

Ejercicios I – Tema 5

Ejercicio 1 – Cuestiones breves del tema

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

a) ¿A qué equivale el shell de Linux en sistemas operativos Windows?

Equivale a la consola de comandos de Powershell

b) ¿Se pueden utilizar diferentes shell en un mismo sistema operativo Linux?

Dependiendo de la distribución puede cambiar el shell, al más común es el Bash

c) ¿De qué formas se puede cerrar el sistema en Linux?

- *Mediante el icono de aplicaciones de la barra de tareas -> botón apagado -> Apagar*
- *Usando la combinación de teclas Alt + F4 -> Apagar*
- *En la terminal con el comando `shutdown`*

d) ¿En qué carpeta se pueden encontrar algunos de los iconos de los que dispone Linux?

Sin relevancia

e) ¿Son modificables los iconos representativos de las carpetas en Linux?

Si

f) ¿De qué forma se introducen los parámetros o modificadores en las órdenes de redes de Linux?

Primero se introduce la orden seguida de las opciones y por último los argumentos

g) ¿Qué indica el símbolo del sistema: `paco@ubuntupaco:~#`?

Nos indica el nombre del usuario, el nombre del equipo, en la ruta en la que nos encontramos y el tipo de usuario que somos.

h) ¿Cómo se representa el directorio raíz en Linux? Compáralo con Windows.

Se representa con el carácter `/`. En Windows se representa con `c:\>`

i) ¿Qué es un directorio en Linux?

Es un tipo de archivo especial que puede contener otros directorios o archivos.

j) ¿Cómo podemos saber en qué directorio nos encontramos en modo comando en Linux?

Con el comando `pwd`

k) ¿En qué parte de la estructura de directorios se almacena la información real de los archivos?

En la carpeta `/home`

l) ¿En qué parte se almacena la estructura jerárquica de directorios?

En el directorio raíz

m) ¿Se puede tener una partición de intercambio en un formato de archivos ext4?

n) ¿Se pueden definir particiones en Linux con formato FAT32?

Si, Linux lo permite

o) ¿Cuál es el directorio equivalente a Usuarios de Windows en Linux?

`/home`

p) ¿En qué directorio se encuentran los programas de administración del sistema?

[/sbin](#)

q) Los directorios `.` y `..` ¿existen siempre en cualquier directorio Linux?

[En todos excepto en el directorio raíz que solo existe el "."](#)

r) ¿De qué forma podemos ver la ayuda del comando `ls` de Linux?

- [man ls](#)
- [ls --help](#)

s) ¿Con qué orden se puede visualizar el directorio en el que nos encontramos actualmente?

[Con el comando `pwd`](#)

t) ¿Se pueden utilizar caracteres comodín o metacaracteres con la orden `mkdir`?

[Se puede pero no es recomendable.](#)

u) ¿Se pueden eliminar directorios que contienen otros subdirectorios con la orden `rmdir`?

[De normal no, pero añadiendo el parámetro `-R` se puede.](#)

v) ¿Se pueden eliminar directorios con una sola orden `rmdir` que estén situados en diferentes rutas?

[Sí, indicando las rutas de los directorios a borrar separadas con un espacio.](#)

w) La orden `mv`, ¿sirve para cambiar la ubicación de un directorio o para cambiarle el nombre?

[Sirve para las dos cosas](#)

Ejercicio 2 – Gestión de directorios con GNU/Linux

Teniendo en cuenta que te has validado en GNU/Linux como alumno y te encuentras en su carpeta realiza los siguientes ejercicios. Ahora mismo estás situado en `/home/alumno`.

1. Crea los directorios A, B, cosas, buenas, malas, música y clásica con una única sentencia y trayectoria absoluta.

[mkdir A A/cosas A/cosas/buenas A/cosas/malas B B/música B/música/clásica](#)

2. Sitúate en malas. Desde ahí crea C, video, HD, infantil, series y película con una única sentencia y trayectoria relativa.

