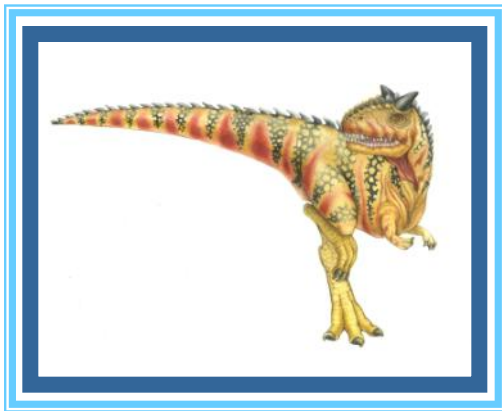


# Topic 3: Memory Management



Silberschatz, Galvin and Gagne ©2018

Rudowsky ©2005

Walpole ©2010

Kubiatowicz ©2010

Stallings ©2015

- Chapter 8: Main Memory
- Chapter 9: Virtual Memory



# Exercise 1

---

- Sea un sistema de gestión de memoria paginada con un direccionamiento físico máximo de 256 MB y un direccionamiento lógico de 1 GB. ¿Qué tamaño de página se debe usar si queremos que el número de entradas en la tabla de página sea  $2^{19}$ ?





## Exercise 2

- Un sistema de gestión de memoria virtual paginada con espacio lógico de 512 MB y espacio físico de 32 MB, con tamaño de página de 8 KB, se pagina en dos niveles.
  - Señala los campos de la dir. virtual y su tamaño (bits) si las entradas de la TP incluyen además del marco 4 bits de control.



- ¿Cuántas páginas se necesitan para ubicar la tabla de 1er nivel (sus entradas de igual tamaño que la TP)?
- Suponiendo que un proceso que ocupa sólo sus tres primeras páginas (espacio virtual) comienza a ejecutarse sin tener residente ninguna página del sistema de traducción y que los marcos disponibles son: 20,21,22,23,24,40,41,42,50,80,90,100, indica:
  - ▶ ¿Qué marcos se asignarán a la tabla de 1er nivel?
  - ▶ ¿Cuántas páginas ocupará la TP (2º nivel)?
  - ▶ ¿En qué marco(s) estará(n) ubicada(s)?





## Exercise 3

---

- Sistema de gestión de memoria de segmentación paginada (una tabla de páginas por segmento).
  - Memoria física: 512 Mbytes
  - Espacio lógico: 32 Gbytes
  - 8 segmentos por proceso como máximo cuya tabla de traducción se almacena en la MMU
  - 4 bytes para cada entrada de la TP
  - La TP de cada segmento cabe exactamente en un página
- ¿Cuál debe ser el tamaño de página?
- ¿Cuántos bits son necesarios para seleccionar un marco de página físico?
- ¿Cuántas entradas debe tener la tabla de páginas de un segmento?





## Exercise 4

- Sistema de gestión de memoria paginada:
  - Tamaño página: 2 KB
  - Tamaño espacio lógico: 64 GB
  - Tamaño espacio físico: 4 GB
  - 11 bits de control en las entradas de las TP
- ¿Tamaño máximo en Bytes de la TP de un solo nivel?

- Paginación en tres niveles (SS, S, P). Indica el número de bits para cada nivel de la dirección lógica, suponiendo que el tamaño de las entradas para todas las tablas de cualquier nivel es el mismo y que las tablas de los niveles S y P ocupan exactamente una página:

SS	S	P	D





## Exercise 4

- Si un proceso tiene un espacio de direccionamiento que abarca las 16 páginas inferiores y las 32 superiores de su espacio lógico, ¿Cuántas páginas consumirían las tablas de traducción si todas residieran en memoria?

- Indica la información que habría en cada uno de los niveles de traducción después de cargar la primera página del proceso, suponiendo que la primera referencia a memoria es la dir. lógica 100 y asumiendo que las páginas [100-4000] están libres. Asignar primero las páginas para SS, luego para S y luego para P que hagan falta.

TP SS

	Marco	V
0		
1		
2		
3		
...		

TP S

	Marco	V
0		
1		
2		
3		
...		

TP P

	Marco	V
0		
1		
2		
3		
...		





# Exercise 5.1

- Sistema de gestión de memoria segmentación paginada (una tabla de páginas por segmento):
  - Tamaño página: 1 KB
  - Tamaño espacio lógico: 128 KB
  - Tamaño espacio físico: 256 MB
  - 4 segmentos por proceso

- ¿Número de entradas de la TP de un segmento?

- ¿Bits necesarios para seleccionar un marco de página físico?





## Exercise 5.2

- La tabla de segmentos se almacena en registros de la MMU y cada tabla de páginas de un segmento se almacena en una página de memoria física diferente.
- Llega un proceso que requiere 3 segmentos de 2900, 2048 y 3100 bytes. Están ocupados los marcos físicos [0-100], [103-115] y [120-131].
- Asignando el primer marco libre disponible (dirección menor), ubicar el proceso en memoria (comienza por los marcos para las TP de cada seg. válido y a continuación con las páginas de cada seg.)

Tabla de segmentos

	Marco de la TP	Tam. Seg.	V
0			
1			
2			
3			

TP S0

	Marco	V
0		
1		
2		
3		
...		

TP S1

	Marco	V
0		
1		
2		
3		
...		

TP S2

	Marco	V
0		
1		
2		
3		
...		







