

## Problemas Resueltos

### Tema 3. Gestión de Memoria

#### Gestión de memoria: Asignación contigua

1. Se tiene un sistema de particiones múltiples con un soporte hardware basado en la técnica de registros base y límite. Dado un programa P que ocupa T palabras y se encuentra ubicado en memoria a partir de la posición de memoria física C.
  - a) ¿Cuál es el valor de cada registro para el programa P?
  - b) ¿Cuál es el rango de las direcciones que emite P?
  - c) ¿Cuál es el rango de direcciones reales en que se transforman las direcciones que emite P?

#### Solución

- a) El registro base y límite se usa en esquemas basados en reubicación en tiempo de ejecución. El valor del registro base será C y el valor del registro límite será T.
  - b) El rango de direcciones de P es entre 0 y T-1.
  - c) Las direcciones físicas reales en las que se ha de transformar se encuentran en el rango entre C y C+T-1.
2. Analizar los esquemas de protección de particiones múltiples desde el punto de vista de la reubicación.

#### Solución

Con particiones múltiples de memoria, aparecen dos opciones que el tamaño sea fijo, o que el tamaño de la partición sea variable. En las de tamaño fijo aparece el problema de la fragmentación interna y en las de tamaño variable aparece el problema de la fragmentación externa. Los esquemas de protección que se utilizan para particiones variables son: registro cota inferior y registro cota superior o registro base y registro límite. En el primero la traducción de direcciones se realiza en tiempo de compilación resultando direcciones absolutas de memoria y procesos no reubicables. En el esquema de registro base y límite la traducción de direcciones lógicas a físicas se realiza en tiempo de ejecución resultados por tanto procesos reubicables.

3. Sea un sistema gestionado con un mecanismo de particiones variables en el que la memoria física tiene 4200 palabras. En un instante la memoria está ocupada por 3 bloques de código/datos de la forma:

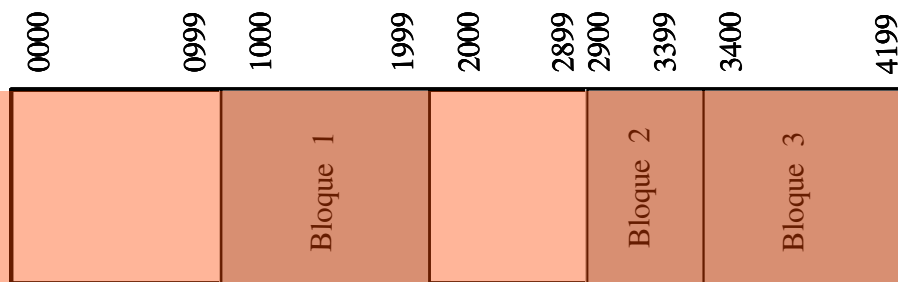
Dirección inicial	longitud
1000	1000
2900	500
3400	800

La estrategia utilizada cuando se carga un nuevo bloque en memoria es la del mejor ajuste en primer lugar. Si falla, se crea un hueco mayor desplazando los bloques en memoria hacia la dirección 0. Esta acción siempre empieza con el bloque actualmente en la dirección de memoria más baja, y prosigue únicamente hasta encontrar un hueco suficiente para el nuevo bloque.

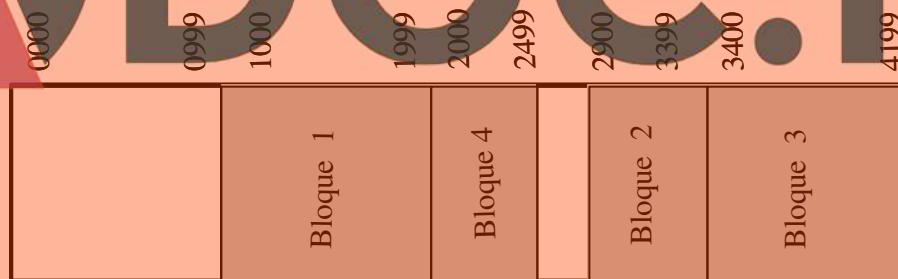
A partir de ese momento, hay que cargar tres bloques de 500, 1200 y 200 (en ese orden). Describir el contenido de la memoria una vez satisfechas las peticiones.

