

# Fundamentos del Software

---

## Ejercicios Tema 2

---

**José Antonio de la Rosa Cubero e Higinio Paterna Ortiz**

---

### **Ejercicio 13. ¿Tiene sentido mantener ordenada por prioridades la cola de procesos bloqueados? Si lo tuviera, ¿en qué casos sería útil hacerlo?**

Sí, porque así los procesos más esenciales para el funcionamiento del sistema se ejecutarían antes que los menos prioritarios. Por ejemplo, un esquema así resulta útil en los sistemas embebidos de tipo *real time*, como pueda ser un coche autónomo. Tiene sentido que un proceso relacionado con la decisión de frenar el vehículo disponga de los recursos antes que uno destinado a controlar la temperatura. Otro ejemplo sería que en un PC tiene sentido que un proceso del sistema operativo tenga preferencia frente a una aplicación de usuario.

### **Ejercicio 14. ¿Por qué se utilizan potencias de dos para los tamaños de página, número de páginas en el espacio lógico de un proceso, y números de marcos de página?**

Principalmente para facilitar el direccionamiento. Una dirección lógica que haga referencia a una parte concreta de cualquiera de los elementos anteriores, puede traducirse directamente al espacio de direcciones físicos sin más que sumar una constante.

Proceso	Código	Pila	Datos
A	20480	14288	10240
B	16384	8200	8192
C	18432	13288	9216

**Ejercicio 19. Suponga que tenemos 3 procesos ejecutándose concurrentemente en un determinado instante. El sistema operativo utiliza un sistema de memoria con paginación. Se dispone de una memoria física de 131072 bytes (128K). Sabemos que nuestros procesos al ser ejecutados tienen los parámetros que se muestran en la tabla.**

Los datos indican el tamaño en bytes de cada uno de los segmentos que forman parte de la imagen del proceso. Sabiendo que una página no puede contener partes de dos segmentos diferentes (pila, código o datos), hemos de determinar el tamaño de página que debería utilizar nuestro sistema y se barajan dos opciones: páginas de 4096 bytes (4K) o páginas de 512 bytes (1/2K). Se pide:

- ¿Cuál sería la opción más apropiada, 4096 bytes o 512 bytes?. Justifica totalmente la respuesta mostrando todos los cálculos que has necesitado para llegar a dicha conclusión.

Supongamos páginas de 4096 bytes.

Calculamos cantidad de marcos de página que hay en la memoria:

$$\frac{131072 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} = 32 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso A para código :

$$\left\lceil \frac{20480 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 5.0 \rceil = 5 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso A para la pila:

$$\left\lceil \frac{14288 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 3.49 \rceil = 4 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso A para los datos:

$$\left\lceil \frac{10240 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 2.5 \rceil = 3 \text{ pags}$$

En total, el proceso A necesitaría 12 páginas.

Vamos a ver cuántas páginas necesita el proceso B para código :

$$\left\lceil \frac{16384 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 4.0 \rceil = 4 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso B para la pila:

$$\left\lceil \frac{8200 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 2.002 \rceil = 3 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso B para los datos:

$$\left\lceil \frac{8192 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 2.0 \rceil = 2 \text{ pags}$$

En total, el proceso B necesitaría 9 páginas.

Vamos a ver cuántas páginas necesita el proceso C para código :

$$\left\lceil \frac{18432 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 4.5 \rceil = 5 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso C para la pila:

$$\left\lceil \frac{13288 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 3.24 \rceil = 4 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso C para los datos:

$$\left\lceil \frac{9216 \text{ bytes}}{4096 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 2.25 \rceil = 3 \text{ pags}$$

En total, el proceso C necesitaría 12 páginas.

Los tres procesos necesitarían 33 páginas para ejecutarse concurrentemente, pero solo disponemos de 32, así que esto no es posible.

Suponemos ahora páginas de 512 bytes.

Calculamos cantidad de marcos de página que hay en la memoria:

$$\frac{131072 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} = 256 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso A para código :

$$\left\lceil \frac{20480 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 40.0 \rceil = 40 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso A para la pila:

$$\left\lceil \frac{14288 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 27.9 \rceil = 28 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso A para los datos:

$$\left\lceil \frac{10240 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 20.0 \rceil = 20 \text{ pags}$$

En total, el proceso A necesitaría 88 páginas.

