

# Ejercicios II – Tema 5

## Ejercicio 5 – Cuestiones breves del tema II

Contesta de forma concisa a las siguientes cuestiones. Para responder algunas de ellas deberás investigar en el nuevo sistema operativo que has instalado.

- a) ¿Es necesario estar situado en el directorio al que se le desea cambiar el nombre?
- b) ¿A qué tipos de personas o usuarios se asignan los privilegios en Linux?
- c) ¿Puede cualquier usuario del sistema cambiar los privilegios de cualquier archivo o carpeta?
- d) ¿A qué equivalen los permisos del número 333 en octal?
- e) ¿Qué significa la siguiente orden: `paco@ubuntupaco:~$ chmod uo=rwx directorio`?
- f) ¿Es obligatorio que un archivo tenga nombre y extensión?
- g) ¿En qué se mide el tamaño de los archivos?
- h) ¿Qué son los archivos de enlace en Linux si los comparamos con Windows?
- i) ¿Todo dispositivo físico en Linux está asociado a un archivo?
- j) ¿Qué tipos de archivos se pueden imprimir en entorno comando con la orden `lpr`?
- k) ¿Cuántos archivos pueden copiarse con una sola orden `cp`?
- l) ¿Para qué sirve la orden `mv` aplicada a archivos?
- m) Los archivos que se eliminan en entorno comando, ¿se almacenan en la papelera de reciclaje?
- n) ¿A qué número en octal equivale el conjunto de permisos `-rw-r-xr-x`?
- o) ¿A qué conjunto de permisos equivale el número 456 en octal?
- p) ¿Quién asigna, modifica o elimina privilegios a los archivos en el sistema?
- q) ¿Cómo se pueden ver los privilegios o permisos asignados a un archivo en Linux?
- r) ¿Se puede comprimir un archivo que ya está comprimido?
- s) ¿Se pueden añadir nuevos archivos a un archivo que contiene otros archivos comprimidos?
- t) ¿Cuál es la diferencia de comprimir archivos con `rar`, `zip` o `tar`?
- u) ¿Podemos descomprimir con `rar` archivos comprimidos con `tar`?
- v) ¿Cuál es el editor más sencillo utilizado para modificar archivos de texto plano en Linux?
- w) ¿Se puede editar un archivo de texto plano con LibreOffice?
- x) Indica que contiene cada uno de los siguientes directorios: `/etc`, `/dev`, `/media` y `/bin`.

## Ejercicio 6 – Comandos en GNU/Linux

Ejecuta un terminal y realiza las siguientes operaciones:

- a) Indica el día y la hora del sistema.
- b) Muestra el calendario de diciembre de 2020.

- c) Muestra el usuario identificado al sistema, fecha y hora con una sola orden.
- d) Muestra el nombre del equipo y el kernel del sistema operativo.
- e) Muestra la identificación del usuario con el que te has validado al sistema y del usuario root.

## Ejercicio 7 – Metacaracteres en GNU/Linux

Completa la tabla que aparece a continuación.

Orden con la expresión	Referencia los archivos y/o directorios que
<b>[a-z]prc.?xt</b>	
	Empiece por cualquier letra en mayúscula, los tres siguiente caracteres sean asa y la extensión sea .txt. Por ejemplo: Casa.txt, Tasa.txt o Rasa.txt.
<b>[A-Z0-9]prc.???</b>	
	Su nombre empieza por un número comprendido entre 1 y 5, seguido de dos caracteres cualesquiera, una p y el resto, que sea cualquier cosa.
<b>[ab]??p*.*</b>	
	Empiecen por G mayúscula o d minúscula y el resto del nombre sea oogle.
<b>[i0-9]*.* o [^0-9]*.*</b>	