### Solución

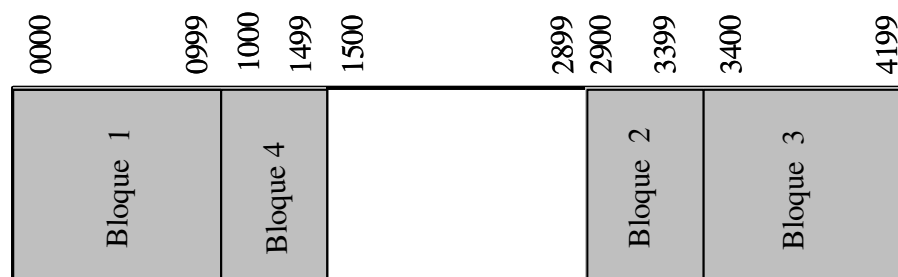
La situación inicial de la memoria seria la siguiente:



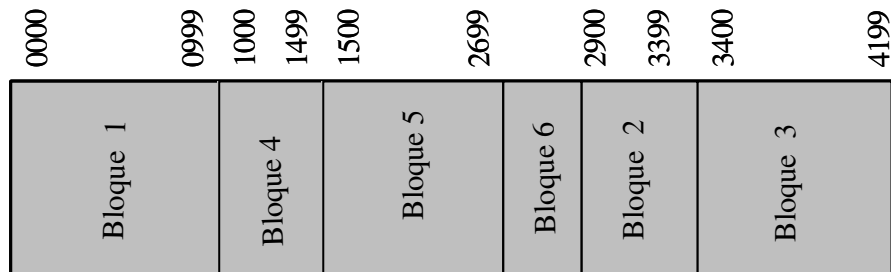
A partir de este momento cargamos los nuevos bloques, comenzando por el bloque 4 de 500 palabras utilizando el mejor ajuste.



El siguiente bloque que se ha de ubicar en memoria es el bloque 5 con 1200 palabras. Hay espacio suficiente en memoria pero aparece el problema de la fragmentación externa por tanto se hace necesario utilizar un algoritmo de compactación.



Después de la compactación han quedado libres 1400 palabras contiguas en un único hueco, la suma de los tamaños de los bloques 5 y 6 son de 1400 por tanto no habra problema para la ubicación.



4. Sea un sistema donde se utiliza un esquema de asignación con múltiples particiones de tamaño fijo. ¿Qué determinaría el límite del grado de multiprogramación de dicho sistema?

#### Solución

Es un esquema donde la memoria se encuentra dividida en varias particiones de tamaño fijo, cada partición únicamente puede contener un proceso y el grado de multiprogramación está limitado por el número de particiones.

5. Sea un sistema gestionado por particiones múltiples de tamaño variable con compactación. En un instante dado, se tiene la siguiente ocupación de la memoria:

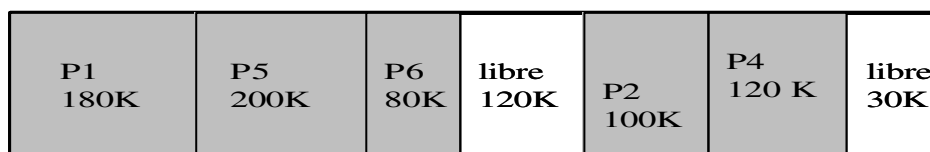


Se utiliza la técnica del mejor ajuste. En la cola de trabajos tenemos en este orden: P4(120K), P5(200K) y P6(80K), los cuales deben ser atendidos en orden FIFO. Suponiendo que no finaliza ningún proceso y tras intentar cargar en memoria todos los procesos que están en la cola.

