

# Tema 7 – Sistemas Informáticos

## Ejercicios 2020/2021

Para cada ejercicio debes entregar un archivo txt con tu nombre y número de ejercicio (ejemplo: **pedro\_martinez\_ortega\_ejercicio1.txt**).

### Ejercicio 1

¡Utiliza sólo rutas relativas!. Ejecuta como root sólo los comandos que lo requieren.

1) Crear el siguiente árbol de directorios usando sólo 1 comando:

```
ejer1
|-----trabajo
|         |-----urgente
|         |-----terminado
|
|-----personal
|         |-----documentos
|         |-----descargas
```

2) Crea un archivo dentro de **documentos** llamado **fecha.txt** que contenga la fecha de hoy (obtenida mediante un comando) con el formato: "miércoles, 12 de marzo de 2021 a las 15:00".

3) Añádele el permiso de escritura para otros usuarios al archivo creado anteriormente

4) Sitúate en **trabajo**. Desde ahí mueve el archivo **fecha.txt** a **terminado**

5) Crea un enlace simbólico llamado **enlfecha** dentro de **documentos** que apunte a **fecha.txt**.

6) Mueve el enlace recién creado al directorio **descargas**, ¿Te sigue funcionando? ¿Por qué?

7) Crea un archivo llamado **usuario.txt**, dentro de **documentos** que contenga el nombre del usuario actual (obtenido mediante un comando). Une los contenidos de **fecha.txt** y **usuario.txt** en un archivo nuevo llamado **login.txt** (créalo en el directorio **documentos**)

8) Cambia el propietario de **login** para que sea root el nuevo propietario. Añádele también el permiso de escritura al grupo del archivo por si no lo tuviera.

9) Intenta cambiar el contenido del archivo login por la frase “Hola como estas”. Explica si te deja o no, y sobre todo, por qué.

10) Sitúate en **urgente** y mueve el directorio descargas dentro del mismo. ¿Funciona ahora el enlace enlfeca que hay en el directorio descargas (apartado 6), y por qué?

11) Muestra todos los archivos regulares (no directorios por ej.) que su nombre contenga el texto conf, dentro del directorio /etc, y que además tengan un tamaño inferior a 3 kilobytes.

12) A partir del resultado del comando anterior, muestra los datos de dichos archivos (ls -l)

13) ¿Cómo mostrarías las características de un archivo en el mismo formato que el siguiente ejemplo?. Prueba con un archivo existente en tu ordenador:

El archivo **archivo.bin** pertenece a **arturo** y su tamaño es de **453** bytes

14) Muestra los grupos del sistema (/etc/group) que empiecen por la letra a. Solo la columna del nombre.

15) Muestra de los usuarios del sistema (/etc/group) las columnas del nombre, id del usuario e intérprete de comandos. Ordena la salida por id de mayor a menor.

16) Muestra los procesos del sistema, pero solo las columnas del usuario, PID, CPU y nombre del proceso, separadas por punto y coma.

17) Muestra el número de grupos del sistema que tienen al menos un usuario del cual son grupo secundario (última columna de /etc/group).

## Ejercicio 2

**A.** Usando el archivo **archivo1.txt**, escribe los comandos (con las expresiones correctas) para conseguir lo siguiente (usa comando egrep, usado en lugar de sed, ya que te resalta lo que la expresión regular ha encontrado como coincidencia). Es decir, no vale sólo con mostrar las líneas correctas sino que la expresión regular debe coincidir con lo que se busca (y tener sentido):

1. Muestra las líneas con teléfonos válidos
2. Muestra las líneas con DNIs válidos
3. Muestra las líneas con correos válidos
4. Muestra las líneas con IPs o MACs válidas
5. Muestra las líneas con etiquetas HTML de apertura y cierre bien puestas

## 6. Muestras las líneas con fechas correctas

**B.** Usando el archivo **archivo2.txt**, escribe los comandos para conseguir lo siguiente (primero debes sustituir lo que separa los 5 campos por un único separador):

1. Muestra el número de línea, nombre y el dinero, ordenados por nombre. Como separador usa el que quieras.
2. Muestra el nombre, calle y teléfono separados por ' <=> '

**Pista:** Con el comando `sed`, sustituye lo que separa a los campos por un único carácter (puedes concatenar varios comandos `sed` usando varios pipes, o con la opción `-e`).