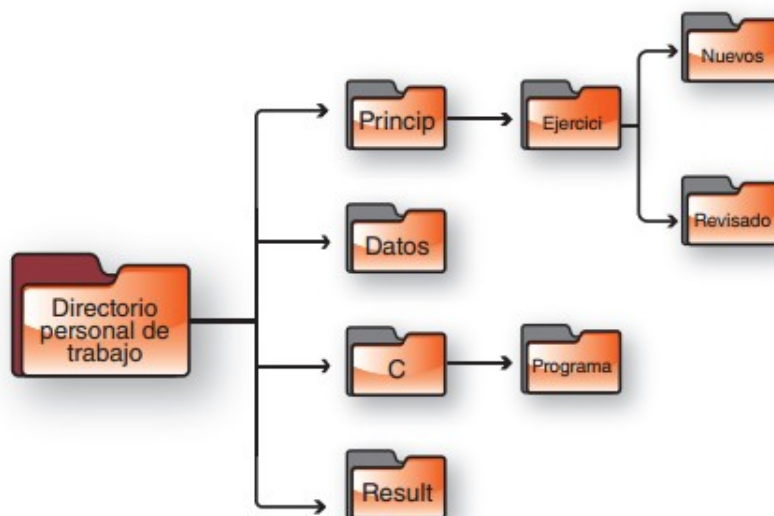
## Ejercicio 8 – Comandos en GNU/Linux II

Rellena esta tabla de órdenes incluyendo un ejemplo en cada una de ellas.

Orden	Ejemplo	Función
<b>ls</b>		
		Sirve para moverse por la estructura de directorios.
	<b>mkdir</b> textos proceso	
		Sirve para eliminar directorios individualmente.
<b>rm -r</b>		
	<b>mv</b> textos datos	
<b>cat</b>		
	<b>rename</b> doc1.txt doc2.txt	
		Copia uno o más archivos desde la ubicación inicial a otro lugar.
	<b>rm</b> s.doc	
		Se utiliza para modificar los atributos a archivos y/o directorios.

## Ejercicio 9 – Archivos y directorios en GNU/Linux

Comprueba que la estructura de directorios que tienes es la de la figura (la de los apuntes del tema) y realiza las siguientes operaciones en entorno comando.



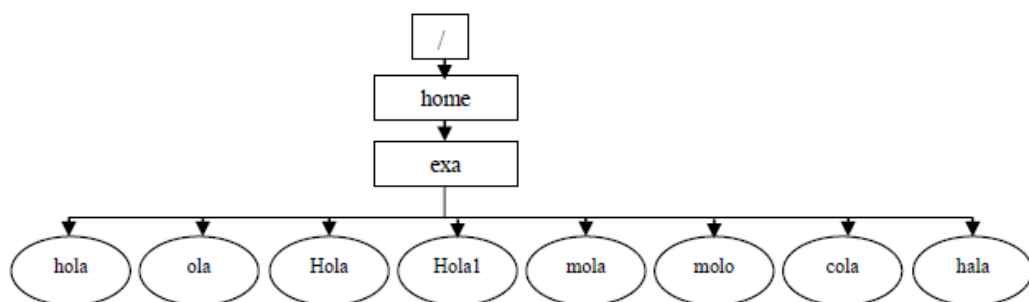
- Crear, en el directorio raíz, el fichero **TEXTO.TXT** cuyo contenido sea tu nombre y la fecha y hora en la que lo estas creando. Visualizar el archivo **TEXTO.TXT** para comprobar su contenido.
- En el directorio **PROGRAMA** crear tres archivos: **TEXTO.TXT**, **TEXTO.BAK** y **TEXTO1.BAS**. El contenido de cada uno de ellos será el de la sintaxis de las órdenes **CD**, **MKDIR** y **RMDIR**.
- Copiar desde **PROGRAMA** al directorio **RESULT**, situados en el directorio **DATOS**, los archivos cuyo nombres empiezan por **TEX** y cuyos dos primeros caracteres de la extensión sea **BA**.
- Renombrar en **RESULT** el fichero **TEXTO1.BAS** por **TEXTO1.BAK**.
- Copiar todos los archivos del directorio **RESULT** en el directorio raíz con el mismo nombre.
- Cambiar el nombre del directorio **DATOS** por el de **DATOS.DAT**.
- En el directorio raíz, renombrar el archivo **TEXTO.TXT** en **PROGRAMA** por **TEXTO.BAS**.
- Visualizar por pantalla el contenido del archivo **TEXTO.DAT** del directorio **RESULT**.
- Mover el archivo **TEXTO.DAT** al directorio **DATOS.DAT** como **TEXTO1.DAT**.
- Borrar todos los archivos del directorio **PROGRAMA** cuya primera letra en la extensión sea B.
- Crear, en **RESULT**, tres archivos, **FICH1**, **FICH2** y **FICH3**.
- Dar permisos de lectura, ejecución y escritura a los tres archivos anteriores usando el **formato no numérico**.
- Poner permisos de escritura a todos los ficheros **RESULT** usando el **formato numérico**.
- Visualizar los permisos de los archivos del directorio **RESULT**. Borrar, desde el directorio **PROGRAMA**, todo el directorio **RESULT**.

