

Proyecto Terminal - Resumen

Básico:

- 'particion.bin' → fichero binario que simula la partición de disco
- Partición de 100 bloques de 512 bytes cada uno.
- No hay bloque de arranque
- Bytemaps → Array de bytes (bloques e inodos: si X byte vale 1, está ocupado)
 - Bloques: 100 elementos
 - 0: superbloque
 - 1: bytemap
 - 2: lista inodos
 - 3: directorio
 - 4 – 99: bloque de datos
 - Inodos: 24 elementos
 - 0 - 1: reservados
 - 2: directorio
 - 3 – 23: libres

Enunciado:

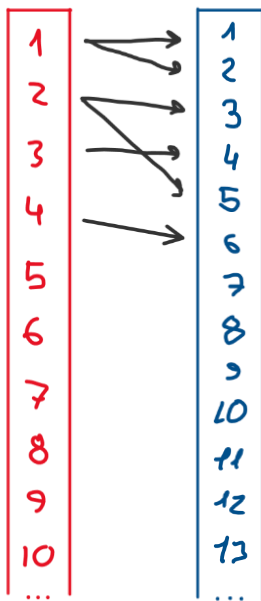
- Se pide desarrollar un 'simul_ext.c' que lea el contenido de 'particion.bin' en un array de bytes en memoria y entre en un bucle infinito de procesamiento de comandos.
- El prompt del usuario debe ser '>> '.
- Comandos ejecutables:
 - 'info' → Info. superbloque
 - 'bytemaps' → Muestra bytemap inodos y bloques (25 prim.)
 - 'dir' → Lista ficheros (menos entrada directorio raíz)
 - 'rename' → No cambia inodos ni listas (err*)
 - 'imprimir' → Imprime contenido (⚠ más de 1 bloque, err*)
 - 'remove' → update bytemaps y inodos a vacíos (no elim.)
 - 'copy' → copiar al primer inodo y bloque libre (err*)
 - 'salir' → termina la sesión
 - Comando desconocido (ERROR: Comando ilegal [...])

err*: Puede devolver un error
(princ. con los nombres)

Esquema:

- 100 bloques → 96 libres
- 24 inodos → 20 ficheros (sale así en 'cabeceras.h').
- 1 inodo → máximo 7 bloques (de 2⁹ Bytes) → máximo 3'584 KB.

Hay una lista con los 20 inodos hábiles y otra para los 96 bloques. Luego cada inodo tiene un array de 7 elementos con los bloques que lo componen (en el ejemplo se ve que pueden no estar en orden).



Incongruencias:

- No he visto constancia de un comando para crear los ficheros.
- Los inodos no deberían poder tener bloques mezclados si no hubiera un comando para modificar el contenido / ampliarlo a posteriori.
- Se supone que solo hay 3 inodos reservados/usados, pero en 'cabeceras.h' aparece máximo 20 ficheros