

# PRÁCTICA 5

## Diego Gonzalez Sanz y Carlos Bermúdez Expósito

Introducción: En esta práctica vamos a ver información sobre el funcionamiento de la gestión de archivos y directorios

Objetivos: Los objetivos en esta práctica es comprender la gestión de archivos y directorios, además de estudiar las funciones aportadas en este documento para comprender su funcionamiento y sus diferentes variaciones de poner un dato u otro.

### Ejercicio 1:

En el primer ejercicio creamos una variable que se llama file que nos abre el fichero y le da los permisos de que el usuario (dueño) permisos para leer y escribir. Hemos puesto los permisos para que lea el contenido del fichero y hemos dado los permisos de escribir para que pueda modificar el fichero y hacer pruebas. Al grupo al que pertenece el propietario le hemos dado solo el permiso de leer, y por último a otros que son otros usuarios solo que puedan leerlo y no modificarlo ya que no nos interesa.

Después nos hacemos una variable tamañoFichero que es el tamaño del fichero que lo hacemos con la función lseek. Después debemos de hacernos la variable restoDeCaracteres con la función malloc que lo empleamos para asignar un bloque de memoria dinámica. Después hacemos un lseek con el segundo parámetro a 50 y el tercer parámetro al inicio del archivo para que al leer el archivo en la siguiente línea hasta la posición 50 no empiece a leer. En la función read como primer parámetro es la ruta del fichero, el segundo parámetro desde qué posición se van a leer y el tercer parámetro hasta qué carácter se van a leer. Por último, se debe cerrar el fichero con la función close para liberar los recursos que tengamos asignados.

```

ejercicio1.c: In function 'main':
ejercicio1.c:15:25: warning: implicit declaration of function 'lseek'; did you mean 'fseek'? [-Wimplicit-function-declaration]
   15 |     int tamanoFichero = lseek(file, 0, SEEK_END);
      |                        ^~~~~~
      |                        fseek
ejercicio1.c:19:5: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]
   19 |     read(file, restoDeCaracteres, tamanoFichero);
      |     ^~~~
      |     fread
ejercicio1.c:23:5: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]
   23 |     close(file);
      |     ^~~~~
      |     pclose
diego@diego-VivoBook-ASUSLaptop-X415DA-M415DA: ~/Sistemas-Operativos$ ./ejecutable1
Desde la posicion 50 hasta el final del fichero esta la siguiente informacion: que pretenden ser prácticas, ha ido enmendando la plan
a a la Naturaleza y convirtiéndola en campo. El hombre, paso a paso, ha hecho su paisaje, amoldándolo a sus exigencias. Con esto, el
campo ha seguido siendo campo, pero ha dejado de ser Naturaleza. Mas, al seleccionar las plantas y animales que le son útiles, ha emp
obrecido la Naturaleza original, lo que equivale a decir que ha tomado una resolución precipitada porque el hombre sabe lo que le es
útil hoy, pero ignora lo que le será útil mañana. Y el aceptar las especies actualmente útiles y desdenar el resto supondría, según n
os dice Faustino Córdón, sacrificar la friolera de un millón de especies animales y medio millón de especies vegetales, limitación in
concebible de un patrimonio que no podemos recrear y del que quizá dependieran los remedios para el hambre y la enfermedad de mañana.
Así las cosas, y salvo muy contadas reservas, apenas queda en el mundo Naturaleza natural
diego@diego-VivoBook-ASUSLaptop-X415DA-M415DA: ~/Sistemas-Operativos$

```

Aquí leemos desde la posición 50 y el fichero contiene esta información. Está frase “El hombre, desde su origen, guiado por unas miras”, son los primeros 50 caracteres

1 El hombre, desde su origen, guiado por unas miras que pretenden ser prácticas, ha ido enmendando la plana a la Naturaleza y convirtiéndola en campo. El hombre, paso a paso, ha hecho su paisaje, amoldándolo a sus exigencias. Con esto, el campo ha seguido siendo campo, pero ha dejado de ser Naturaleza. Mas, al seleccionar las plantas y animales que le son útiles, ha empobrecido la Naturaleza original, lo que equivale a decir que ha tomado una resolución precipitada porque el hombre sabe lo que le es útil hoy, pero ignora lo que le será útil mañana. Y el aceptar las especies actualmente útiles y desdenar el resto supondría, según nos dice Faustino Córdón, sacrificar la friolera de un millón de especies animales y medio millón de especies vegetales, limitación inconcebible de un patrimonio que no podemos recrear y del que quizá dependieran los remedios para el hambre y la enfermedad de mañana. Así las cosas, y salvo muy contadas reservas, apenas queda en el mundo Naturaleza natural

