

# Diseño de Sistemas Operativos

## Curso 2000- 2001

### Prácticas iniciales con llamadas al sistema (subsistema de ficheros)

1. Realizar un programa que realice la copia de ficheros. La sintaxis del programa es:  
Copia <fichero origen> <fichero destino> <número de bytes>  
La copia se realizará leyendo y escribiendo en cada iteración, tantos datos como los especificados en “número de bytes”. Además, el programa mostrará por pantalla los tiempos de ejecución del programa, tanto en modo usuario como en modo kernel (llamada *getrusage*). Deducir el tamaño óptimo de transferencia.
2. Extraer la información de un inodo. La sintaxis del programa sería:  
inodo <nombre fichero>  
El programa tomará el nombre del fichero que se le pasa por argumento y obtendrá la información del inodo correspondiente (llamada *stat*). Tras mostrar la información general (propietario, grupo, permisos, tipo de fichero, tiempos de acceso, modificación del fichero y modificación de inodo, número de inodo, nº del sistema de ficheros donde reside el fichero, tamaño del fichero, número de bloques usados y número de enlaces (links)), analizará el tipo de fichero, actuando de la siguiente forma:
  3. *fichero* regular: mostrar un pequeño menú que permita cambiar los permisos, propietario, renombrar el fichero, o borrarlo previa confirmación.
  4. *Directorio*: Mostrar un listado de los contenidos del directorio (funciones *opendir*, *readdir*).
  5. *Fichero tipo especial (dispositivo)*: Mostrar los números mayor y menor representativos de este dispositivo.
6. Implementar comando *df* con la llamada al sistema *ustat*. La sintaxis sería:  
midf <dispositivo representativo del sistema de ficheros>
7. Usando la llamada al sistema *fcntl*, implementar una base de datos sencilla que permita el acceso concurrente a ella. La base de datos constará de una serie de registros donde la información que contenga cada uno de ellos consistirá de un identificador y de datos opcionales (nombre, dirección,...). El programa nos mostrará un menú que nos permita:
  - a. Crear base de datos
  - b. Abrir base de datos
  - c. Buscar registro
  - d. Insertar registro
  - e. Modificar registro
  - f. Borrar registro
  - g. Cerrar base de datos.Crear las funciones que nos permita realizar cada una de las operaciones anteriores, teniendo en cuenta que la base de datos es concurrente (pueden acceder simultáneamente varios procesos a ella). Por ello, es preciso aplicar el bloqueo correspondiente **por registro** dependiendo de la función a realizar, es decir, un bloqueo de lectura cuando sólo se pretenda leer el registro y un bloqueo de escritura cuando se pretenda modificar, insertar o borrar un registro.  
Nota: Con el fin de apreciar el correcto funcionamiento del programa, insertar pausas (*getchar*) antes de desbloquear los registros.  
En las fotocopias se puede ver la sintaxis de la llamada al sistema *fcntl*.