

1- Supongamos que dentro del directorio /tmp existe un directorio de nombre FS dentro del cual existen los directorios de nombres TEORIA y PRACTICAS, ambos contienen únicamente archivos regulares. Realice cada uno de los apartados siguientes utilizando para cada uno una única orden preferentemente y sin utilizar la orden cd para cambiar de directorio. Utilice la variable \$HOME para hacer generales sus respuestas. Deberá tenerse en cuenta que desconocemos en qué directorio de trabajo nos encontramos situados; es decir, sus respuestas deberán ser válidas independientemente del directorio de trabajo actual.

a) [2*] crear dentro del directorio inicial asociado a su cuenta un directorio de nombre FS

```
mkdir $HOME/FS
```

b) [2*] crear dentro del directorio FS creado en (a) dos directorios de nombres m1T y m1P

```
mkdir $HOME/FS/{m1T,m1P}
```

c) [2*] copiar en el directorio m1T creado en (b) los archivos existentes dentro del directorio TEORIA (mencionado al comienzo del enunciado) cuyo nombre termine en 2015

```
cp /tmp/FS/TEORIA/*2015 $HOME/FS/m1T
```

2- [6*] Orden para añadir permisos de ejecución a usuario, grupo y otros, para todos los archivos/directorios del directorio /tmp cuyo nombre contenga solo 1 carácter (Solo una orden que sea válida independientemente del directorio de trabajo):

```
chmod a+x /tmp/?
```

3- [6*] Con varias órdenes en una única línea de órdenes, querremos en que que muestre un mensaje indicando si como directorio de trabajo tenemos un descendiente de tmp o no (queremos aludir al directorio tmp que cuelga directamente de /)

```
pwd | cut -d/ -f2 | grep tmp && echo si || echo no
```

4- [12*] Cree un guion que se ejecutará pasándole dos argumentos: el primero será un literal y el segundo será la ruta de un archivo. Tenemos el directorio ArchivosAyuda (dentro del directorio inicial de su cuenta de usuario) con cierto número de archivos regulares; el guion deberá comprobar si existe dentro de dicho directorio un archivo regular con el nombre proporcionado como argumento, y en ese caso debe...

a) mostrar su contenido en pantalla

b) añadir al archivo cuya ruta se pasa como segundo argumento una línea con este contenido: nombre del archivo visualizado:nº de día dentro del año:usuario

Programa la actuación adecuada ante las distintas situaciones de error que puedan ocurrir.

```
#!/bin/bash
```

```
if [ $# -ne 2 ]; then
```

```
echo Error en $0: Debe dar dos argumentos
```

```
exit
```

```
fi
```

```
if ! test -a $HOME/ArchivosAyuda/$1 ; then
```

```
echo Error en $0: no existe el archivo $1
```

```
exit
```

```
fi
```

```
if ! test -r $HOME/ArchivosAyuda/$1 ; then
```

```
echo Error en $0: no tiene permisos de lectura sobre $1
```

```
exit
```

```
fi
```

```
cat $HOME/ArchivosAyuda/$1
```

```
echo $1:`date +%j`:$USER >> $2
```