

# SISTEMAS OPERATIVOS – Grupo D – Curso 2015-16

## Ejercicios – Tema 4

En el libro

Aranda, J.; *Sistemas Operativos. Teoría y Problemas*; Ed. Sanz y Torres  
se pueden encontrar problemas resueltos. Se incluyen para su estudio los siguientes:

Cuestiones 5-4, 5-7, 5-10, Problemas 5-3, 5-5, 5-6, 5-7, 5-9 a 5-14  
(se encuentran entre las páginas 344 y 371)

1. En la siguiente figura se representa una tabla FAT. Al borde de sus entradas se ha escrito, como ayuda de referencia, el número correspondiente al bloque en cuestión. También se ha representado la entrada de cierto directorio. Como simplificación del ejemplo, suponemos que en cada entrada del directorio se almacena: Nombre de archivo/directorio, el tipo (F=archivo, D=directorio), la fecha de creación y el número del bloque inicial.

Nombre	Tipo	Fecha	NºBloque
DATOS	F	8-2-13	3

Tenga en cuenta que:

- el tamaño de bloque es de 512 bytes
- el asterisco indica último bloque
- todo lo que está en blanco en la figura está libre.

FAT			
1		10	
2		11	
3	15	12	
4		13	
5		14	
6		15	*
7		16	
8		17	
9		18	

Rellene la figura para representar lo siguiente:

- Creación del archivo DATOS1 con fecha 1-3-2013, y tamaño de 10 bytes.
- Creación del archivo DATOS2 con fecha 2-3-2013, y tamaño 1200 bytes.
- El archivo DATOS aumenta de tamaño, necesitando 2 bloques más.
- Creación del directorio D, con fecha 3-3-2013, y tamaño 1 bloque.
- Creación del archivo CARTAS con fecha 13-3-2013 y tamaño 2 kBytes.

2. Si se pierde el primer puntero de la lista de espacio libre, ¿podría el Sistema Operativo reconstruirla? ¿Cómo?

3. Sobre conversión de direcciones lógicas dentro de un archivo a direcciones físicas de disco. Estamos utilizando únicamente la estrategia de indexación a tres niveles para asignar espacio en disco. Tenemos que el tamaño de bloque es igual a 1024 bytes, y el tamaño de puntero es de 4 bytes. Se recibe la solicitud por parte de un proceso de usuario de leer el carácter número N de determinado archivo. Suponemos que ya hemos leído la entrada del directorio asociada a este archivo, es decir, tenemos en memoria los datos PRIMER-BLOQUE y TAMAÑO. Calcule la sucesión de direcciones de bloque que se leen hasta llegar al bloque de datos que posee el citado carácter.

4. ¿Cuál es el tamaño que ocupan todas las entradas de una tabla FAT32 que son necesarias para referenciar los cluster de datos de una partición de 20 GB ocupada exclusivamente por la propia FAT32 y dichos cluster de datos? Tamaño de cluster = 16 KB; tamaño de entrada de una FAT32 = 32bits

5. Cuando se abre el archivo `/home/lopez/f1` es necesario realizar varios accesos a disco. Calcule el número de accesos a disco requeridos (como máximo) bajo la suposición de que el i-nodo raíz ya se encuentra en memoria y que todos los directorios necesitan como máximo 1 bloque para almacenar sus entradas.

6. Nos situamos en la teoría explicada sobre generalizades de los sistemas Unix. Supongamos que un proceso **P1** abre el archivo "**datos**" en modo **lectura/escritura** y otro proceso **P2** abre el mismo archivo y con el mismo modo, y a continuación crea un proceso hijo que abre el archivo `"/home/lopez/f1"` en modo **lectura/escritura**. Represente toda la información relevante sobre el estado de las tablas de descriptores de archivos, tabla de archivos y tabla de i-nodos después de dichas operaciones.

7. En Unix, ¿qué espacio total (en bytes) se requiere para almacenar la información sobre la localización física de un archivo que ocupa 3 Mbytes?. Suponga que el tamaño de un bloque lógico es de 1 Kbytes y se utilizan direcciones de 4 bytes. Justifique la solución detalladamente.

8. En Unix, ¿qué espacio total (en bytes) se requiere para almacenar la información sobre la localización física de un archivo que ocupa 3 Mbytes?. Suponga que el tamaño de un bloque es de 1 Kbytes y se utilizan direcciones de 4 bytes

9.