## Ejercicio 2:

En el ejercicio 2 nos piden leer un fichero y crear otro escribiendo en él, el contenido del leído, para realizarlo primero hemos abierto mediante fopen el fichero de lectura llamado leer.txt en modo lectura (“r”) y justo después nos hemos creado mediante la misma función un fichero llamado escribir2.txt en modo escritura (“w”). Entonces para pasar el contenido de uno a otro nos hemos creado una variable que cuenta todos los caracteres (contCaracteres) los cuales se almacenan en un buffer que mediante un malloc nos reservamos memoria para que quepa todo el contenido, en este malloc le pasamos el número de caracteres \* sizeof(char) este ultimo nos calcula el tamaño en bytes pero le forzamos a char, el resultado de esto es el contenido del fichero leer.txt en el buffer, luego hacemos un rewind para mover el puntero al inicio del fichero otra vez, luego nos leemos todo el contenido del buffer mediante fread y gracias al uso posterior de fwrite nos lo escribimos en escribir2.txt. Y para finalizar con un printf del buffer nos lo mostramos por pantalla y finalmente liberamos el contenido del buffer con free y cerramos los archivos con fclose.

Luego en cuanto a los permisos hemos hecho que el de lectura (leer.txt) sea rw para el usuario para poder realizar pruebas y comprobar su posterior funcionamiento en el de escribir y para todo lo demás a read, todo esto realizado desde propiedades y comprobado con ls -li.

Y luego el que nos creamos (escribir2.txt) nos lo creamos a rw para usuario y grupo y para los demás a read, esto lo hacemos para cubrir más casos posibles y comprobar su funcionamiento.

```

diego@diego-VivoBook-ASUSLaptop-X415DA-M415DA:~/Sistemas-Operativos$ gcc -o ejecutable2 ej2.c
diego@diego-VivoBook-ASUSLaptop-X415DA-M415DA:~/Sistemas-Operativos$ ./ejecutable2
El perro de pepe
diego@diego-VivoBook-ASUSLaptop-X415DA-M415DA:~/Sistemas-Operativos$ ls -li
total 60
275995 drwxr-xr-x 2 diego diego 4096 dic  9 19:19 Directorio_Ej3
275948 -rw-rw-r-- 1 diego diego  583 dic 15 10:16 ej2.c
263024 -rwxrwxr-x 1 diego diego 16264 dic 15 09:39 ejecutable1
272318 -rwxrwxr-x 1 diego diego 16296 dic 15 10:34 ejecutable2
275959 -rw-rw-r-- 1 diego diego  661 dic 15 09:39 ejercicio1.c
275972 -rw-r--r-- 1 diego diego  2227 dic  9 19:45 ejercicio3.c
275958 -rw-rw-r-- 1 diego diego   18 dic 15 10:34 escribir2.txt
275465 -rw-r--r-- 1 diego diego  1013 dic 15 08:43 ficheroEjercicio1S0.txt
275989 -rw-r--r-- 1 diego diego   18 dic 15 10:16 leer.txt
diego@diego-VivoBook-ASUSLaptop-X415DA-M415DA:~/Sistemas-Operativos$ █

```

### Ejercicio 3:

En el ejercicio 3 hemos pasado por argumento el directorio a sumar con `opendir`, luego nos creamos un `struct dirent` que apunta a cada archivo y otro llamado `stat` para guardar los atributos de estos mismos, luego nos creamos una variable `int` para la suma y gracias a `readdir` nos leemos el directorio recorriendolo y realizando su respectiva suma, y finalmente nos imprimimos la suma.

¿Qué ocurre cuando hay dos enlaces hard que hacen referencia al mismo fichero?

Un enlace duro es un archivo que apunta al mismo contenido almacenado en disco que el archivo original. Esto significa que los archivos originales y los enlaces duros dispondrán del mismo inodoro y ambos estarán apuntando al mismo contenido almacenado en el disco duro.

En el caso de que tengamos dos enlaces duros que se llamen "fichero1.txt" y "fichero2.txt", ambos enlaces apuntan a los mismo datos ficheros, si en el fichero1.txt modificas un dato y lo guardas esos cambios serán visibles para el archivo original y para el otro enlace duro, y pasa también en la viceversa

Conclusión: Hemos llegado a la conclusión de que ya entendemos todos los permisos que pueden tener el usuario, un creador de los archivos y todos los usuarios, además de enlaces hard.