[cd A/cosas/malas](#)

[mkdir ../../C ../../C/video ../../C/video/HD ../../C/video/HD/infantil](#)

[../../C/video/HD/infantil/series ../../C/video/HD/infantil/película](#)

3. Sigues situado en malas. Crea un fichero llamado árbol en buenas que muestre el árbol de directorios dependiente de alumno utilizando trayectorias relativas.

[ls -R ../../>../regulares/arbolista](#)

4. Sigues situado en malas. Mueve el directorio video dentro de clásica.

[mv ../../C/video ../../B/música/clásica](#)

5. Sigues situado en malas. Crea una copia de series en buenas.

[cp -R ../../B/música/clásica/video/HD/infantil/series ../buenas](#)

6. Sigues situado en malas. Elimina el directorio HD.

[rm -r ../../B/música/clásica/video/HD](#)

7. Sigues situado en malas. Cambia el nombre de buenas por el de regulares.

[mv ../buenas ../regulares](#)

8. Sigues situado en malas. Muestra el fichero árbol.

```
cat ../regulares/arbolista
```

9. Sitúate en música. Elimina el directorio cosas (no puedes usar la orden `rm -r`).

```
cd ../../B/música
rm ../../A/cosas/regulares/arbolista
rmdir ../../A/cosas/regulares/series
rmdir ../../A/cosas/regulares ../../A/cosas/malas
rmdir ../../A/cosas
```

Ejercicio 4 – Gestión de directorios con GNU/Linux II

1. Teniendo en cuenta el árbol de arriba. Crea con una única sentencia los directorios A, B, C, D, E, A1, A2, A21.

```
mkdir A A/A1 A/A2 A/A2/A21 B C D E
```

2. Situado en usuario, mueve A21 a E.

```
mv A/A2/A21 E/
```

3. Situado en usuario, cambia el nombre de A21 por A21CAMBIADO.

```
mv E/A21 E/A21cambiado
```

4. Situado en usuario, borra A (prohibido usar `rm -r`)

```
rmdir A/A1
rmdir A/A2
rmdir A
```

5. Situado en usuario, crea un fichero llamado árbol en E que contenga el árbol de directorios que depende de usuario.

```
ls -R > E/arbol.txt
```

6. Situado en usuario, ordena el fichero árbol y almacenarlo en B con el nombre ordenado.

```
sort arbol.txt > ../B/ordenado.txt
```

7. Situado en usuario, crea una copia de B en D.

```
cp -r ../B ../D/
```

8. Situado en usuario, crea en el directorio activo un fichero llamado calendario que contenga el calendario del mes de diciembre de 2018.

```
cal 12 2018 > calendario.txt
```

9. Situado en usuario, cambia el nombre del fichero calendario por el de calendario2.

```
mv calendario.txt calendario2.txt
```

10. Situado en usuario, elimina el fichero calendario2.

```
rm calendario2.txt
```

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?
No, se puede realizar el cambio mediante el uso de rutas absolutas o relativas y el comando mv
- b) ¿A qué tipos de personas o usuarios se asignan los privilegios en Linux?
Al propietario, a los grupos y a otros usuarios.
- c) ¿Puede cualquier usuario del sistema cambiar los privilegios de cualquier archivo o carpeta?
No, solo el administrador del sistema.
- d) ¿A qué equivalen los permisos del número 333 en octal?
- -WX -WX -WX
- e) ¿Qué significa la siguiente orden: paco@ubuntupaco:~\$ chmod uo=rwx directorio?
Asigna privilegios de lectura, escritura y ejecución al propietario y al resto de usuarios sobre el archivo que contengan los caracteres directorio más un carácter cualquiera adicional.
- f) ¿Es obligatorio que un archivo tenga nombre y extensión?
En Linux la extensión no es obligatoria.
- g) ¿En qué se mide el tamaño de los archivos?
Mide el tamaño en bytes
- h) ¿Qué son los archivos de enlace en Linux si los comparamos con Windows?
Son los accesos directos.
- i) ¿Todo dispositivo físico en Linux está asociado a un archivo?
Si, todo dispositivo físico lo está.
- j) ¿Qué tipos de archivos se pueden imprimir en entorno comando con la orden lpr?
Se pueden imprimir cualquier archivo.
- k) ¿Cuántos archivos pueden copiarse con una sola orden cp?
Un montón
- l) ¿Para qué sirve la orden mv aplicada a archivos?
Puede usarse para renombrar y mover archivos.
- m) Los archivos que se eliminan en entorno comando, ¿se almacenan en la papelera de reciclaje?
No, se eliminan directamente.
- n) ¿A qué número en octal equivale el conjunto de permisos -rw-r-xr-x?
Pertenece al número en octal 655
- o) ¿A qué conjunto de permisos equivale el número 456 en octal?
- r-- r-x rw-
- p) ¿Quién asigna, modifica o elimina privilegios a los archivos en el sistema?

