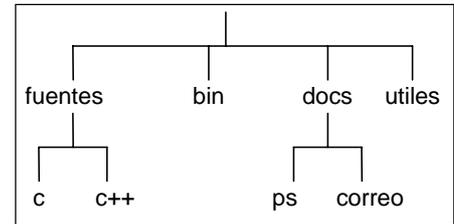


- 1.- Establezca una palabra de paso para su cuenta. Anótelas en lugar seguro.
- 2.- Limpiar el contenido del directorio de conexión (ejecutando la orden **rm -r \***).
- 3.- Hacer un esquema que refleje la estructura del árbol de los principales directorios del sistema hasta el tercer nivel.
- 4.- Averiguar de todas las formas que conozca (al menos tres) cuál es su directorio **HOME**.
- 5.- Cree un subdirectorio que se llame **pruebas** cuyo padre sea el directorio raíz. ¿Qué ocurre y por qué?
- 6.- Cree el siguiente árbol de directorios a partir de su directorio **HOME**:
- 7.- ¿Qué secuencia de órdenes ha utilizado?
- 8.- Sitúese en el subdirectorio **utiles** e intente borrarlo desde él mismo. ¿Qué ocurre?
- 9.- Desde el subdirectorio **utiles**, sitúese en el subdirectorio **c** utilizando direccionamiento relativo.
- 10.- Indique todas las formas posibles para situarse desde el subdirectorio **c** a su directorio **HOME**.
- 11.- Sitúese en el subdirectorio **fuentes** usando direccionamiento absoluto.
- 12.- Intente borrar, desde el subdirectorio **fuentes**, el subdirectorio **docs** usando la orden **rmdir**. ¿Qué ocurre y por qué?
- 13.- Muestre el contenido del directorio padre del directorio raíz. ¿Qué ocurre?
- 14.- Sitúese en el subdirectorio **/dev** y liste el contenido del mismo.
- 15.- Sitúese en su directorio **HOME** y liste el contenido del subdirectorio **/dev** en formato largo. ¿Qué indican los caracteres **c** y **b** que aparecen a la izquierda de los permisos de acceso a los ficheros?
- 16.- Liste todos los ficheros, incluidos aquellos cuyo nombre comienza por punto (**.**), del fichero raíz en formato largo. Indique cuántos ficheros, enlaces y directorios cuelgan del directorio raíz.
- 17.- Indique cuántos subdirectorios tiene el subdirectorio **/etc** de todas las formas que conozca.
- 18.- Averigüe los usuarios que están conectados al sistema en este momento.
- 19.- Averigüe cuántos usuarios tienen cuenta en el sistema.
- 20.- Visualice el contenido del fichero **/etc/hosts**.
- 21.- Copie el fichero **/etc/hosts** en su subdirectorio **docs**.
- 22.- Calcule el número de líneas, palabras y caracteres del fichero **hosts**.
- 23.- Busque aquellas líneas del fichero **/etc/passwd** que contengan la subcadena **sso3**.
- 24.- Copie, desde su directorio **HOME**, el fichero **hosts** del directorio **docs** en los directorios **ps** y **utiles**.
- 25.- Cree en su directorio **HOME** un fichero que denomine **mppc.c** con el siguiente contenido usando la orden **cat**:

```
#include <stdio.h>
main ()
{
    printf("Este es Mi Primer Programa en C\n") ;
}
```

- 26.- Mueva este fichero al subdirectorio **c**.
- 27.- Sitúese en el directorio **c++** y haga un enlace al fichero **mppc.c** del directorio **c** con el mismo nombre.
- 28.- Haga otro enlace pero con el nombre **mppc2.c**.
- 29.- Compruebe cuántos enlaces tiene el fichero **mpp2.c** y cuántos el **mppc.c** del directorio **c**.
- 30.- Compruebe que se trata del mismo fichero observando su número de nodoi (ver opciones de **ls**).
- 31.- Elimine el fichero **mppc.c** del subdirectorio **c++**. ¿Cuántos enlaces existen ahora del fichero **mpp2.c**?
- 32.- Añada la siguiente línea al final del fichero **mppc.c** con la orden **cat >>**:
 

```
/* Esto es un comentario en c */
// Esto es un comentario en c++
```
- 33.- Compruebe que esta modificación afecta también al fichero **mpp2.c**.
- 34.- Elimine el fichero original y observe cómo se sigue accediendo al fichero con el enlace restante.
- 35.- Compruébese que el fichero es el mismo por su número de inodo.
- 36.- Cree un fichero de texto, llamado **texto**, y haga un enlace simbólico al mismo.
- 37.- Compruebe que se trata de ficheros distintos obteniendo sus números de nodoi.
- 38.- Elimine el fichero original y compruebe que ya no es posible acceder al contenido del mismo mediante el enlace.
- 39.- Cree un nuevo un fichero, llamado también **texto**. ¿Qué pasa si mostramos el contenido del enlace anterior?
- 40.- Sitúese en subdirectorio **fuentes**. Borre su contenido usando la orden **rm**.
- 41.- Ordene de forma descendente el fichero **/etc/hosts**, seleccione las cinco primeras líneas y guárdelas en un fichero en el subdirectorio **correo**.
- 42.- Cree una colección de ficheros con los nombres del ejemplo visto en clase (punto 5.2.2.4, pág. 6) y compruebe todas las combinaciones que allí se muestran para referencias ambiguas. Nota. Algunas combinaciones sólo son válidas en el Bourne Shell. Para iniciar un shell de Bourne, lance la orden **sh**, para volver al c shell, haga **exit**.
- 43.- Muestre todos los ficheros del subdirectorio **/bin** que comiencen por **l**. ¿Qué ocurre?
- 44.- Igual que el anterior pero que comiencen por **c** y tengan al menos dos caracteres.
- 45.- Igual que el anterior pero que comiencen por **d**, **c** ó **l** y terminen por **s** ó **d**.