## Exercise 5.3

- Traduce las siguientes direcciones lógicas a físicas:

Dir. Lógica	Dir. Física
1:1:200	
0:2:128	
2:2:128	
1:2:255	

- ¿Cuál es la fragmentación interna que provoca el proceso?



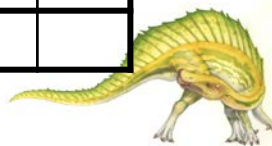


## Exercise 5.4

- Mem. Virtual con reemplazo WS (working set) con tamaño de ventana 4, con la misma ocupación de memoria anterior y con las TP cargadas en memoria en la inicialización del proceso.
- 15 primeras referencias a memoria:
  - 2:3:002, 2:1:000, 2:3:003, 2:3:200, 2:1:001, 2:1:002, 1:1:120, 1:1:121, 0:2:020, 0:2:021, 2:3:002, 1:1:122, 2:1:100, 2:1:003, 2:3:003, ...
- Completa la sig. tabla (evolución del working set) marcando los fallos de página:

Ref	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Pag															
WS															
Fallo															

- La quinta referencia (2:1:001) se traduciría a @F:





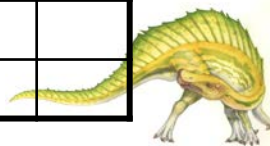
## Exercise 6

- Completa la sig. Tabla marcando los fallos de página para cada política de reemplazo:

Pag	1	2	3	2	1	4	3	1	2	1	5	3	1	2	1
LRU															
Fallo															

Pag	1	2	3	2	1	4	3	1	2	1	5	3	1	2	1
Reloj															
Fallo															

Pag	1	2	3	2	1	4	3	1	2	1	5	3	1	2	1
WS( 3)															
Fallo															





# Exercise 7.1

- Se analiza un sistema de memoria virtual con las siguientes características:
  - Se trata de paginación multinivel de 3 niveles
  - El espacio lógico se compone de  $2^{18}$  páginas
  - El espacio físico contiene 128 marcos
  - Memoria direccionada a nivel de byte
  - Las tablas se hayan paginadas en fragmentos de tamaño de una página
  - Las entradas de todas las tablas de página son de 4Bytes
- Sabiendo que los niveles 2º y 3º de la paginación se direccionan con 8 bits, obtener la longitud de los campos que faltan en la dirección lógica:

L1	L2	L3	offset
	8	8	





## Exercise 7.2

---

- Calcular el número de bits de la dirección física y el tamaño total del espacio físico en bytes.
- ¿Cuántos bits, al menos, debería tener el registro base de la tabla de página (BTPR)?
- ¿Cuál el espacio total que ocuparían todas las tablas de página de todos los niveles?





## Exercise 7.3

---

- Un proceso tiene los siguientes requerimientos de direccionamiento: segmento de datos de 4 páginas, segmento de código de una página, segmento de pila de 3 páginas.
  - En ausencia de TLB, ¿cuántos accesos a memoria son necesarios para acceder a una posición del proceso?
  - Si la totalidad del proceso se haya residente en memoria, ¿cuántos marcos de memoria estarán ocupados por las tablas de página a lo sumo? ¿Y cómo mínimo?





## Exercise 7.4

- Si duplicamos el tamaño de la página, manteniendo el tamaño en bytes del espacio lógico y físico inicial y manteniendo las dos últimas condiciones del comienzo del enunciado (Las tablas se hayan paginadas en fragmentos de tamaño de una página y las entradas de todas las tablas de página son de 4Bytes):
  - ¿Cuál será ahora el número de marcos en memoria física?
  - ¿Cuál es el tamaño ahora de cada uno de los campos de direccionamiento de la dirección virtual? (indicar con 0 si alguno ha desaparecido)

L1	L2	L3	offset





## Exercise 7.5

- En la figura adjunta se muestra parte del contenido de la memoria física así como de la TLB. Se supondrá una TLB de 4 posiciones, completamente asociativa y con reemplazo FIFO. Asimismo el reemplazo de páginas es también FIFO.
- La tabla de nivel 1 está en el marco cero, esto es, PTBR=0.
- Completa la siguiente tabla indicando la información que se pide para cada una de las siguientes referencias lógicas de memoria. En caso de fallo de página ó un direccionamiento fuera del espacio válido del proceso, indicar con -1.

@L	Válida?	Fallo TLB	Fallo página	@F	contenido
0:1:2:4					
4:0:0:4					
0:1:0:0					
1:2:0:0					
2:0:1:4					







### TLB:

Página lógica	Marco (pág. física)	V
4:0:0	7	1
4:0:8	9	1
4:1:0	15	1
		0

Marco 0	Offset	Contenido
	0	(1, V=1, P=1)
	4	(2, V=1, P=1)
	8	(6, V=0, P=0)
	12	

....

Marco 4	Offset	Contenido
	0	(6, V=1, P=1)
	4	
	8	(?, V=1, P=0)
	12	

....

Marco 1	Offset	Contenido
	0	
	4	(3, V=1, P=1)
	8	
	12	

....

Marco 5	Offset	Contenido
	0	0x00000000
	4	0x12345678
	8	
	12	

....

Marco 2	Offset	Contenido
	0	
	4	
	8	(4, V=1, P=1)
	12	

....

Marco 6	Offset	Contenido
	0	0xFFFFFFFF
	4	0x00000000
	8	

....

Marco 3	Offset	Contenido
	0	(?, V=1, P=0)
	4	
	8	(5, V=1, P=1)
	12	

Marco 7	Offset	Contenido
	0	0x00000000
	4	0x0000000A
	8	

....

