

## **Boletín de ejercicios de la práctica 1**

En este boletín se recogen una series de ejercicios encaminados a que el alumno practique las órdenes vistas en cada uno de los apartados de la práctica.

Para cada uno de los apartados hay dos conjuntos de ejercicios:

- El primer conjunto de ejercicios está diseñado para practicar los conceptos básicos del tema, y se espera que el alumno los complete durante las horas de clase, pudiendo de esta manera solicitar ayuda al profesor si es preciso.
- El segundo conjunto de ejercicios está pensado para que el alumno practique por su cuenta fuera de las horas de clase (como trabajo no presencial). Pueden ser ejercicios muy similares a los del primer conjunto o, en ocasiones, otros bastante más complejos. De esta manera se intenta incitar al alumno a profundizar en el tema tratado.

Estos ejercicios están diseñados como guía para ayudar al aprendizaje del alumno. No forman parte de ningún entregable ni su realización será tenida en cuenta para la evaluación de la asignatura. La forma en que se evaluará esta parte de la materia será comentada por el profesor en clase.

## 1. Ejercicios del apartado 1: *Primeros pasos en Unix*

### 1.1. Ejercicios para realizar en clase

1. Entra en `petra` o en el sistema Unix/Linux del que dispongas.
2. Cambia tu palabra de paso. Intenta distintas posibilidades para ver el nivel de seguridad que incorpora el sistema a la hora de permitir palabras más o menos sencillas (palabras muy cortas, sólo con letras, palabras relacionadas con el usuario, etc.)
3. Obtén ayuda de las siguientes órdenes: `man`, `passwd`, `apropos`, `help`, `info`, `type`.
4. ¿Qué fichero se ejecuta cuando introducimos la orden `man`?
5. ¿Con qué orden obtengo información sobre las órdenes internas del shell? ¿Cómo se utiliza?
6. ¿Con qué orden obtengo información sobre órdenes externas? ¿Cómo se utiliza?
7. ¿Cómo puedo saber que ordenes he introducido en el `bash` últimamente?
8. ¿Cómo puedo ejecutar una orden que aparece en el historial?
9. Quiero visualizar un fichero que se llama `Nombrefichero muy largo denarices`. Sólo tengo en el directorio actual ese fichero que comience por `N`. Indique cómo puedo introducir el nombre de fichero con solo dos pulsaciones de teclado.
10. Repite la última orden introducida, utilizando las facilidades del shell. Obtén ayuda de la orden `history` (¿es interna o externa? Averígualo).

### 1.2. Ejercicios para realizar como trabajo no presencial

11. Utilizando las facilidades del shell, averigua qué órdenes externas comienzan con la letra `p`. ¿Cuántas por `pa`? ¿Y por `pas`?
12. ¿Cómo puedo repetir la última orden que he introducido, si estoy utilizando el `bash` como shell?
13. ¿Con qué orden puedo saber si una orden es interna o externa del shell? ¿Cómo se utiliza?
14. ¿Para qué sirve la orden `whatis`?
15. ¿Qué opción del `man` hace que funcione de manera similar a la orden `apropos`?
16. Consulta la página del `man` de la orden `history` para estudiar las distintas posibilidades que ofrece y averigua cómo puedes:
  - a. Ejecutar la última orden introducida (sin usar la `↑`).
  - b. Ejecutar la orden que introdujiste hace 6 órdenes.
  - c. Ejecutar la última orden que introdujiste que comenzaba por `hi`.