Como proyecto de fin de asignatura unos estudiantes han diseñado un sistema de archivos parecido al del s.o. de Unix con una estructura básica de 20 bloques de 8 bytes cada uno, mostrada en la Figura 5.41.

Bloque de arranque	Super-bloque	Tabla de nodos- <i>i</i>	Tabla de bloques
--------------------	--------------	--------------------------	------------------

Figura 5.41: Estructura del sistema de archivos.

Las características de cada parte de esta estructura son las siguientes:

- El **bloque de arranque** (bloque 0) está reservado al programa de carga inicial. El bloque cero no tiene significado dentro del sistema de ficheros. La información efectiva del sistema comienza realmente en el bloque uno del dispositivo.
- El **super-bloque** (bloque 1) contiene la información más importante acerca del sistema de ficheros como el tamaño de la tabla de nodos-*i* dado en número de bloques (1 byte), un mapa de bits que indica que nodos-*i* están ocupados y cuales están libres (1 byte), el tamaño de la tabla de bloques, dado en número de bloques (1 byte), un mapa de bits que indica que bloques están ocupados y cuales están libres (1 byte), el número del nodo-*i* que apunta al directorio raíz (1 byte), etc.
- La **tabla de nodos-*i*** (bloques 2 al 9) que corresponde a ocho nodos-*i* numerados de 0 a 7 cada uno de un bloque de tamaño y con la información siguiente: El tipo de fichero (1 byte) (0 si es un fichero/1 si es un directorio), el tamaño del fichero dado en bytes (1 byte), la tabla de asignación de bloques (6 bytes) cuyos elementos son punteros a la tabla de bloques para indicar los bloques que están ocupados por el fichero. Los elementos no utilizados tendrán el valor de NIL.
- La **tabla de bloques** son los bloques que contienen los datos realmente almacenados en los directorios y archivos. Si se trata de un fichero contendrá los datos (el último bloque contiene el caracter ^C). Si se trata de un directorio, tendrá una entrada por cada archivo del directorio con la información: La longitud del nombre del archivo (1 byte), el nombre del mismo (tantos bytes como se indique en el campo anterior), el número del nodo-*i* que corresponde al fichero (1 byte).

En la figura siguiente se muestra el estado de archivos en cierto momento inicial.

a) Realice los cambios adecuados sobre la figura para representar que dentro del directorio raíz (es decir como hijo del raíz) se ha creado *da2015* como enlace duro al archivo *file1*

<b>Bloque de arranque</b>								
<b>Super-bloque</b>	8	11100000	10	11100000	00000000	0		
<b>Tabla de nodos-i / 0</b>	1	6	0	NIL	NIL	NIL	NIL	NIL
<b>1</b>	1	7	1	NIL	NIL	NIL	NIL	NIL
<b>2</b>	0	7	2	NIL	NIL	NIL	NIL	NIL
<b>3</b>								
<b>4</b>								
<b>5</b>								
<b>6</b>								
<b>7</b>								
<b>Tabla de bloques - 0</b>	4	d	i	r	l	l		
<b>1</b>	5	f	i	l	e	l	2	
<b>2</b>	d	a	t	o	s	l	^C	
<b>3</b>								
<b>4</b>								
<b>5</b>								
<b>6</b>								
<b>7</b>								
<b>8</b>								
<b>9</b>								

b) Realice los cambios adecuados sobre la figura para representar la creación del archivo */dir1/feb15* de contenido *"Aproximacion al espacio"*. Partimos de la situación inicial representada en la figura.

<b>Bloque de arranque</b>								
<b>Super-bloque</b>	8	11100000	10	11100000	00000000	0		
<b>Tabla de nodos-<i>i</i> / 0</b>	1	6	0	NIL	NIL	NIL	NIL	NIL
<b>1</b>	1	7	1	NIL	NIL	NIL	NIL	NIL
<b>2</b>	0	7	2	NIL	NIL	NIL	NIL	NIL
<b>3</b>								
<b>4</b>								
<b>5</b>								
<b>6</b>								
<b>7</b>								
<b>Tabla de bloques - 0</b>	4	d	i	r	l	l		
<b>1</b>	5	f	i	l	e	l	2	
<b>2</b>	d	a	t	o	s	l	^C	
<b>3</b>								
<b>4</b>								
<b>5</b>								
<b>6</b>								
<b>7</b>								
<b>8</b>								
<b>9</b>								