- Indicad cuántas particiones quedan libres y de qué tamaño son.
- Si en esta situación se aplica compactación, indicar qué proceso o procesos deberían moverse para que el número de Kbytes manejados fuese el menor posible y quede un único hueco.
- Si los registros base de cada proceso son, respectivamente, B1, B2, B3, B4, B5 y B6, indicar cómo han cambiado los registros base correspondientes al proceso o procesos que se han movido debido a la compactación.

#### Solución

- a) Quedan dos particiones de tamaños 120K y 30 K respectivamente.



- b) Debería moverse el proceso P4 al hueco de 120K, con lo cual quedaría un único hueco de 150K.
- c) Todos los registros bases quedan igual que antes de la compactación exceptuando el del proceso P4 cuyo B4 contendrá 460 K (se ha considerado que P1 se encuentra ubicado en las direcciones más bajas de memoria).

**6. Indique si las siguientes afirmaciones son verdaderas (V) o falsas (F) y justifiquelo.**

- a) La gestión de memoria basada en particiones fijas utiliza el método de compactación para resolver el problema de fragmentación externa.
- b) El método de superposiciones ("overlays") se puede utilizar en un sistema que no gestione memoria virtual.
- c) El espacio de direcciones físicas depende del búf de direcciones del sistema.
- d) El espacio de direcciones lógicas está limitado por el tamaño de la memoria física.
- e) La compactación de un proceso sób se puede realizar en un método de reubicación dinámica de la memoria.

#### **Solución**

- a) FALSA. No es posible, ya que las particiones de tamaño fijo no presentan el problema de la fragmentación externa.
- b) VERDADERA. Las superposiciones es un método que se emplea cuando no existe memoria virtual para poder ejecutar procesos cuyo tamaño es superior al de la memoria física. El programador debía definir donde comenzaban y acababan las superposiciones.
- c) VERDADERA. No se puede emitir una dirección de mayor número de bits que el permitido por el sistema.
- d) FALSO. Las direcciones lógicas son emitidas por la CPU y sólo dependen del número de bits que esta sea capaz de emitir. El espacio de direcciones lógicas de un proceso depende del tamaño del proceso.
- e) VERDADERO. Si los proceso no son reubicables, no se pueden utilizar algoritmos de compactación, ya que la ejecución de los mismos podría ser errónea, por estar accediendo a direcciones de memoria donde no se encuentran ubicadas sus instrucciones.

**7. Suponga un sistema en el que no se dispone de MMU. ¿Existe alguna técnica que permita ejecutar en dicho sistema un programa cuyas necesidades de memoria superen la cantidad de memoria física total disponible?**

#### **Solución**

Sí, utilizando superposiciones. Dividiendo el programa en diferentes fragmentos que no deban estar todos ellos simultáneamente en memoria se puede reducir significativamente la cantidad de memoria física que el programa necesitará.

8. ¿Qué tipo de algoritmos se utilizan para gestionar sistemas con asignación contigua de memoria? ¿Qué tipo de fragmentación se puede producir en estos sistemas?

**Solución**

Los algoritmos que se pueden utilizar para gestionar los sistemas con asignación contigua de memoria son: “el primer ajuste” (*first fit*), “el mejor ajuste” (*best fit*), “el peor ajuste” (*worst fit*). Sin embargo, en particiones variables de tamaño fijo, no tiene sentido el uso del algoritmo del peor ajuste, ya que la fragmentación interna que se produce no puede ser aprovechada por otro proceso (sólo un proceso por partición). En la asignación contigua con particiones de tamaño variable puede aparecer la fragmentación externa, siendo necesario el uso de algoritmos de compactación. Mientras que en las de tamaño fijo aparece la fragmentación interna.

9. Cite tres ventajas ofrecidas por las bibliotecas de enlace dinámico cuando se las compara con las bibliotecas enlazadas estáticamente para formar un fichero ejecutable.

**Solución**

Cuando se emplea enlace dinámico los ficheros ejecutables ocuparán menos espacio, ya que no contienen el código de las bibliotecas.

Los procesos pueden compartir memoria.

