

# PRACTICAS DE SISTEMA OPERATIVO UNIX

## PRACTICAS DE UNIX

### **PRACTICA 4. Comodines y caracteres de sustitución. Gestión de ficheros y selección de información.**

#### **Objetivos.**

En esta práctica se realizarán ejercicios de redirección, tuberías, filtros, caracteres de sustitución, selección de información y salidas divididas. Se pretende familiarizarse con el uso de caracteres especiales de redirección de información así como conseguir soltura en la selección de información de un fichero o entrada de información, teniendo en cuenta algún campo o información a partir de alguna columna.

#### **Herramientas.**

Los herramientas a utilizar serán :

```
tee          cut          find          grep          egrep
fgrep       uniq          cmp          diff          comm          sort
```

#### **DESARROLLO DE LA PRACTICA**

##### 1. Redirección y canalización de información.

- 1.1 Introduzca nuevas líneas al final de un fichero de texto que ya tenga, conservando el texto original.

```
cat >> fichero
```

- 1.2 Utilice la siguiente sentencia "\$ cat f1 f2". EL fichero f1 debe existir y el f2 no debe existir. Observe la salida por el monitor.

- a. Consiga que la salida de errores vaya a un fichero llamado "errores".

```
cat f1 f2 2> errores
```

- b. Haga ahora lo necesario para que el mensaje de error se desprecie.

```
cat f1 f2 2> /dev/nul
```

- 1.3 Visualice por pantalla un listado de los usuarios, que estén conectados en este momento al sistema y el número de ellos. Así mismo se debe almacenar el listado obtenido en un archivo llamado "presentes".

```
who | grep -n tty > presentes
```

- 1.4 Obtener un listado por pantalla de los usuarios conectados actualmente al sistema y su hora de conexión.

```
who
```

- 1.5 Visualice y almacene en un archivo llamado "TTYs" el nombre de los usuarios conectados, así como el tty asociado a cada usuario.

```
who | tee TTYs
```

##### 2. Caracteres de sustitución y selección de información.

- 2.1 Liste del directorio "/usr/bin":

- a. aquellos ficheros cuyo nombre empiece por "c"

```
find /usr/bin -name "c*" -print
```

- b. los ficheros que comiencen por alguna letra comprendida entre la "a" y la "p"

```
find /usr/bin -name "[a-p]*" -print
```

- c. todos aquellos ficheros cuya letra inicial del nombre no esté comprendida entre la "a" y la "p"

```
find /usr/bin ! -name "[d-p]*" -print
```

- d. todos aquellos ficheros cuyo nombre tenga cuatro letras.

```
find /usr/bin -name "?????" -print
```

- 2.2 Consiga un listado en el que aparezca únicamente el nombre y el tamaño de todos sus ficheros a partir de su directorio de conexión.

- 2.3 Cree un fichero que esté formado por el identificativo (UID), el nº de usuario y el nº de grupo de los usuarios dados de alta.

```
cut -d -f1,3,4 /etc/passwd > users
```

- 2.4 Obtenga un listado en el que aparezcan los nombres de los grupos de usuarios dados de alta en el sistema, así como los usuarios que formen parte de cada grupo.

```
cut -d: -1,4 /etc/group
```

- 2.5 Cree un fichero llamado "agenda1" cuyo contenido sea nombre, apellidos, prefijo y teléfono de una serie de personas (como separador de campos se utilizará el signo dos puntos ":").

