

# Guía práctica de memoria

1. Dadas las siguientes particiones fijas de memoria de 1000KB, 400KB, 1800KB, 700KB, 900KB, 1200KB y 1500KB (definidas en este orden):
  - a. ¿Cómo se situarán en memoria una serie de procesos de 1200KB, 1000KB y 900KB con los algoritmos de *Primer Ajuste*, *Mejor Ajuste* y *Peor Ajuste*?
  - b. ¿Qué algoritmo hace el uso más eficiente de memoria?
2. Dadas cinco particiones fijas de memoria de 100KB, 500KB, 200KB, 300KB, 600KB (definidas en este orden):
  - a. ¿Cómo se situarán en memoria una serie de procesos de 212KB, 417KB, 112KB y 426KB con los algoritmos de *Primer Ajuste*, *Mejor Ajuste* y *Peor Ajuste*?
  - b. ¿Qué algoritmo hace el uso más eficiente de memoria?
3. En un sistema operativo con particiones fijas, la memoria tiene un tamaño de 64KB de los cuales el sistema operativo ocupa la primer posición y tiene un tamaño de 10K, el resto de la memoria está dividido en 4 particiones como se representa en la siguiente tabla:

BASE	TAMAÑO
10	18KB
28	12KB
40	8KB
48	16KB

¿Cómo se situarán en memoria una serie de procesos de 8KB, 14KB, 18KB, 6KB y 14KB con el *Mejor Ajuste*?

4. En un sistema operativo con particiones variables, la memoria tiene un tamaño de 64KB de los cuales el sistema operativo ocupa la primera posición y tiene un tamaño de 10KB. ¿Cómo se situarán en memoria la serie de procesos que se representan en la siguiente tabla con el algoritmo de *Mejor Ajuste* y planificación de CPU con FIFO?:

PROCESO	TAMAÑO	LLEGADA	DURACIÓN
A	8KB	1	4
B	14KB	2	5
C	18KB	3	3

D	6KB	4	3
E	14KB	5	2

5. En un sistema operativo con particiones variables, la memoria tiene un tamaño de **128KB**, de los cuales el sistema operativo ocupa la primer posición y tiene un tamaño de **10KB**. ¿Cómo se situarán en memoria la serie de procesos que se representan en la siguiente tabla con el algoritmo de *Mejor Ajuste* y planificación de CPU con FIFO?.

PROCESO	TAMAÑO	LLEGADA	DURACIÓN
A	20KB	1	2
B	40KB	1	5
C	35KB	1	3
D	25KB	2	5
E	15KB	4	3
F	35KB	5	2

6. Un bloque de **128KB** de memoria utiliza el algoritmo de asignación de memoria Buddy. Después de un tiempo el bloque queda de la siguiente manera, donde los bloques de memoria identificados con X están asignados y el resto están libres.

MEM	X	X	X			X	X		X		X	X	X	X		X
DIR	0	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120

Luego, estos bloques son liberados de la siguiente manera:

ORDEN	DIRECCIÓN	TAMAÑO
1	64KB	8KB
2	0KB	24KB
3	96KB	16KB

Muestre cual es el estado del bloque de memoria después de cada liberación.

7. En un sistema operativo que utiliza el algoritmo de asignación de memoria Buddy, la memoria tiene un tamaño de **512KB**, ¿cómo se situarán en dicha memoria los procesos que se muestran en la siguiente tabla?:

PROCESO	TAMAÑO	LLEGADA	DURACIÓN
A	65KB	1	4

B	30KB	2	4
C	250KB	3	3
D	70KB	4	2

8. Suponiendo que la tabla de páginas de un proceso que se está ejecutando en el procesador es:

Nº PÁGINA	Nº MARCO DE PÁGINA	R	M	V
0	4	1	0	1
1	7	1	1	1
2		0	0	0
3	2	0	0	1
4		0	0	0
5	0	0	1	1

donde **R** es el bit de referencia (R=1 la página ha sido referenciada), **M** es el bit de modificación, (M=1 la página ha sido modificada) y **V** es el bit de presente/ausente (V=1 la página está en memoria principal, tiene un marco asociado).

Si el tamaño de las páginas es de **1KB** y el marco **0** está cargado en la dirección física **0** y el resto de forma sucesiva, ¿a qué direcciones físicas corresponden las siguientes direcciones virtuales?

- (1,125)
- (2,324)
- (5,322)
- (7,321)
- (3,1026)

El formato en que se da la dirección virtual es (número de página, desplazamiento).