## 2. Ejercicios del apartado 2: El sistema de ficheros

### 2.1. Ejercicios para realizar en clase

1. Vete al directorio raíz y utiliza la orden `cd` de la manera más sencilla posible para volver a tu directorio de trabajo.
2. Sitúate en el directorio `/etc` y accede a uno de los ficheros situados en el directorio de trabajo propio utilizando el camino, ruta de acceso o pathname de las dos formas posibles: absoluta y relativa.
3. Sitúate en el directorio raíz. Comprueba qué contenido tiene. Comprueba si existen directorios.
4. Crea un fichero que sea accesible a través de dos nombres diferentes. Comprueba que aparece un 2 en la segunda columna, indicando con ello que existen dos nombres para el mismo fichero. Comprueba que efectivamente se trata de un sólo fichero, modificando desde uno de ellos y comprobando desde el otro que ha sido modificado.
5. Crea un fichero de prueba para modificarle los permisos, por ejemplo que contenga el listado de los ficheros del directorio (para realizar eso únicamente debes añadir `> nombreFichero`, sin las comillas, a cualquier orden; esto creará un fichero que contendrá la salida de la orden).
6. Comprueba los permisos actuales y haz que sólo el propietario pueda leerlo.
7. Crear un directorio de prueba. Copia algunos ficheros a su interior. Modifica sus permisos de éste para que sólo el propietario pueda leerlo. Intenta situarte en él. ¿Puedes? ¿Por qué?. Intenta ver su contenido. ¿Puedes? ¿Por qué?
8. Cambia los permisos del directorio anterior para que sólo el propietario pueda acceder a él. Intenta situarte en él. ¿Puedes? ¿Por qué?. Intenta ver su contenido. ¿Puedes? ¿Por qué?
9. Indica la expresión regular que hay que introducir para referirnos a todos los ficheros cuyo nombre contenga tres caracteres, siendo el primero necesariamente una letra (mayúscula o minúscula).
10. Indica la expresión regular que hay que introducir para referirnos a todos los ficheros cuyo nombre contenga un "." en cualquier posición a excepción de la primera.

### 2.2. Ejercicios para realizar como trabajo no presencial

11. Inspeccionar el contenido de algunos ficheros. Pueden ser interesantes ficheros como el `/etc/passwd` y el `/etc/group`. Averigua, a partir de los ficheros anteriores, el número de usuario, número de grupo y nombre de grupo correspondiente a cada usuario.
12. Ve a tu directorio HOME. Mira cuántos enlaces tiene. Crea un directorio dentro de él. Comprueba cuántos enlaces tiene ahora tu directorio HOME. ¿Por qué ha cambiado así?
13. Busca mediante la ayuda los tipos de ficheros que puedan aparecer como primer campo del `ls`.
14. Indica la expresión regular que hay que introducir para referirnos a todos los ficheros cuyo nombre comience por "a" y terminen por un dígito.

15. Indica la expresión regular que hay que introducir para referirnos a todos los ficheros cuyo nombre no termine por ".c".
16. Indica la expresión regular que hay que introducir para referirnos a todos los ficheros cuyo nombre tenga al menos 2 caracteres.
17. Inspeccionando los ficheros que tengo en mi directorio me he encontrado con que tengo uno que se llama "." y otro "..". Yo no los he creado y no sé lo que significan. ¿Quién los ha creado, y qué utilidad tienen?
18. Ve al directorio /dev y comprueba que contiene ficheros de tipo "dispositivo". Comprobar también que se da información acerca del número mayor y menor justo antes de la fecha de modificación.
19. ¿Para qué sirve la orden df? ¿Cómo se utiliza?
20. ¿Para qué sirve la orden mount? ¿Cómo se utiliza?
21. ¿Para qué sirve la orden quota? ¿Cómo se utiliza?
22. ¿Qué permisos necesito tener para poder crear un fichero? Indica todos aquellos que sean necesarios, tanto del fichero como del directorio al que pertenezca.
23. ¿Qué permisos necesito tener para poder ejecutar un fichero (se supone que contiene código ejecutable)? Indica todos aquellos que sean necesarios, tanto del fichero como del directorio al que pertenezca.
24. ¿Qué permisos necesito tener para poder eliminar un fichero? Indica todos aquellos que sean necesarios, tanto del fichero como del directorio al que pertenezca.
25. ¿Qué permisos necesito tener para poder leer un fichero? Indica todos aquellos que sean necesarios, tanto del fichero como del directorio al que pertenezca.
26. ¿Qué permisos necesito tener para poder modificar un fichero? Indica todos aquellos que sean necesarios, tanto del fichero como del directorio al que pertenezca.
27. Quiero borrar todos los ficheros del directorio actual. Suponiendo que tengo los permisos necesarios, ¿cómo puedo hacerlo?
28. Quiero borrar un directorio "d1", perteneciente al directorio actual, y todo su contenido. Suponiendo que tengo los permisos necesarios, ¿cómo puedo hacerlo mediante una única orden?
29. Quiero cambiar el directorio actual a un "hermano" (hijo del mismo padre) del actual. El directorio al que quiero cambiar se llama "d1". ¿Cómo se hace con una única orden.
30. Quiero conocer la mayor información posible (permisos, tamaño, etc) del directorio actual. ¿Cómo puedo hacerlo mediante una única orden?
31. Quiero conocer la mayor información posible (permisos, tamaño, etc) del directorio padre del actual. ¿Cómo puedo hacerlo mediante una única orden?
32. Quiero copiar todos los ficheros del directorio actual a un directorio que se llama /tmp. Suponiendo que tengo los permisos necesarios, ¿Cómo puedo hacerlo?
33. Quiero crear un directorio hijo del directorio "d1", que es hijo del directorio actual. Indique cómo se hace con una única orden?