- a. Liste los datos de las personas que vivan en Madrid (prefijo 91).

```
grep 91 agenda1
```

- b. Liste los datos de aquellos cuyo nombre empiece por "A" y su teléfono acabe por "00".

```
grep "^A" agenda1 | grep "00$"
```

- c. Liste el número de personas que no sean de Madrid.

```
grep -v "Madrid" agenda1
```

- d. Liste los datos de todas las personas de "agenda1", menos aquellos que se llamen "juan".

```
grep -v "juan"
```

- 2.6 Suponiendo que los números de los usuarios dados de alta van desde el 100 en adelante, visualice el nombre de las cuentas dadas de alta que no pertenezcan a usuarios.

```
grep "0[0-9][0-9$]"
```

### 3. Búsqueda de ficheros.

- 3.1 Liste todos los nombres de ficheros y directorios a partir de su directorio de conexión:

- a. utilizando el comando "find"

```
find $HOME -print
```

- b. utilizando "ls".

```
ls -R $HOME
```

- 3.2 Visualizar el contenido de todos los ficheros que se encuentren a partir de su directorio de conexión y que empiecen por "fich". Repita lo anterior pero pidiendo confirmación antes de visualizar.

```
find $HOME -name fich* -exec cat {} \;
```

```
find $HOME -name fich* -ok cat {} \;
```

- 3.3 Liste la ubicación y nombre de todos aquellos ficheros ordinarios del sistema de ficheros cuyo nombre empiece por "p" y ocupen más de un bloque.

```
find / -name "p*" -size +1 -type f -print
```

- 3.4 Liste por pantalla el nombre los ficheros ordinarios que cuelguen de su directorio de conexión.

```
find $HOME -type f -print
```

- 3.5 Visualizar un listado en dos columnas con el nombre y número de nodo-i de todos los directorios de su propiedad.

```
find / -type d -user f941162 -exec ls -di {} \;
```

- 3.6 Envíe a su propio correo un listado largo de todos los archivos de su propiedad que tengan únicamente permisos de ejecución.

```
find / -user f941162 -perm 111 | ls -li | mail f941162
```

- 3.7 Orden que permita borrar, solicitando confirmación, todos los archivos ordinarios de su propiedad que tenga en la estructura de directorios.

```
find / -user f941162 -type f -ok rm {} \;
```

- 3.8 Orden que busque a partir de cada uno de los caminos incluidos en su PATH, los ficheros que incluyan en su nombre la cadena "core".

```
for a in $PATH ^j do ^j find $a -name "*core*" -print ^j  
done
```

### 4. Manipulación de ficheros.

- 4.1 Liste por pantalla el número de personas que almacena el fichero "agenda1" sin repeticiones.

```
sort -u agenda1
```

- 4.2 Liste en orden alfabético de apellidos el fichero "agenda1".  
`sort +1 -d agenda1`
- 4.3 Cree otro fichero con la estructura de "agenda1" y llámelo "agenda2". Clasifique en orden inverso los ficheros "agenda1" y "agenda2", colocando la salida en el fichero "ageninv" y utilizando el segundo campo como clave de clasificación.  
`sort 1 +1 -r agenda1 agenda2 > ageninv`
- 4.4 Visualice las líneas del fichero "agenda2" ordenadas, suprimiendo todos menos la primera ocurrencia de las líneas que tengan el mismo apellido.  
`sort -d agenda2 | uniq -d -n0`
- 4.5 Visualice el fichero de palabras de paso, ordenado por identificativo de usuario (recuerde que se trata del tercer campo y que están separados por ":" ).  
`sort -d -t:/etc/passwd`
- 4.6 Partiendo de "agenda1" y "agenda2", obtenga un fichero llamado "telefonos" cuyo contenido sea la fusión de los dos anteriores sin repetir elementos y clasificado en primer orden por prefijo y en segundo por nombre.  
`sort +2 +0 -u agenda1 agenda2 > telefonos`
- 4.7 Genere un par de ficheros de texto "fich1" y "fich2" que tengan semejanzas. Utilice el comando "cmp" para ver cual es la primera diferencia entre ambos ficheros y en qué posición se ha producido.  
`cmp f1 f2`
- 4.8 Utilizando los mismos ficheros del ejemplo anterior, muestre por pantalla todas las diferencias que existan entre ambos.  
`cmp -l f1 f2`
- 4.9 Haga un listado:
- de todas las líneas del primero que no estén en el segundo y todas las líneas comunes  
`comm -2 f1 f2`
  - todas las líneas del segundo que no estén en el primero.  
`comm -13 f1 f2`