Vamos a ver cuántas páginas necesita el proceso B para código :

$$\left\lceil \frac{16384 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 32.0 \rceil = 32 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso B para la pila:

$$\left\lceil \frac{8200 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 16.02 \rceil = 17 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso B para los datos:

$$\left\lceil \frac{8192 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 16.0 \rceil = 16 \text{ pags}$$

En total, el proceso B necesitaría 65 páginas.

Vamos a ver cuántas páginas necesita el proceso C para código :

$$\left\lceil \frac{18432 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 36.0 \rceil = 36 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso C para la pila:

$$\left\lceil \frac{13288 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 25.95 \rceil = 26 \text{ pags}$$

Vamos a ver cuántas páginas necesita el proceso C para los datos:

$$\left\lceil \frac{9216 \text{ bytes}}{512 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 18.0 \rceil = 18 \text{ pags}$$

En total, el proceso C necesitaría 80 páginas.

Los tres procesos necesitarían 233 páginas para ejecutarse concurrentemente, con lo que se quedan 23 páginas libres.

Es preferible utilizar páginas de 512 bytes porque nos van a permitir que los tres programas estén a la vez alojados en memoria principal.

**- ¿Cuál es el formato de cada entrada de la Tabla de Páginas con el tamaño de página elegido? Justifica el tamaño de los campos con direcciones. Puedes añadir los bits que consideres necesarios para el buen funcionamiento del sistema indicando para que van a ser utilizados.**

$$\log_2(256) = 8$$

Vamos a necesitar 8 bits para determinar el marco de página correspondiente.

Luego habrá en la tabla de páginas una entrada por cada página del proceso con entradas con 8 bits para el número de marco donde está alojada y algunos bits adicionales para otra información, como el modo de acceso o protección.

**- ¿Cuántas Tablas de Páginas habrá en este sistema?  
¿Cuántas entradas hay en cada tabla de páginas (filas)?**

En el sistema habrá 3 tablas de páginas, una por cada proceso. En cada una de esas tablas habrá tantas como páginas tenga el proceso, es decir, 68 para el proceso A, 65 para el B y 80 para el proceso C.

**Ejercicio 21. Estamos trabajando con un sistema operativo que emplea una gestión de memoria paginada. Cada página tiene un tamaño de 2.048 bytes. La memoria física disponible para los procesos es de 8 MBytes. Suponga que primero llega un proceso que necesita 31.566 posiciones de memoria (o bytes) y, después, llega otro proceso que consume 18.432 posiciones cuando se carga en memoria. Se pide calcular la fragmentación interna provocada en cada proceso.**

Supongamos que inicialmente la memoria está vacía.

Calculamos el número de páginas usadas por el primer programa:

$$\left\lceil \frac{31566 \text{ bytes}}{2048 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 15.41 \rceil = 16 \text{ pags}$$

Hacemos lo mismo para el segundo:

$$\left\lceil \frac{18432 \text{ bytes}}{2048 \frac{\text{bytes}}{\text{pag}}} \right\rceil = \lceil 9.0 \rceil = 9 \text{ pags}$$

Donde hemos redondeado hacia arriba el número de páginas porque no se pueden usar fraccionariamente.

Veamos de cuántos marcos de página dispone nuestra memoria:

$$\frac{8 \text{ MB}}{2048 \frac{\text{bytes}}{\text{pag}}} = \frac{8 \cdot 1024^2 \text{ bytes}}{2048 \frac{\text{bytes}}{\text{pag}}} = 4096 \text{ pags}$$

Como la suma de 16 y 9 es 25, que es menor que la cantidad total de páginas que puede albergar el sistema, tanto el primer como el segundo programa pueden ocupar la memoria a la vez sin problemas. En el caso del primer programa, se desperdicia parte de la memoria (aproximadamente un 60%) en la última página. Si el programa 2 termina antes que el uno no hay problema, simplemente se quedan libres las páginas posteriores a la última del primer programa.

El caso más interesante es si el primer programa acaba antes que el segundo. En ese caso, primeros dieciséis marcos de página y los últimos 4071 estarían libres. Pero esto no supondría un problema realmente, puesto que se trata de un modelo paginado: si un programa nuevo que entrara necesitase más de 16 páginas para funcionar, digamos 20, las primeras 16 tomarían los primeros 16 marcos de página, y las últimas 4 páginas se colocarían en los marcos posteriores a los que reside el programa dos. Esto ilustra como la paginación resuelve el problema de fragmentación de la memoria.