[https://www.gnu.org/software/sed/manual/html\\_node/Multiple-commands-syntax.html](https://www.gnu.org/software/sed/manual/html_node/Multiple-commands-syntax.html)

**C.** A partir del archivo **archivo3.txt**, escribe los comandos para conseguir lo siguiente (quita además la primera línea del archivo en la salida):

1. Muestra sólo la columna del teléfono (sin espacios a los lados)
2. Muestra las columna del nombre y el home, separadas por un guión, y ordenadas alfabéticamente por nombre.

## Ejercicio 3

1. Utilizando los archivos: **cifs.txt**, **correos.txt**, **dominios.txt**, **fechas.txt**, **precios.txt** y **telefonos.txt**. Utilizando en cada apartado el archivo correspondiente, escribe los comandos que te muestren sólo las siguientes líneas de dichos archivos:

1. Fechas con el día, mes y año separados por "-" o por "/". Además los días y los meses pueden ser de 1 cifra o de 2, y el año de 2 o de 4 (de 3 no). No se debe comprobar si es un día o un mes válido en el calendario, sólo el formato.
2. Teléfonos de nueve cifras que además pueden llevar opcionalmente prefijo de país. El prefijo puede empezar por "+", por "0", o nada, seguido de 2 números. Después del prefijo (si lo lleva) puede llevar un guión, un espacio, o nada que lo separe del numero normal.
3. CIFs o DNIs: Los CIFs empiezan por una letra seguidos de 8 números, o también por una letra, seguida de 7 números y otra letra al final. Los DNIs (no vamos a comprobar DNI extranjero)

son 8 números y una letra al final. Con una única expresión hay que filtrar ambos.

4. Correos electrónicos con el siguiente formato: antes de la arroba pueden ir caracteres alfanuméricos, puntos y guiones, pero siempre comenzando por un alfanumérico. Después de la arroba, caracteres alfanuméricos o guion (nada más) el punto, y un dominio de 2 a 4 letras.
5. Dominios de internet con el siguiente formato. Pueden contener letras, números y guiones los nombres de dominio y subdominios, siempre que empiecen por letra (todo minúscula). La longitud también será mínimo de 1 carácter. La extensión final del dominio (.com,.net etc...), sólo podrá contener letras y una extensión entre 2 y 4 caracteres. Se podrán anidar todos los subdominios que se quieran -> (este.es.un.dominio.muy-muy.largo.com)
6. Precios que sean un número entero, o también que puedan tener uno o dos decimales máximo separados por punto o por coma. Siempre debe haber un número en la parte entera (,23 no es válido, debería ser 0,23 en todo caso).

2. Utilizando sed y awk, apartir del archivo **reservas.txt** saca sólo las líneas de las personas que vayan a comer algún tipo de pescado en el formato: "**En la mesa 9 comerán Lubina 3 personas a las 13:00**"

3. Utilizando la posibilidad de awk de realizar operaciones matemáticas con los valores de las columnas, vamos a calcular la suma de 2 números. A partir del archivo **numeros.txt** te debe mostrar lo siguiente:

**1 + 4 = 5**

4. Filtra el contenido de **productos.txt** para que sólo muestre aquellos que contengan una referencia (primer campo) válida. Una referencia válida es la que empieza por una categoría (de una o dos cifras), seguida de un punto, un número de 5 cifras, un guión y una letra(ejemplo: 15.40234-A).

Con los productos filtrados, muéstralos con el siguiente formato (no importan los decimales que te salgan en el total. Si quieres que siempre te salgan 2, mira los ejemplos de printf en el anexo de AWK del aula virtual):

**15.40234-A – Toalla: 2.45€ x 10 unidades = 24.50€. Con IVA = 29.645€**

5. Del archivo **passwords.txt** muestra sólo las contraseñas que cumplan lo siguiente (para ello puede serte útil simplemente ocultar las líneas que no cumplan la condición con el comando `sed`, y concatenar varios):

- Tengan entre 6 y 12 caracteres
- Que al menos tengan una letra mayúscula, una minúscula y un número (las 3 cosas)
- Que no contengan los caracteres espacio o tabulador