones virtuales a direcciones físicas.
- Se encargará de traducir las solicitudes de vengan desde la CPU, mapeando para los programas las direcciones físicas correspondientes.
- Para hacer esto, la MMU debe almacenar una tabla que tenga la asociación virtual-física. Sin embargo, un mapeo 1:1 de las direcciones físicas con las virtuales no es implementable en la práctica, ya que para cada programa se requeriría un espacio de almacenamiento igual al tamaño de la memoria física.

↯((→_Λ←))↯

Paginación $O((\ast^{\nabla}\ast))O$

- Corresponde a dividir la memoria en bloques de palabras **contiguos**, conocidos como **páginas** en el espacio virtual, o **marcos** en el espacio físico.
- Cada programa tendrá asociada una cierta cantidad de páginas que se encontrarán mapeadas a marcos físicos.
- Con esto, las tablas de mapeo, denominadas tablas de páginas, tendrán un tamaño razonable.

Translation Lookaside Buffer (TLB)

- Un problema de tener las tablas de páginas en memoria es, que para acceso a memoria de un programa dado, se requieren **dos accesos** en la práctica:
 1. Ir a buscar el mapeo virtual-físico en la tabla de página
 2. Realizar el acceso real

- Para mejorar el rendimiento en los accesos a memoria y evitar este doble acceso, se agrega una caché dedicada a almacenar entradas de tabla de página, conocida como Translation Lookaside Buffer, o TLB.

- La TLB almacenará algunas de las entradas de la tabla de página del programa actualmente en ejecución. Tendrá el mismo funcionamiento de una caché tradicional.

Page Fault (° □ °) ☹️ 📀

- ¿Qué sucede cuando ya no tenemos marcos físicos disponibles? Panik
 - No podemos asociar nuestra página a un marco, ya que no hay más disponibles, por lo que ocurre un **Page Fault**.
 - Para solucionar esto, usaremos el disco duro como almacenamiento de respaldo para los marcos. Reservaremos un espacio especial en el disco, llamado **swap file**, el que será usado para respaldar los marcos.
- Entonces, cuando un programa requiera de un marco y tengamos nuestra memoria llena, se copiará un marco de memoria al disco, para dejar espacio para la nueva solicitud. Esto se conoce como **Swap Out**.
- Además, deberemos agregar un nuevo bit de información a nuestra tabla para determinar si la página está en disco o no.
- Cuando queramos una página que se encuentra en disco y no en memoria, realizaremos un **Swap In**. Esto consistirá en tomar la página en disco y restaurar el mapeo.

Considere un computador con una memoria principal de 64 palabras, y un tamaño de página de 8 palabras. Sea un programa con la siguiente secuencia de accesos:

38, 10, 21, 57, 46, 24, 34, 27

Desarrolle la secuencia de mapeo de memoria virtual de dos procesos del mismo programa, considerando que los accesos a memoria de cada uno de ellos se realizan alternadamente de 2 en 2. Indique las faltas de página y los *swaps*. La política de remplazo debe ser FIFO.

EJERCICIO 3

Accesos y asignación de páginas

número	página (0-7)	binario	decimal
4	100	100110	38
1	001	001010	10
2	010	010101	21
7	111	111001	57
5	101	101110	46
3	011	011000	24
4	100	100010	34
3	011	011011	27

Tablas Iniciales luego de: Proceso 1, acceso MEM[38]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	X	0	0
	2	X	0	0
	3	X	0	0
1	4	1	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	X	0	0
	2	X	0	0
	3	X	0	0
	4	x	0	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 1, acceso MEM[10]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
	2	X	0	0
	3	X	0	0
1	4	1	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	X	0	0
	2	X	0	0
	3	X	0	0
	4	x	0	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2, acceso MEM[38]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
	2	X	0	0
	3	X	0	0
1	4	1	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	X	0	0
	2	X	0	0
	3	X	0	0
3	4	3	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2, acceso MEM[10]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
	2	X	0	0
	3	X	0	0
1	4	1	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
4	1	4	1	0
	2	X	0	0
	3	X	0	0
3	4	3	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 1, acceso MEM[21]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
5	2	5	1	0
	3	X	0	0
1	4	1	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
4	1	4	1	0
	2	X	0	0
	3	X	0	0
3	4	3	1	0
	5	X	0	0
	6	X	0	0
	7	x	0	0

Proceso 1, acceso MEM[57]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
5	2	5	1	0
	3	x	0	0
1	4	1	1	0
	5	x	0	0
	6	x	0	0
6	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
4	1	4	1	0
	2	x	0	0
	3	x	0	0
3	4	3	1	0
	5	x	0	0
	6	x	0	0
	7	x	0	0

Proceso 2, acceso MEM[21]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
5	2	5	1	0
	3	x	0	0
1	4	1	1	0
	5	x	0	0
	6	x	0	0
6	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
4	1	4	1	0
7	2	7	1	0
	3	x	0	0
3	4	3	1	0
	5	x	0	0
	6	x	0	0
	7	x	0	0

Proceso 2, acceso MEM[57]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	2	1	0
5	2	5	1	0
	3	x	0	0
1	4	1	1	0
	5	x	0	0
	6	x	0	0
6	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
4	1	4	1	0
7	2	7	1	0
	3	x	0	0
3	4	3	1	0
	5	x	0	0
	6	x	0	0
8	7	8	1	0

Proceso 1, acceso MEM[46]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
1	1	2	1	0
4	2	5	1	0
	3	x	0	0
	4	1	1	1
8	5	1	1	0
	6	x	0	0
5	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
3	1	4	1	0
6	2	7	1	0
	3	x	0	0
2	4	3	1	0
	5	x	0	0
	6	x	0	0
7	7	8	1	0

Page Fault! Debemos hacer Swap Out por FIFO. Haremos Swap Out del marco 1.
Escribimos ahora, la página 5 en el marco 1.