- 46.- Igual que el anterior pero que comiencen por **d**, **c** ó **l**, terminen por **s** ó **d** y ocupen más de 20 bloques.
- 47.- Liste los nombres de los ficheros que cuelgan a partir del directorio `/usr/bin` cuyo nombre comienza por **p**.
- 48.- Obtenga los subdirectorios del sistema que han sido modificados en los últimos 20 días. ¿Por qué aparecen mensajes de error?.
- 49.- Muestre el contenido del directorio `HOME` del usuario `antonio`.
- 50.- Obtenga los nombres de los ficheros que son propiedad del superusuario (`root`).
- 51.- Obtenga los nombres de ficheros del directorio `/usr` cuyo nombre empiece por **s** y tengan un tamaño inferior a 20 bloques.
- 52.- Obtenga los nombres de ficheros del directorio `/usr` cuyo nombre empiece por **s** o tengan un tamaño inferior a 20 bloques, seleccione los 10 primeros y almacénelos en un fichero.
- 53.- Supongamos que tenemos un fichero sobre el que existen varios enlaces duros (p.e, el usado en los ejercicios sobre enlaces). ¿Cómo podríamos, con una única orden eliminar dicho fichero y todos sus enlaces?
- 54.- Obtenga un listado largo de los ficheros del sistema cuyo nombre contenga la subcadena `as`.
- 55.- Calcule el espacio en disco que ocupa el subdirectorio `/usr`.
- 56.- Calcule el espacio en disco que ocupa su directorio `HOME`.
- 57.- Calcule el espacio libre y ocupado del sistema de ficheros.
- 58.- Determine los tipos de los ficheros del directorio raíz.
- 59.- Determine los tipos de los ficheros de los subdirectorios `/bin`, `/etc` y `/dev` y almacene el resultado en un fichero que se llame `tipos.txt` en su directorio `HOME`.
- 60.- Copie el fichero `tipos.txt` al fichero `fichs.txt` en el subdirectorio `correo` usando la orden `cat`.
- 61.- Calcule cuántas entradas tienen los subdirectorios `/bin` y `/etc`.
- 62.- Calcule cuántos directorios hay en el sistema.
- 63.- Calcule cuántos ficheros hay en el sistema. Ejecute la orden necesaria en background y redireccione el resultado al fichero `numero` de su directorio `HOME`. ¿Cuándo se sabe que se ha terminado la orden?. Nota. Lanzar la orden en un shell Bourne y redirigir la salida de error.
- 64.- Averigüe los permisos que tienen los ficheros del directorio raíz.
- 65.- Obtenga un listado de todos los procesos que le pertenecen.
- 66.- Use la orden `man` para ver el formato de la orden `ps`. Pruebe y anote cuáles son las principales opciones.
- 67.- Obtenga un listado de todos los procesos que se están ejecutando en el sistema, paginando la salida.
- 68.- Averigüe los procesos que se están ejecutando en el sistema que pertenezcan al usuario `root`.
- 69.- Ejecute en background un proceso que almacene en un fichero el nombre de aquellos ficheros cuyo tamaño es mayor de 200 bloques a partir del fichero raíz. Transcurridos unos segundos, si no ha terminado, elimine el proceso. ¿Qué pasa con el fichero de salida?
- 70.- Lance un proceso de larga duración (por ejemplo, una orden `find` sobre todo el sistema). Detenga su ejecución con CTRL-Z y observe que el proceso sigue estando en el sistema. Elimínelo.
- 71.- Busque en el manual cómo funciona la familia de órdenes `jobs`. Pruebe, con el ejemplo anterior, cómo se puede reanudar un proceso que previamente ha sido detenido.
- 72.- Haga algunos ejemplos de procesos para ver cómo se puede cambiar la ejecución entre primer y segundo plano y viceversa.
- 73.- Supongamos que se tiene una orden ejecutando en segundo plano (p.e., un `find`) y queremos detenerla temporalmente (no abortarla). ¿Cómo lo haría?
- 74.- Averigüe a qué grupo pertenece.
- 75.- Compruebe cuál es el valor de la máscara de usuario.
- 76.- Cree un fichero nuevo y compruebe los permisos que adquiere.
- 77.- Modifique la máscara de usuario para que los miembros de su grupo puedan leer sus nuevos ficheros y los de otros grupos no. Cree un nuevo fichero y directorio y compare sus permisos con los del problema anterior.
- 78.- Modifique los permisos del fichero y directorio del apartado anterior para que únicamente pueda ser leído y modificado por el propietario. Usar la forma octal del modo.
- 79.- Modifique los permisos del directorio de conexión para que los demás miembros del grupo tengan permiso de lectura y ejecución sobre el mismo. Usar la forma simbólica del modo.
- 80.- Cree un nuevo fichero y protéjalo al máximo de forma que ni siquiera pueda leerlo. Restablezca sus permisos para que pueda borrarlo.
- 81.- Queremos que los ficheros de nueva creación adquieran los permisos `rw-r-----`, ¿qué orden usaremos?
- 82.- Utilice la orden `tar` para hacer un fichero `tar` del contenido de su directorio `HOME`. Guárdelo en un fichero que se llame `copia.tar`.
- 83.- Observe el contenido del fichero `copia.tar`.
- 84.- Comprima el fichero `copia.tar` con la orden `compress`.
- 85.- Descomprima el fichero `copia.tar.Z` y expanda su contenido a partir del subdirectorio `ps`.
- 86.- Obtenga la descripción y sintaxis de la orden `talk`. Póngase de acuerdo con otro usuario y utilicen dicha orden.