El administrador

q) ¿Cómo se pueden ver los privilegios o permisos asignados a un archivo en Linux?

`ls -l nombreadchivo.algo`

r) ¿Se puede comprimir un archivo que ya está comprimido?

No pone nada en el tema

s) ¿Se pueden añadir nuevos archivos a un archivo que contiene otros archivos comprimidos?

No pone nada en el tema

t) ¿Cuál es la diferencia de comprimir archivos con rar, zip o tar?

tar empaqueta archivos, zip y rar los comprimen.

u) ¿Podemos descomprimir con rar archivos comprimidos con tar?

Sí, puede descomprimir múltiples formatos y entre ellos `tar`

v) ¿Cuál es el editor más sencillo utilizado para modificar archivos de texto plano en Linux?

Nano es el editor de textos más sencillo.

w) ¿Se puede editar un archivo de texto plano con LibreOffice?

Sí, pero en caso de modificarlo solo debería contener caracteres.

x) Indica que contiene cada uno de los siguientes directorios: `/etc`, `/dev`, `/media` y `/bin`.

`/etc` : Ficheros de configuración y utilidades de administración del sistema.

`/dev` : Archivos especiales de bloques y caracteres asociados a dispositivos de hardware.

`/media` : Contiene todas las unidades físicas montadas en nuestro sistema.

`/bin` : Tiene ficheros de comandos ejecutables utilizables por todos los usuarios.

Ejercicio 6 – Comandos en GNU/Linux

Ejecuta un terminal y realiza las siguientes operaciones:

a) Indica el día y la hora del sistema.

`date`

b) Muestra el calendario de diciembre de 2020.

`cal 12 2020`

c) Muestra el usuario identificado al sistema, fecha y hora con una sola orden.

`whoami`

d) Muestra el nombre del equipo y el kernel del sistema operativo.

`uname -a`

e) Muestra la identificación del usuario con el que te has validado al sistema y del usuario root.

`id`

Ejercicio 7 – Metacaracteres en GNU/Linux

Completa la tabla que aparece a continuación.

Orden de la expresión	Referencia los archivos y/o directorios que
[a-z]prc.?xt	Que el primer carácter sea minúscula de la a a la z seguido de prc y que en la extensión el primer carácter sea cualquiera seguido de xt
[A-Z]asa.txt	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.
[A-Z0-9]prc.???	Que empiece por cualquier letra mayúscula o cualquier número seguido de prc y con cualquier tipo de extensión con tres caracteres.
[1-5]??p*.*	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.
[ab]??p*.*	Su nombre empieza por las letras a ó b , seguido de dos caracteres cualesquiera, una p y el resto, que sea cualquier cosa.
[Gd]oogle.*	Empiecen por G mayúscula o d minúscula y el resto del nombre sea oogle.
[!0-9]*.* o [^0-9]*.*	Cualquier archivo que no empiece por un número del 0 al 9

Ejercicio 8 – Comandos en GNU/Linux II

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

Orden	Ejemplo	Función
ls	ls A/cont	Hace un listado de los archivos o carpetas de la carpeta cont
cd	cd ../ficheros	Sirve para moverse por la estructura de directorios.
mkdir	mkdir textos proceso	Crea las carpetas textos y proceso
rm	rm textos	Sirve para eliminar directorios individualmente.
rm -r	rm -r proceso	Elimina archivos y directorios de manera recursiva.
mv	mv textos datos	Cambia el nombre de la carpeta textos a datos
cat	cat doc1.txt	Muestra el contenido de un archivo
mv	rename doc1.txt doc2.txt	Cambia el nombre de un archivo
cp	cp arch1.txt B/archivos/	Copia uno o más archivos desde la ubicación inicial a otro lugar.
rm	rm s.doc	Elimina un archivo
chmod	chmod 531 fich.txt	Se utiliza para modificar los atributos a archivos y/o directorios.