Los programas no tendrán que ser recompilados ni reenlazados para tener acceso a las mejoras de implementación efectuadas en las bibliotecas.

## Gestión de memoria: paginación y segmentación

10. Suponga que un proceso emite una dirección lógica igual a 2453 y que se utiliza la técnica de paginación, con páginas de 1024 palabras.

- a) Indique el par de valores (número de página, desplazamiento) que corresponde a dicha dirección.
- b) ¿Es posible que dicha dirección lógica se traduzca en la dirección física 9322?. Razónelo.

**Solución**

a) página = 2 y desplazamiento = 405.

b) No porque el desplazamiento de la dirección lógica 9322 es 106 ( $(9322 \bmod 1024) = 106$ ) y debería ser 405 para que fuese posible.

11. En un sistema de gestión de memoria por paginación, ¿es recomendable que el tamaño de página sea potencia de 2?. Justifíquese la respuesta.

**Solución**

Si. Porque facilita la traducción de una dirección lógica a un número de páginas y desplazamiento. Los bits de mayor peso se dedican al número de página, y los de menor peso al desplazamiento dentro de la página. De esta manera se ahorra una operación aritmética de división, que tiene un coste mayor.

Se puede aprovechar totalmente el espacio de direcciones.

12. Considere los 4 sistemas con gestión de memoria mediante paginación A, B, C y D siguientes:

Sistema	A	B	C	D
Tamaño de página (en palabras)	512	512	1024	1024
Tamaño de palabra (en bits)	16	32	16	32

Asumiendo que sólo hay una tabla de páginas para todo el sistema y que la dirección lógica ocupa una palabra (de 16 o 32 bits, según el caso), determine para cada sistema:

- El tamaño de la tabla de páginas (número de entradas).
- El tamaño de la memoria lógica (número de páginas).

### Solución

Sistema	A	B	C	D
Tamaño de página (en palabras)	512	512	1024	1024
Tamaño de palabra (en bits)	16	32	16	32
Tamaño de la tabla de páginas (en descriptores)	128 descriptores	8 M descriptores	64 descriptores	4M descriptores
Tamaño de memoria virtual en páginas del sistema	128 páginas	8M páginas	64 páginas	4M páginas

13. Indique en cuales de los siguientes mecanismos de gestión de memoria puede producirse fragmentación externa: particiones variables, paginación, segmentación paginación por demanda, segmentación por demanda.

### Solución

Los mecanismos que pueden producir fragmentación externa son: particiones variables y segmentación. En ambos casos pueden quedar en memoria huecos no contiguos cuya suma de tamaños sea suficiente para ubicar un nuevo proceso.

14. Sea un sistema de gestión de memoria basado en la técnica de la paginación. Sean tres procedimientos que ocupan 700, 200 y 500 palabras respectivamente. Determine cuánta memoria se pierde por fragmentación interna cuando estos procedimientos se cargan en memoria si el tamaño de la página es de: 1200 palabras, 2500 palabras, 3600 palabras ó 4700 palabras.

### Solución

Tamaño de página (en palabras)	1200	2500	3600	4700
Tamaño de procedimiento ( en palabras )	700	700	700	700



Fragmentación interna (en palabras)	500	1800	2900	4000
--	-----	------	------	------

Tamaño de página (en palabras)	1200	2500	3600	4700
Tamaño de procedimiento ( en palabras )	200	200	200	200
Fragmentación interna (en palabras)	1000	2300	3400	4500

Tamaño de página (en palabras)	1200	2500	3600	4700
Tamaño de procedimiento ( en palabras )	500	500	500	500
Fragmentación interna (en palabras)	700	2000	3100	4200

15. Explique la motivación y las ventajas de la técnica de compartir páginas.

**Solución**

Cuando dos o más procesos ejecutan un mismo programa, la técnica de compartición de páginas permite no tener que duplicar el código en memoria con lo que se ahorra espacio en la misma. Piense en un sistema de tiempo compartido donde existen varios usuarios que se encuentran utilizando a la vez un mismo editor o compilador.