34. Quiero mover todos los ficheros del directorio actual a un directorio que se llama /tmp. Suponiendo que tengo los permisos necesarios, ¿ Cómo puedo hacerlo?

### 3. Ejercicios del apartado 3: *Ordenes relativas a procesos y usuarios*

#### 3.1. Ejercicios para realizar en clase

1. Localiza los usuarios que se encuentran conectados en este momento a la máquina.
2. Ejecuta las líneas de órdenes:  

```
sleep 15; echo He terminado  
(sleep 15; echo He terminado)&
```

e intenta en ambos casos ejecutar la orden `ps -f` en el tiempo de espera. Explica la diferencia de funcionamiento.
3. Ejecuta el programa `/home/profesores/albizu/publico/bucle` en primer plano. Haz que termine. Haz lo mismo, pero lanzándolo en segundo plano.
4. Ejecuta la orden `ps` para descubrir los procesos en curso.
5. Busca información acerca de las órdenes `who` y `finger` y comprueba la diferencia.
6. Usa la orden `top` para ver los procesos en ejecución. Intenta explicar el contenido de cada pieza de información que se muestra (acude al man en caso necesario).
7. Define una orden para suspender el proceso en primer plano. Ejecuta la orden `man` y suspende su ejecución. Usa `ps` para ver el estado en que está. Reanuda la ejecución del proceso.

#### 3.2. Ejercicios para realizar como trabajo no presencial

8. Indica cómo podemos conocer todos los procesos que el usuario tiene en ejecución.
9. Indica cómo podemos lanzar procesos en segundo plano.
10. Indica cómo podemos reanudar en segundo plano un proceso que acabamos de suspender.
11. Indica cómo podemos suspender el proceso que se está ejecutando en primer plano.
12. Indica cómo podemos conocer la lista de procesos que hemos lanzado en segundo plano.
13. Indica cómo podemos suspender un proceso que no está en primer plano, pero que se ha lanzado desde otra terminal.
14. Indica cómo se eliminan procesos en ejecución. Explica las distintas alternativas que tenemos.
15. Utiliza la orden `man` para identificar las opciones de que dispone la orden `ps`. Pruébalas para comprobar cómo funcionan.
16. Lanza el programa `bucle` antes comentado en segundo plano dos veces, una normal y otra utilizando la orden `nice`. Déjalas ejecutando unos 10 minutos o más y comprueba cuánto tiempo de procesador han utilizado. Explica la diferencia entre los tiempos consumidos.
17. Utiliza la orden `at` para lanzar el programa `bucle` dentro de unos minutos. Comprueba entonces que se ha lanzado.
18. Utiliza la orden `crontab` para lanzar el programa `bucle` dentro de unos minutos. Comprueba entonces que se ha lanzado.

## 4. Ejercicios del apartado 4: Conductos, filtros y redirecciones de e/s

### 4.1. Ejercicios para realizar en clase

1. Elimina la recepción de mensajes procedentes de otros usuarios.
2. Crea un fichero de texto con la orden `cat`.
3. Visualiza a través de una tubería el contenido del fichero `/etc/passwd`, haciendo que la pantalla se detenga cuando ésta se llene.
4. Añade un texto a un fichero de texto ya existente.
5. Redirecciona el error que produzca una orden incorrecta a un fichero de error.
6. Cuenta el número de usuarios que están conectados en este instante, mediante una orden que utilice tubería.
7. Averigua cuántas veces te has conectado en el último mes. Cuenta también el número total de conexiones que ha habido a `petra` en el último mes.
8. Cuenta cuantos usuarios distintos hay dados de alta en `petra`.
9. Cuenta el número de usuarios distintos que se han conectado al sistema en el último mes.