Ejercicio 9 – Archivos y directorios en GNU/Linux

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
- -WX -W- --X
► chmod 256 prueba1
- -W- r-X rW-
► chmod 172 prueba1
- --W rWX -W-
► chmod 610 prueba1
- rW- --X ---
► chmod 006 prueba1
- --- --- rW-
► chmod 631 prueba1
- rW- -WX --X
```

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
- rWX r-- r-X
► chmod a=x prueba1
- --X --X --X
► chmod o-x prueba1
- --X --X ---
► chmod a+w prueba1
- -WX -WX -W-
► chmod u+r prueba1
- rWX -WX -W-
► chmod a= prueba1
- --- --- ---
```

Ejercicio 11 – Metacaracteres en GNU/Linux II

Utilizando metacaracteres, indica cómo verías los siguientes archivos:

a) Hola1, Hola

```
ls Ho???
```

b) cola, mola, molo

```
ls [cm]ol?
```

c) hola, ola, Hola1, Hola, mola, cola, hala

```
ls [hoHmca]?l*
```

d) hola, Hola, Hola1

```
ls [hH]ola*
```

e) hola, Hola, mola, molo, cola, hala

```
ls [hHmc]?l?
```

Ejercicio 12 – El comando umask

	Directorio	ficheros
555	222 d -w- -w- -w-	222
312	456	
744		
021		
131		
458		
123		

Ejercicio 13 – Permisos en GNU/Linux II

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 777 prueba1`

- `rwX rwX rwX`

► `chmod 234 prueba1`

- `-W- -WX r--`

► `chmod 010 prueba1`

- `--- --X ---`

► `chmod 112 prueba1`

- `--X --X -W-`

► `chmod 762 prueba1`

- `rwX rw- -W-`

► `chmod 210 prueba1`

- `-W- --X ---`

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 a-rwx prueba1`

- `--- --- ---`

► `chmod g=x prueba1`

- `--- --X ---`

► `chmod o+x prueba1`

- `--- --X --X`

► `chmod a=w prueba1`

- `-W- -WX -WX`

► `chmod u+w,o+w prueba1`

- `-W- -WX -WX`

► `chmod = prueba1`

- `--- --- ---`

Ejercicios III – Tema 5

1. Al crear un directorio, quieres tener todos los permisos activados, que los usuarios de tu grupo solo lo puedan leer y los que no pertenezcan a tu grupo no puedan hacer nada, ¿qué valor debería tener umask?

Si no quieres tocar umask y ya tienes creado el directorio, cámbiaselos de manera que cumplan los permisos que se especifican. Hazlo en modo numérico y con letras.

```
Umask 037  
rwx r-- --- ; 111 100 000 ; 740  
chmod 740 directorio
```

2. Crea dos usuarios nuevo y nuevo2.

```
sudo adduser nuevo
```

```
adduser --ingroup grupo2 nuevo2
```

3. Haz que aparezca por pantalla el nombre (solo el nombre) de los usuarios que tengan una ! en la contraseña en el fichero shadow.

```
sudo grep "^!" /etc/shadow
```

4. Cambia el shell de nuevo2 por el de /bin/false. Intenta entrar en el sistema como nuevo2. ¿A qué es debido que no puedas entrar? Cámbiasela de nuevo a /bin/bash.

```
Sudo usermod -s /bin/false nuevo2
```

Ejercicios IV – Tema 5

Ejercicio 15 – Repaso de comandos

1. Utiliza el comando que te lleva a tu directorio personal directamente. Después, ejecuta el comando que te dice en qué subdirectorio estás y, a continuación, el comando que te dice quién eres.

```
cd ~  
pwd  
whoami
```