## Ejercicio 10 – Permisos en GNU/Linux

1. Si tengo un archivo llamado **prueba1** del que soy **propietario**, explica en cada caso cómo quedarían los **permisos**, es decir, la máscara de este archivo, según vaya ejecutando las siguientes **sentencias** de forma **consecutiva**, es decir, el resultado de la anterior sirve para la siguiente.
  - ▶ `chmod 321 prueba1`
  - ▶ `chmod 256 prueba1`
  - ▶ `chmod 172 prueba1`
  - ▶ `chmod 610 prueba1`
  - ▶ `chmod 006 prueba1`
  - ▶ `chmod 631 prueba1`
2. Si tengo un archivo llamado **prueba1** cuya **máscara es -rwxr-xr-x** del que soy **propietario**, explica en cada caso cómo quedarían los **permisos** de este archivo según vaya ejecutando las siguientes **sentencias** de forma **consecutiva**, el resultado de la anterior sirve para la siguiente.
  - ▶ `chmod g-x prueba1`
  - ▶ `chmod a=x prueba1`
  - ▶ `chmod o-x prueba1`
  - ▶ `chmod a+w prueba1`
  - ▶ `chmod u+r prueba1`
  - ▶ `chmod a= prueba1`

## Ejercicio 11 – Metacaracteres en GNU/Linux II

Crea dentro de **/home/exa** los siguientes archivos.



Utilizando metacaracteres, indica cómo verías los siguientes archivos:

- a) Hola1, Hola
- b) cola, mola, molo
- c) hola, ola, Hola1, Hola, mola, cola, hala
- d) hola, Hola, Hola1
- e) hola, Hola, mola, molo, cola, hala

## Ejercicio 12 – El comando umask

Teniendo en cuenta que utilizaremos 777 para directorios y 666 para ficheros, ¿qué máscara se aplicará a los directorios y a los ficheros si al ejecutar umask nos devuelve lo que se indica? Escribe la máscara (10 caracteres alfabéticos) de directorios y de ficheros para cada apartado.

Permisos	Directorios	Ficheros
555		
321		
744		
021		
131		
458		
123		

## Ejercicio 13 – Permisos en GNU/Linux II

- Si tengo un archivo llamado **prueba1** del que soy **propietario**, explica en cada caso cómo quedarían los **permisos**, es decir, la máscara de este archivo, según vaya ejecutando las siguientes **sentencias** de forma **consecutiva**, es decir, el resultado de la anterior sirve para la siguiente.
  - ▶ `chmod 777 prueba1`
  - ▶ `chmod 234 prueba1`
  - ▶ `chmod 010 prueba1`
  - ▶ `chmod 112 prueba1`
  - ▶ `chmod 762 prueba1`
  - ▶ `chmod 210 prueba1`
- Si tengo un archivo llamado **prueba1** cuya **máscara es -rwxr-xr-x** del que soy **propietario**, explica en cada caso cómo quedarían los **permisos** de este archivo según vaya ejecutando las siguientes **sentencias** de forma **consecutiva**, el resultado de la anterior sirve para la siguiente.
  - ▶ `chmod a-rwx prueba1`
  - ▶ `chmod g=x prueba1`
  - ▶ `chmod o+x prueba1`
  - ▶ `chmod a=w prueba1`
  - ▶ `chmod u+w,o+w prueba1`
  - ▶ `chmod = prueba1`