### 4.2. Ejercicios para realizar como trabajo no presencial

10. Guarda el contenido de un directorio en un fichero.
11. Guarda en un fichero la localización de todos aquellos ficheros del sistema que tengan como extensión `.out`. Realiza la operación en modo subordinado.
12. Guarda en el fichero `CONECTADOS` la máxima cantidad de información que puedas obtener a través de alguna orden, sobre los usuarios conectados en este momento.
13. Escribe en un fichero la información de los usuarios que tengan “Alvarez” como alguno de sus apellidos.
14. Cuenta el número de usuarios que existen en el sistema.
15. Cuenta el número de grupos que existen en el sistema.
16. Escribe un mensaje a un compañero que esté conectado a la máquina de forma que le aparezca a éste de inmediato en pantalla.
17. Entabla un diálogo por ordenador con un compañero que esté conectado a la máquina.
18. Busca mediante una orden el fichero `passwd` en el directorio `/etc`.
19. Busca mediante una orden el fichero `passwd` en toda la jerarquía de directorios.
20. Comprueba a través de una tubería si existe algún usuario conectado en este instante que se apellide “García”.
21. Crea un fichero que contenga el contenido del directorio raíz denominado `RAIZ`.
22. Cuenta cuantos usuarios están conectados actualmente desde una máquina de la Escuela
23. Cuenta cuantos usuarios están dados de alta en el sistema que se apelliden Sanchez.
24. Cuenta el número de máquinas distintas desde las que se han realizado conexiones a `petra` este último mes (desde que se inicializo el fichero `wtmp`).
25. Cuenta el número de usuarios distintos que se han conectado a `petra` este último mes (desde que se inicializo el fichero `wtmp`).
26. Haz una estadística (ordenada de mayor a menor) del número de sesiones que tiene abiertas cada usuario.

27. Haz una estadística (ordenada de mayor a menor) del número de veces que se está ejecutando cada proceso en el sistema.
28. Haz una estadística (ordenada de mayor a menor) del número de veces que se ha conectado a petra cada maquinas desde las que se accedido al sistema este último mes (desde que se inicializo el fichero wtmp).
29. Haz una estadística (ordenada de mayor a menor) del número de veces que se ha conectado a petra cada usuario en este último mes (desde que se inicializo el fichero wtmp).



## 5. Ejercicios del apartado 7: Programación del shell

### 5.1. Ejercicios para realizar como trabajo no presencial

1. Realiza un programa del shell que muestre por pantalla los números del 1 al 100
2. Realiza un programa del shell que pase a minúsculas los nombres de todos los ficheros del directorio actual.
3. Realiza un programa del shell que edite un fichero (con cualquiera de los editores disponibles) en caso de que tengamos permiso de lectura y escritura sobre él. En caso de que sólo tengamos permiso de lectura, que lo visualice con el *less* y si no tenemos ninguno de los dos que nos muestre un error por pantalla.
4. Realiza un programa del shell que cambie el directorio actual al directorio raíz del sistema. ¿Cómo debo ejecutarlo?
5. Realiza un programa del shell que renombre todos los ficheros que hay en el directorio actual añadiendo a su nombre la extensión ".viejos".
6. Realiza un programa del shell que renombre todos los ficheros que se le pasan como parámetro añadiendo a su nombre la extensión ".viejos".
7. Realiza un programa del shell que, a partir de un fichero de texto, cree tantos ficheros como líneas tenga, conteniendo cada uno de ellos una línea del fichero. El nombre de cada fichero será el mismo que el fichero original pero añadiendo un . y el número de la línea de contiene.
8. Realiza un programa del shell que "borre" ficheros. El programa lo que hará será mover los ficheros que se le pasen como parámetros a un directorio denominado ".papelera" dependiente del directorio \$HOME del usuario. En caso de que no exista el directorio, deberá crearse. En caso de que ya exista un fichero con ese nombre en la papelera, se le añadirá un . y un número al final de su nombre (si borro el fichero f y ya existe uno con este nombre, lo intentaré con f.1; si ya existe uno con este nombre, lo intentaré con f.2; así sucesivamente hasta que encuentre uno que no exista).
9. Realiza un programa del shell que pare todos los procesos que tiene el usuario, a excepción del propio shell que está ejecutando el programa.
10. Realiza un programa del shell que elimine todos los procesos que tiene el usuario, a excepción del propio shell que está ejecutando el programa.
11. Realiza un programa del shell que mande una señal que pasaremos como parámetro a todos los procesos que tiene el usuario, a excepción del propio shell que está ejecutando el programa.
12. Realiza un programa del shell que muestre para cada usuario el nombre del login, el nombre real de usuario y los nombres de los grupos a los que pertenece.
13. Realiza un programa del shell que cree 50 ficheros en el directorio, que se llamen fich1, fich2, ..., fich50 y que el contenido de cada fichero sea 5 líneas con el propio nombre del fichero.
14. Realiza un programa del shell que muestre, para cada uno de los usuarios que están conectados en el sistema, su directorio HOME.
15. Realiza un programa del shell que renombre todos los ficheros del directorio que se apelliden .C, cambiando sus nombres a .c.
16. Realiza un programa del shell que renombre todos los ficheros del directorio, pasando sus nombres a minúsculas