2. Crea los siguientes directorios, desde el subdirectorio personal de usuario y sin salir de él.

```
mkdir ejercicios/ingresos/alquiler ejercicios/ingresos/ventas ejercicios/gastos
```

3. Ve al directorio gastos. Utilizando el comando cat, crea dos ficheros: emple1.txt y emple2.txt, que contengan tres palabras cada uno, una debajo de otra: roble, tigre, casa.

```
cat > emple1.txt  
roble  
tigre  
casa  
(Ctrl + D)  
cat roble tigre casa > emple2.txt
```

4. Comprueba si el fichero emple1.txt está ordenado. Si no lo está, ordénalo y guarda el resultado en empleord.txt.

```
sort emple1.txt > empleord.txt
```

5. Utilizando caracteres comodines, copia los tres ficheros en ventas, sin salir de gastos y utilizando una trayectoria relativa.

```
cp emp*. * ../ingresos/ventas
```

6. Utilizando caracteres comodines, mueve los ficheros emple1.txt y emple2.txt a alquiler, sin salir de gastos y utilizando una trayectoria absoluta.

```
mv /home/ciclost/ejercicios/gastos/emple?.* /home/ciclost/ejercicios/ingresos/alquiler
```

7. Ve al directorio raíz. Muestra el listado de todos los directorios en formato largo y después cambia la salida estándar a un fichero llamado directorios que esté en tu carpeta personal.

```
cd /  
ls -lR > /home/ciclost/directorios.txt
```

8. Vuelve a gastos utilizando la opción del comando que te lleva al directorio anterior.

```
cd -
```

Ve a alquiler utilizando una trayectoria relativa.

```
Cd ../ingresos/alquiler
```

Muestra el número de líneas del fichero emple1.txt.

```
cat -n emple1.txt
```

Utilizando el comando cat, une los ficheros emple1.txt y emple2.txt en emple3.txt.

```
cat emple1.txt emple2.txt > emple3.txt
```

Finalmente, muestra el número de palabras del fichero emple3.txt.

```
wc -w emple3.txt
```

9. Crea un enlace a emple2.txt llamado enlace2.txt.

(En la carpeta adecuada) `ln emple2.txt enlace2.txt`

Modifica enlace2.txt. Comprueba que se ha modificado emple2.txt.

`nano enlace2.txt`

`cat ./ingresos/alquiler/emple2.txt` (El archivo se ha modificado)

Crea un enlace simbólico a emple3.txt llamado enlsim3.txt.

?????

Muestra el listado largo de todos las entradas del directorio y mira el primer carácter de cada línea.

?????

10. Borra los directorios y ficheros creados anteriormente.

`rm -r ejercicios`

Ejercicios VI – Tema 5

Ejercicio 20 – Utilidad GREP

El fichero `/etc/passwd` es un fichero de texto que contiene información acerca de las cuentas de usuario definidas en el sistema. Cada línea contiene siete datos, separados por el carácter ":". Estos datos son: nombre de la cuenta, contraseña encriptada (opcional), el código de usuario (user ID), el código del grupo (group ID), un comentario, el nombre del directorio 'home' y el nombre del intérprete de comandos que se ejecutará cuando el usuario se conecte.

Se utilizará este fichero en las cuestiones siguientes.

1. Muestra la línea correspondiente a la cuenta de usuario.

`grep ciclost`

2. Muestra las cuentas que empiecen por a.

`grep "^a" passwd`

3. Muestra las cuentas que empiecen por a o r.

`grep "^[ar]" passwd`

4. Muestra las cuentas que terminen por u.

`grep "[^:]*u:" passwd`

5. Muestra las cuentas que usen bash como intérprete de comandos.

`grep "bash$" passwd`

6. Muestra las cuentas que NO usen bash como intérprete de comandos.

`grep -v "bash$" passwd`

7. Muestra las cuentas que no empiecen por vocal.

`grep -v "[a,e,i,o,u,A,E,Y,O,U]" passwd`

8. Muestra las cuentas que empiecen por mayúscula.

`grep "[A-Z]" /etc/passwd`

12. Muestra las cuentas root o de ftp.

```
grep -e "^(ftp|root):" passwd
```

En el directorio /usr/include se almacenan todos los ficheros de cabecera de los lenguajes C/C++.