9. Considere el espacio de direccionamiento lógico de **8** páginas de **1024B** cada una, mapeado en una memoria física de **32 frames**:

- ¿Cuántos bits hay en la dirección lógica?.
- ¿Cuántos bits hay en la dirección física?.

10. Dado un sistema con la siguiente tabla de páginas:

PÁGINA DE MEMORIA VIRTUAL	PÁGINA DE MEMORIA FÍSICA	BIT DE PRESENCIA
15	000	0
14	000	0
13	000	0
12	000	0
11	111	1
10	000	0
9	101	1
8	000	0
7	000	0
6	000	0
5	011	1
4	100	1
3	000	1
2	110	1
1	001	1
0	010	1

Obtener las direcciones físicas correspondientes para las siguientes direcciones virtuales. Las direcciones virtuales tienen 4 bits de página y 12 bits de desplazamiento.

- a. 400h
- b. 0BF4h
- c. 1101010011100000
- d. 1011001110010010
- e. 20895
- f. B120h
- g. 2A120h

11. Considere la siguiente tabla de segmentos:

SEGMENTO	BASE	LONGITUD
0	219	600
1	2300	14

2	90	100
3	1327	580
4	1952	96

¿Cuáles son las direcciones físicas para las siguientes direcciones lógicas?

- a. (0,430)
- b. (1,10)
- c. (2,500)
- d. (3,400)
- e. (4,112)

12. Considere una máquina con direcciones de **18 bits**. Los primeros **2 bits** se utilizan para identificar el segmento, y los últimos **16 bits**, el desplazamiento (*offset*) dentro del segmento. Asumir que las bases y los límites de la tabla de segmentos están configuradas de la siguiente forma:

SEGMENTO	BASE	DESPLAZAMIENTO	PROTECCIÓN
0	00000h	0ABCDh	R - -
1	1B000h	007FFh	- - X
2	1B800h	00FFFh	RW -
3	30000h	01234h	RW -

Donde la protección se especifica en formato RWX (R=Read, W=Write y X=Exec).

Responder en forma clara y concisa que sucede cuando el proceso intenta escribir en cada una de las siguientes direcciones virtuales: **20000h**, **10000h**, **0BEEFh**, **00ACEh**.

13. Considere un sistema de paginación en el que se puede direccionar como máximo **1GB** de memoria, el tamaño de página es de **16KB** y cada **byte** se direcciona independientemente. ¿Cuántas páginas podrá tener asignadas como máximo un proceso en este sistema?. Si empleamos una tabla de páginas con dos niveles, en el que la tabla de primer nivel contiene **1024** entradas, ¿cuántas tablas de segundo nivel son necesarias para un proceso que requiere **6401** páginas?. Razone su respuesta.
14. Considere un sistema de paginación en el que las direcciones lógicas son de **22 bits** y el tamaño de página es de **2KB**. Sabiendo que cada byte se direcciona independientemente, calcule el ahorro de memoria que obtendríamos para representar la tabla de páginas de un

proceso que está utilizando 90KB de memoria, si empleamos una tabla de páginas con dos niveles en lugar de tener una tabla de un solo nivel.

En el sistema con dos niveles, debes considerar que se emplean 5 bits de la dirección para el segundo nivel. Además, cada entrada de las tablas de páginas precisa 8 bytes.

15. Considere un sistema de paginación en el que las direcciones lógicas son de 20 bits y el tamaño de página es de 4KB. Sabiendo que cada byte se direcciona independientemente, calcule el ahorro de memoria que obtendríamos para representar la tabla de páginas de un proceso que está utilizando 192KB de memoria, si empleamos una tabla de páginas con dos niveles en lugar de tener una tabla de un solo nivel.

En el sistema con dos niveles debes considerar que se emplea el mismo número de bits de la dirección para cada nivel. Cada entrada de las tablas de páginas precisa 16 bytes.

16. Dado un sistema de gestión de memoria basado en múltiples niveles de paginación, se trata de determinar el número de niveles necesarios para que la tabla de primer nivel pueda caber en una TLB de 256 bytes. Se supone que la dirección lógica tiene un formato de 32 bits, el tamaño de página es de 1024 bytes, el tamaño de los descriptores de cualquiera de las tablas de páginas es de 8 bytes y que cada tabla de nivel superior al primero ocupa el tamaño de una página.

17. Dado un sistema de memoria *segmentado-paginado*. Los espacios de direcciones lógicas son de 8GB y el tamaño de página es de 2KB. Un espacio de direcciones puede tener hasta 256 segmentos y los descriptores de página tienen 4 bytes. Indique cuál será el tamaño de la tabla de páginas de un segmento.