16. Los sistemas que usan tabla de páginas invertidas, pueden tener problemas para que los procesos compartan páginas. Indique porqué y como podría solucionarse.

**Solución**

En los sistemas de páginas invertidas la tabla de marcos asocia a cada entrada una página de un determinado proceso. Sin embargo para poder compartir páginas, un mismo marco se ha de encontrar asociado al menos a dos páginas de diferentes procesos. Por lo tanto una entrada de la tabla de páginas invertidas debería poder asociarse a más de una dirección lógica de distintos procesos.

17. Sea un sistema de gestión de memoria basado en la técnica segmentación paginada. Las direcciones lógicas tienen en total 28 bits, de los cuales 10 indican el número de página. Las páginas físicas o marcos son de 1024 octetos, mientras que las direcciones físicas son de 24 bits. Suponga que un segmento determinado contiene 4800 octetos. ¿Cuánta memoria se desperdicia en fragmentación interna y externa por causa de dicho segmento?. Justifique la respuesta.

NOTA: considere únicamente la memoria asignada al segmento, sin tener en cuenta las tablas de segmentos ni de páginas.

**Solución**

Respuesta fragmentación interna 320 octetos.

En los sistemas de segmentación paginada no hay fragmentación externa, sólo interna, ya que lo que se ubica en memoria son páginas del proceso que pertenecen a un segmento u otro, del mismo.

La dirección lógica 4800 en este sistema pertenece al segmento 0, página 4, desplazamiento 704.

27	20	19	10	9	0
0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 1 0 0		1 0 1 1 0 0 0 0 0 0	
Nº de segmento		Nº de página		desplazamiento	

Por tanto la fragmentación interna vendrá dada por  $1024 - 704 = 320$  octetos.

- 18.** Compare la fragmentación interna provocada por un sistema de memoria virtual con segmentación paginada frente a uno que únicamente utiliza paginación.

### Solución

(Consultar el material proporcionado para las clases de teoría y la bibliografía).

- 19.** Enuncie los diferentes esquemas de gestión de memoria vistos en clase detallando, para cada uno, el hardware que requieren y su problemática de fragmentación.

### Solución

(Consultar el material proporcionado para las clases de teoría).

- 20.** Razone las ventajas de utilizar la técnica de compartir segmentos en la llamada al sistema `exec()` de Unix desde los puntos de vista de gestión de la memoria y de accesos a disco.

## Solución

Gestión de la memoria: Resulta sencillo compartir el “segmento” de código con otros procesos que estén ejecutando el mismo programa que se ha especificado como argumento de “exec()”. Basta para ello, con tener en el PCB de cada proceso un atributo que indique el nombre del programa que está ejecutando. Los segmentos de datos y pila no deben compartirse (no hay que hacer nada especial para ello. El de datos inicializados hay que leerlo del fichero ejecutable).

Accesos a disco: Podemos ahorrar algunos accesos si el programa ya está siendo ejecutado por otro proceso y el sistema guarda una copia de la cabecera del fichero ejecutable en memoria (para ese caso). Si el programa no está siendo ejecutado hay que efectuar, como mínimo, la lectura de la cabecera del ejecutable para obtener una descripción de su mapa de memoria. Normalmente no hay que leer nada más, pues en la actualidad se utilizan técnicas de memoria virtual y los “segmentos” de código, datos, pila y bibliotecas dinámicas se van asignando (y leyendo del disco) a medida que se necesitan.

21. Sea un sistema de memoria segmentado-paginado. Los espacios de direcciones lógicas son de 8Gbytes y el tamaño de página es de 2Kbytes. Un espacio de direcciones puede tener hasta 256 segmentos y los descriptores de página tienen 4 bytes. Diga cual será el tamaño de la tabla de páginas de un segmento.



**Solución**

La solución es 64Kb.

Será necesario conocer el número máximo de páginas que puede tener un segmento y multiplicarlo por el tamaño del descriptor de segmento. Para conocer el número máximo de páginas será necesario conocer el número de bits destinado a la página en la dirección lógica.