13. ¿Qué ficheros de cabecera usan la constante MAXDOUBLE?

```
-r
```

```
-ri
```

14. ¿En qué ficheros aparece la palabra bash? (En mayúscula o minúscula).

```
-r
```

```
-ri
```

Ejercicio 21 – Utilidad SORT

Fichero etc/passwd

1. Ordena el fichero /etc/passwd por orden alfabético.

```
sort /etc/passwd
```

2. Ordénalo en sentido inverso.

```
sort -r /etc/passwd
```

3. Ordénalo ignorando mayúsculas.

```
sort -f passwd
```

4. Ordénalo según el tercer campo.

```
sort -t ":" -k3 /etc/passwd
```

5. Ordénalo según el tercer campo en orden numérico.

```
sort -t ":" -k4 -n /etc/passwd
```

Ejercicio 22 – Utilidad CUT

1. Muestra la columna 4 del fichero /etc/passwd.

```
cut -d ":" -f 4 /etc/passwd
```

2. Muestra la columna 1 y la 7.

```
cut -d ":" -f 1-7 /etc/passwd
```

3. Muestra las columnas 1, 3 y de la 5 a la 7.

```
cut -d ":" -f 1,3,5-7 /etc/passwd
```

Ejercicio 24 – Utilidad UNIQU

1. Obtener, a partir de /etc/passwd, una lista de los intérpretes de comandos utilizados en el sistema, ordenados de más usados a menos usados.

```
cut -d ":" -f7 /etc/passwd | sort | uniq -c | sort -rn
```

Ejercicio 25 – Utilidad TR

1. Muestra el fichero /etc/passwd con las letras a minúsculas escritas como mayúsculas.

```
tr [a-z] [A-Z] /etc/passwd
```

2. Cambia las vocales minúsculas por mayúsculas.

```
tr [a,e,i,o,u] [A,E,I,O,U] < ./etc/passwd.txt
```

3. Cambia las minúsculas por mayúsculas.

```
tr [a-z] [A-Z] /etc/passwd
```

4. Cambia cada letra por la que le sigue en orden alfabético, excepto la letra z que será sustituida por la letra a.

```
tr [a-z] [b-za] < ./etc/passwd.txt
```

Ejercicio 26 – Utilidad WC

1. ¿Cuántas líneas hay en el fichero /usr/include/stdlib.h?

```
wc -l /usr/include/stdlib.h
```

2. ¿Cuántos caracteres hay en el fichero /usr/include/stdlib.h?

```
wc -m /usr/include/stdlib.h
```

3. ¿Cuál es la longitud de la línea más larga del fichero /usr/include/stdlib.h?

```
wc -L /usr/include/stdlib.h
```

Ejercicio 29 – Utilidad DIFF

1. ¿Qué diferencias hay entre el programa ordenar1.c y ordenar2.c?

2. ¿Qué diferencias hay entre los directorios Dir1 y Dir2?

3. ¿Y si queremos ver las diferencias dentro de un contexto de 3 líneas que permanecen iguales en los dos programas del primer ejercicio?

Ejercicio 30 – Utilidad FIND

1. ¿En qué lugar del directorio /usr/include está el fichero iostream.h?

```
find /usr/include -name iostream.h
```

2. ¿Qué ficheros de /usr/include ocupan más de 50 KB?

```
find /usr/include -size +50k
```

3. ¿Y que además no sean cabeceras?

```
???-----
```

4. ¿Qué ficheros (no directorios) hay en /usr/include que no terminen en .h?

```
find /usr/include -type f -name *.h
```

5. Busca los ficheros .h que contengan la variable MAXDOUBLE.

```
find /usr/include -name "*.h" -exec
```

6. Borra todos los ficheros core que hay en el directorio actual.

```
???
```