18. Una computadora provee a cada proceso con un espacio de direccionamiento de 65536 bytes dividido en páginas de 4096 bytes. Si un programa particular tiene un código de 32768 bytes, datos 16386 bytes y pila de 15870 bytes. ¿Entrará este programa en el espacio de direcciones?. Si el tamaño de cada página fuera de 512 bytes, ¿entraría?.

Recordar que una página no puede contener partes de dos segmentos diferentes.

19. Se tiene un sistema operativo con las siguientes características:

- Gestión de memoria virtual formada por la combinación de segmentación y paginación.
- Direcciones virtuales de 32 bits (igual que las direcciones físicas) con el siguiente formato:

Nº DE SEGMENTO	Nº DE PÁGINA	DESPLAZAMIENTO
8 bits	12 bits	12 bits

- a. ¿Cuántos segmentos distintos puede direccionar cualquier proceso del sistema?
- b. ¿Cuántos marcos de página distintos puede haber como máximo?

20. Dado un sistema con dirección virtual de **32 bits** y tamaño de página de **4KB (212 bytes)**. Determine:

- a. Bits necesarios para el desplazamiento.
- b. Bits necesarios para el número de páginas.
- c. Número posible de páginas virtuales.

21. Considere un sistema de memoria virtual que utiliza segmentación con paginación por demanda y posee una dirección de **24 bits** (tanto lógica como física), repartidos de la siguiente manera:

- a. **8 bits** representan el número de segmentos.
- b. **8 bits** representan el número de página.
- c. **8 bits** representan el desplazamiento (*offset*).

Se pide que indique:

- a. Tamaño máximo direccionable en la memoria real.
- b. Tamaño de la página.
- c. Cantidad de páginas lógicas.
- d. Cantidad de páginas físicas.
- e. Determine si son correctas las siguiente cadena de referencia con formato (segmento, página, desplazamiento): **(10,250,200)**, **(10,123,211)**, **(151,102,20)**, **(10,250,176)**, **(79,5,148)**, **(217,42,0)**, **(280,25,15)**, **(10,250,1)**, **(123,89,103)**, **(43,170,315)**.

22. Dado un sistema que posee una dirección de **32 bits** y utiliza segmentación con paginación bajo demanda. Se utilizan **12 bits** para el número de página y cada proceso puede tener un máximo de **256** segmentos.

Se pide informar:

- a. Tamaño de la página.

- b. Tamaño máximo direccionable de memoria.
- c. Cantidad de páginas por segmento.
- d. Cantidad de páginas máximas por proceso.

23. Dada la siguiente cadena de referencia:

0	1	2	3	0	1	4	0	1	2	3	4
---	---	---	---	---	---	---	---	---	---	---	---

¿Cuántos fallos de página tendrán lugar si se dispone de 4 marcos de página inicialmente vacíos para el algoritmo *Óptimo*, *FIFO*, *Segunda Oportunidad*, *LRU* y *LFU*?. Compruebe si se produce la anomalía de Belady.

24. Dada la siguiente cadena de referencia:

2	3	2	1	5	2	4	5	3	2	5	2
---	---	---	---	---	---	---	---	---	---	---	---

¿Cuántos fallos de página tendrán lugar si se dispone de 4 marcos de página inicialmente vacíos para el algoritmo *Óptimo*, *FIFO*, *Segunda Oportunidad*, *LRU* y *LFU*?. Compruebe si se produce la anomalía de Belady.

25. Dada la siguiente cadena de referencia:

0	1	7	2	3	2	7	1	0	3	0	2	3	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---

¿Cuántos fallos de página tendrán lugar si se dispone de 4 marcos de página inicialmente vacíos para el algoritmo *Óptimo*, *FIFO*, *Segunda Oportunidad*, *LRU* y *LFU*?. Compruebe si se produce la anomalía de Belady.

#### REFERENCIAS RÁPIDAS

• Bit (b): 0 o 1	
• Byte (B): 8 bits	=> $2^3$
• Kilobyte (KB): 1.024 bytes	=> $2^{10}$
• Megabyte (MB): 1.048.576 bytes	=> $2^{20}$
• Gigabyte (GB): 1.073.741.824 bytes	=> $2^{30}$
• Terabyte (TB): 1.099.511.627.776 bytes	=> $2^{40}$
• Kilobit (Kb): 1.000 bits	
• Megabit (Mb): 1.000.000 bits	
• Terabit (Tb): 1.000.000.000.000 bits	