Proceso 1, acceso MEM[24]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
3	2	5	1	0
8	3	2	1	0
	4	1	1	1
7	5	1	1	0
	6	x	0	0
4	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
2	1	4	1	0
5	2	7	1	0
	3	x	0	0
1	4	3	1	0
	5	x	0	0
	6	x	0	0
6	7	8	1	0

Page Fault! Debemos hacer Swap Out por FIFO. Haremos Swap Out del marco 2.
Escribimos ahora, la página 3 en el marco 2.

Proceso 2, acceso MEM[46]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
2	2	5	1	0
7	3	2	1	0
	4	1	1	1
6	5	1	1	0
	6	x	0	0
3	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
1	1	4	1	0
4	2	7	1	0
	3	x	0	0
	4	3	1	1
8	5	3	1	0
	6	x	0	0
5	7	8	1	0

Page Fault! Debemos hacer Swap Out por FIFO. Haremos Swap Out del marco 3.
Escribimos ahora, la página 5 en el marco 3.

Proceso 2, acceso MEM[24]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
1	2	5	1	0
6	3	2	1	0
	4	1	1	1
5	5	1	1	0
	6	x	0	0
2	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	4	1	1
3	2	7	1	0
8	3	4	1	0
	4	3	1	1
7	5	3	1	0
	6	x	0	0
4	7	8	1	0

Page Fault! Debemos hacer Swap Out por FIFO. Haremos Swap Out del marco 4.
Escribimos ahora, la página 3 en el marco 4.

Proceso 1, acceso MEM[34]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
	2	5	1	1
5	3	2	1	0
8	4	5	1	0
4	5	1	1	0
	6	x	0	0
1	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	4	1	1
2	2	7	1	0
7	3	4	1	0
	4	3	1	1
6	5	3	1	0
	6	x	0	0
3	7	8	1	0

Page Fault! Debemos hacer Swap Out por FIFO. Haremos Swap Out del marco 5. Luego hacemos Swap In de la página 4.

Proceso 1, acceso MEM[27]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
	2	5	1	1
5	3	2	1	0
8	4	5	1	0
4	5	1	1	0
	6	x	0	0
1	7	6	1	0

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	4	1	1
2	2	7	1	0
7	3	4	1	0
	4	3	1	1
6	5	3	1	0
	6	x	0	0
3	7	8	1	0

Se accede a la página 3 (Marco 2).

Proceso 2, acceso MEM[34]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
	2	5	1	1
4	3	2	1	0
7	4	5	1	0
3	5	1	1	0
	6	x	0	0
	7	6	1	1

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	4	1	1
1	2	7	1	0
6	3	4	1	0
8	4	6	1	0
5	5	3	1	0
	6	x	0	0
2	7	8	1	0

Page Fault! Debemos hacer Swap Out por FIFO. Haremos Swap Out del marco 6. Luego hacemos Swap In de la página 4.

Proceso 2, acceso MEM[27]

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
	2	5	1	1
4	3	2	1	0
7	4	5	1	0
3	5	1	1	0
	6	x	0	0
	7	6	1	1

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	4	1	1
1	2	7	1	0
6	3	4	1	0
8	4	6	1	0
5	5	3	1	0
	6	x	0	0
2	7	8	1	0

Se accede a la página 3 (Marco 4).

Tablas Finales

Proceso 1				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	2	1	1
	2	5	1	1
4	3	2	1	0
7	4	5	1	0
3	5	1	1	0
	6	x	0	0
	7	6	1	1

Proceso 2				
Orden	Página	Marco	Validez	Disco
	0	x	0	0
	1	4	1	1
1	2	7	1	0
6	3	4	1	0
8	4	6	1	0
5	5	3	1	0
	6	x	0	0
2	7	8	1	0

Resumen de Acciones

Decimal	Página	Acción
38	4	Se escribe P4-proc1 en M1
10	1	Se escribe P1-proc1 en M2
38	4	Se escribe P4-proc2 en M3
10	1	Se escribe P1-proc2 en M4
21	2	Se escribe P2-proc1 en M5
57	7	Se escribe P7-proc1 en M6
21	2	Se escribe P2-proc2 en M7
57	7	Se escribe P7-proc2 en M8
46	5	Page Fault. Swap Out M1(P4-proc1-proc1). Se escribe P5 en M1
24	3	Page Fault. Swap Out M2(P1-proc1-proc1). Se escribe P3 en M2
46	5	Page Fault. Swap Out M3(P4-proc2). Se escribe P5-proc2 en M3
24	3	Page Fault. Swap Out M4(P1-proc2). Se escribe P3-proc2 en M4
34	4	Page Fault. Swap Out M5(P2-proc1). Swap In(P4-proc1)
27	3	Se Accede
34	4	Page Fault. Swap Out M6(P7-proc1). Swap In(P4-proc2)
27	3	Se Accede