

# Sistemas Operativos

## 1º Recuperatorio 2º Parcial 1C2018 - TM/TT - Resolución

*Aclaración: La mayoría de las preguntas o ejercicios no tienen una única solución. Por lo tanto, si una solución particular no es similar a la expuesta aquí, no significa necesariamente que la misma sea incorrecta. Ante cualquier duda, consultar con el/la docente del curso.*

### Teoría

1. V o F
  - a. Falso, serían 2 inodos, los hardlinks apuntan al mismo inodo mientras que el softlink es un archivo independiente.
  - b. Verdadero. Deberíamos guardar los páginas siempre en el mismo lugar y si no está disponible no se podría ejecutar yendo en contra de cómo funciona memoria virtual
2. En caso de poder seguir realizando un trabajo útil sin el resultado de la e/s. Se podría implementar en una espera activa preguntando por el valor de retorno de la e/s no bloqueante hasta que esté el resultado.
3. En el esquema de TP normal tendríamos 3 Tp mientras que en TP invertida sería una tabla para todo el sistema. En el esquema tradicional, se accedería por índice con el NRO de pág en la TP de dicho proceso. En la invertida debería recorrer toda la TP hasta encontrar el nro de pág (el índice sería el frame asignado). Se podría usar una tabla de hash para optimizarlo.
4. En caso de UFS probablemente tenga al menos 2 ptrs directos, por lo que habría que acceder una vez al inodo para recuperar la dirección y luego al bloque de datos. En FAT habría que acceder una vez a la FAT para recuperar el ptr del 2do bloque y luego se accedería directamente al cluster.
5. Sirve para poder indicar permisos sobre un archivo para distintos grupos y usuarios. Una partición de SWAP debe ser lo más simple posible para ser eficiente por lo que es probable que no sea de utilidad un ACL. Además, sólo el SO accede a la misma.

### Practica

#### Ejercicio 1

1. En principio a través de 4: el softlink "luis" que apunta a "miguel" que es un hardlink a archivo en cuestión, y los 3 hardlinks indicados por la columna referencias .
2. No, el archivo en cuestión es un directorio (lo indica la "d") por lo que no se puede realizar hardlinks sobre el mismo. La columna referencia indica las referencias con sus directorios hijos (quiere decir que tiene una entrada de tipo directorio.
3. Por lo que se ve en la tabla, el archivo the\_dir tiene 10 bloques de datos, por lo que se estaría accediendo con ptrs directos (10 al menos). El archivo miguel tiene 150 bloques de datos, por lo que hay 3 accesos extras correspondientes a bloques de punteros.  
Esto quiere decir que si tiene al menos 10 ptrs directos, el bloque de punteros puede direccionar menos de 140 (150-10) bloques ya que sino sería suficiente con un acceso extra a BP. Una opción es que cada BP pueda direccionar 128 bloques  
Por lo tanto, lo que puede estar ocurriendo es que se acceda al ptr indirecto simple (1 BP) y al primer BP del indirecto doble ( 2 BP de 1er y 2do nivel).

Posible configuración inodo: 10 ptrs directos , 1 ind simple, 1 ind doble

4. El FS es de  $2 \text{ TiB} = 2^{41} = 2^{\text{dirección}} * \text{TamBloque}$   
 Como sabemos que las direcciones son potencia de 2, una posible dirección es de 32 bits, dejando bloques de  $2^9 = 512 \text{ bytes}$   
 Esto tiene sentido ya que dijimos que cada bloque puede direccionar 128 bloques  $= 2^9 / 2^2$   
 Entonces, el archivo ocupa 512 bytes

## Ejercicio 2

Son 14 bits de direccionamiento.

Si 2930h = Página 5 quiere decir que de los 16 bits que tiene este Hexa solo tomo los 14 bits

2930 = 0010 - 1001 - 0011 - 0000 por lo cual al tomar los 14 bits = 10 - 1001 - 0011 - 0000

Son los 3 primeros bit para Numero de página 5 = 101 y 11 bits restante para el desplazamiento  $2^{11} = 2048 = 2 \text{ Kb}$ .

### a) Proceso A:

0A34h = Pág 1 (E) = 0000 - 1010

3810h = Pág 7 (L) = 0011 - 1000

07AEh = Pág 0 (E)

990Ah = Pág 3 (E)

D61Bh = Pág 2 (E)

### Proceso B:

90A2h = Pág 2 (E) = 1001 - 0020

8A10h = Pág 1 (E) = 1000 - 1010

28A3h = Pág 5 (L)

Proceso A							
Frame	Página	Bits	Pág 1 (E)	Pág 7 (L)	Pág 0 (E)	Pág 3 (E)	Pág 2 (E)
0	5-->	UM	5M	5M	0UM	0UM	0UM
2	0	M	1UM	1UM	1UM-->	1UM-->	1UM
3	4	U	4U -->	4U -->	4	4	2UM
6	7	UM	7UM	7UM	7M	7M	7M-->
7	3	U	3U	3U	3	3UM	3UM
Cantidad: PF			PF	No	PF	No	PF
Escrituras A didsco			E= Pág 0	No	E= Pág 5	No	NO

Proceso B					
Frame	Página	Bits	Pág 2 (E)	Pág 1 (E)	Pág 5 (L)
1	3-->	M	2UM	2UM	2UM
4	5	UM	5UM-->	5UM-->	5UM-->
5	1	M	1M	1UM	1UM
Cantidad: PF			PF	No	NO
Escrituras A didsco			E= Pág 3	No	NO

## Ejercicio 3

Capacidad disco =  $2^{27}$  = cabezales \* sectores por pista \* cilindros \* tamaño sector

$2^{27} = 2^2 * 2^5 * \text{cilindros} * 2^{10}$

$2^{27}/2^{17} = \text{cilindros} = 2^{10} = 1024 \text{ cilindros (0-1023)}$

El disco está subiendo desde 50

Con N-STEP-SCAN:

En  $T = 0$  llegaron 100 y 60, se cierra esa primera cola,

Se atienden: 50 | 60 - 100

Estamos en  $T = 100$  ms, llegaron en orden: 80 -102 -100 - 100 , como la cola es de 3 atendiendo los primeros 3

100 | 100 - 102 - (1023, TOPE) Cambio de sentido .- 80

Ya llegaron el resto de los pedidos: 100 - 10 - 10

80 | 10 - 10 - (0) Cambio de sentido - 100

Orden atención pedidos: (50) - 60 - 100 - 100 - 102 - (1023) - 80 -10 - 10 - (0) - 100

Tiempo total:  $(1023 - 50 + 1024 + 100) * 2 \text{ ms} = 4,194 \text{ ms}$

Con C-SCAN

En  $T = 0$  llegaron 100 y 60, Atiendo primero a 60

$T = 20 \text{ ms}$  llegaron 80 y 102, atiendo al primero

$T = 60 \text{ ms}$  quedan pendientes 100 y 102 y llegaron 100 y 100

Atiendo 100- 100 - 100- 102

$T = 104 \text{ ms}$  no hay nada pendiente

En  $T = 110 \text{ ms}$  llega 10 , lo voy a atender voy hasta el tope y cae hasta cero y luego sube al 10

$T = 1972 \text{ ms}$  ya está en otro 10, lo atiendo

$T_{\text{final}} = 1972 \text{ ms}$

T total de atención, podrían restar el tiempo en el que no se atendió a ninguno  $T_{\text{final}} - 6$