Los espacios de direcciones lógicas son de 8Gb  $\rightarrow$  Dir. lógicas de 33 bits.

El número máximo de segmentos de un proceso 256 segmentos  $\rightarrow$  8 bits de la dirección para el número de segmento.

El tamaño de página es de 2Kb  $\rightarrow$  11 bits de la dirección para el desplazamiento dentro de la página.

El número de bits para el n° de página vendrá dado por:  $33 - (8 + 11) = 33 - 19 = 14$

Con lo que los campos y bits de la dirección lógica quedarían:



Se tienen un total de 14 bits para la página y se necesitarán  $2^{14} = 16384$  descriptores de páginas por segmento.

El tamaño de la tabla de páginas =  $16384 * 4 \text{ bytes} = 65536 = 64 \text{ KB}$

22. Suponga que un proceso emite las direcciones lógicas (2,18004) (1, 2735) (5, 12219) utilizando un modelo de gestión de memoria basado en segmentación paginada y que estas tres direcciones lógicas producen respectivamente las tres siguientes direcciones físicas: 1108, 1327 y 1211.

- ¿Cuál es el número máximo de bits que se están utilizando para el desplazamiento dentro de una página?
- ¿Cuál es el número mínimo de bits que se están utilizando para indicar un número de página?
- ¿Cuál es el número mínimo de bits que se están utilizando para indicar un número de marco?

**Solución**

a) La solución es 7 bits.

Necesitamos conocer el tamaño de la página. Para ello el enunciado únicamente nos indica la correspondencia entre direcciones lógicas y físicas. Por lo tanto será necesario ir probando diferentes tamaños de páginas hasta encontrar uno donde los desplazamientos de la dirección física y la dirección lógica coincidan. Los tamaños de página han de ser potencia de dos y además ha de ser igual en todo el sistema. Comenzaremos probando por un tamaño de página de 1024 e iremos calculando el resto de dividir las respectivas direcciones por dicho tamaño. Si coinciden los desplazamientos será necesario probar con un tamaño mayor de página y si no coinciden probaremos con tamaños menores.

	Tamaño	
--	--------	--

	Pág. dirección	1024	512
lógica	18004	596	84
física	1108	84	84

Para esta dirección es válida un tamaño de página de 512 o menor. Ahora será necesario comprobarlo para el resto de direcciones.

	Tamaño Pág. dirección	512	128
lógica	2735	175	47
física	1327	303	47

Para esta dirección es válida un tamaño de página de 128 o menor. Ahora será necesario comprobarlo para la última dirección que nos proporciona el enunciado.

	Tamaño Pág. dirección	128
lógica	12219	59
física	1211	59

El tamaño máximo de página que puede estar utilizando este sistema es de 128 palabras, por tanto el número máximo de bits que se están utilizando para el desplazamiento es de 7 bits.

b) La solución es 8 bits.

Teniendo en cuenta el tamaño de página calculado en el apartado anterior se puede calcular el número de páginas que como mínimo contienen los segmentos de las direcciones lógicas referenciadas.

	Tamaño Pág. dirección	128
lógica	(2,18004)	$18004 / 128 = 140$ n° de página $18004 \bmod 128 = 84$ desplazamiento
lógica	(1,2735)	$2735 / 128 = 21$ n° de página $2735 \bmod 128 = 47$ desplazamiento
lógica	(5,12219)	$12219 / 128 = 95$ n° de página $12219 \bmod 128 = 59$ desplazamiento

Necesitamos un número de bits  $m$  para indicar las páginas de manera que  $2^m \geq 140$ , con lo que  $m=8$ .

c) La solución es 4 bits.

El tamaño de página y de marco ha de ser el mismo por tanto dividiendo la dirección física más alta por el tamaño de página calculado en el apartado a) obtendremos el número de marcos que como mínimo hay en el sistema.

	Tamaño marco dirección	128
Física	1108	$1108 / 128 = 8$ n° de marco $1108 \bmod 128 = 84$ desplazamiento