

Nombre

DNI/NIE

PC

| | | |
|--|--|--|
| | | |
|--|--|--|

Dispone de **DOS HORAS Y MEDIA** para realizar esta parte del examen. Al finalizar, entregue cada una de sus respuestas en un fichero de **TEXTO PLANO** separado para cada pregunta. La entrega se realizará a través de las tareas que se habilitarán para ello en el Campus Virtual.

- » Escriba su nombre completo al principio del fichero.
- » Indique el número de pregunta que responde y, en el caso de que la pregunta tenga apartados, numere las respuestas de cada apartado que responda.

Cuando haya terminado, rellene esta hoja con sus datos y entréguela al profesorado antes de salir del laboratorio.

En todas las cuestiones, asuma que el sistema es un CentOS 7.

1 (4 puntos) Cuatro cuestiones sobre órdenes (todas puntúan lo mismo). Todas las cuestiones planteadas deben resolverse en una sola línea, empleando una combinación de órdenes comunes y tuberías. No se considerarán válidas las respuestas que utilicen más de una línea o scripts.

CUESTIÓN A. Obtener el nombre del usuario que lleve más tiempo en sesión y la fecha y la hora en que inició su sesión.

CUESTIÓN B. Encontrar los ficheros regulares que puedan ser ejecutados por algún usuario del sistema y estén ubicados en el directorio `"/etc"`, o en cualquiera de sus descendientes, que contengan al principio de una de sus líneas la cadena `"DEVICE="`.

(Nota: El carácter `"` no forma parte del patrón a buscar).

CUESTIÓN C. Mostrar por pantalla los procesos del sistema que no son propiedad del usuario `"root"`. Para cada uno de estos procesos se deberá mostrar, sólo y en el orden que se especifica, la siguiente información: usuario propietario, identificador de proceso y orden que dio lugar al proceso. El listado deberá estar ordenado alfabéticamente por el usuario propietario.

CUESTIÓN D. Encontrar en el sistema todos los nombres de fichero que se repitan dos o más veces. Para cada uno de estos ficheros debe mostrar una línea por pantalla que contenga, sólo y en el orden que se especifica, la siguiente información: nombre del fichero, número de veces con que se repite el nombre del fichero. Las líneas que se muestren deben estar ordenadas decrecientemente por el número de veces con que se repite el nombre del fichero.

2 (2 puntos) Desarrollar un script en bash al que se le pase como argumento un directorio y muestre por pantalla la primera línea de cada fichero “.txt” que se encuentre dentro de ese directorio siguiendo este formato:

```
=====
Fichero: fichero1.txt
=====
Primera línea del fichero fichero1.txt
=====
Fichero: fichero2.txt
=====
Primera línea del fichero fichero2.txt
...
```

El script debe realizar las siguientes comprobaciones:

1. Que el número de argumentos de entrada sea el correcto.
2. Que el argumento suministrado indique un directorio válido.
3. Que dentro del directorio indicado existan ficheros “.txt”.

Además, el script debe cumplir con estas especificaciones:

1. Solo deben buscarse los ficheros que estén en el directorio exacto indicado como parámetro. No se buscarán ficheros en los posibles subdirectorios que desciendan de él.
2. Por pantalla sólo deben aparecer los mensajes generados por el script, es decir, si el script ejecuta una orden, los mensajes que pueda emitir esa orden no deben aparecer por pantalla.
3. Los mensajes que genere el script deben emitirse por el canal apropiado, en función si se trata de la salida esperada o de mensajes de error.
4. Si se cumplen todas las comprobaciones, entonces el script debe finalizar retornando el valor apropiado que indica una ejecución sin error al shell del sistema. En caso contrario, debe finalizar retornando un valor que indique al shell del sistema una ejecución con error, permitiendo identificar la causa del error.

3 (2 puntos) Desarrollar un script en bash al que se le pase como argumentos nombres de cuentas de usuarios y escriba en el fichero “./NumFileUser.txt” el usuario, el número de procesos lanzados por el usuario y el número de ficheros en el directorio de inicio para cada usuario cuyo directorio de inicio sea descendiente del directorio “/home.” Por ejemplo:

```
"El usuario Jorge tiene 3 procesos activos, su directorio de inicio es /home/jorge y en él hay 35 fichero(s)"
```

Si el usuario no existe o su directorio de inicio no desciende de “/home”, se mostrará un mensaje por pantalla que lo notifique. Algunos ejemplos:

```
"El usuario Jorge no existe"
```

```
"El directorio de inicio del usuario Jorge (/DirJorge) no desciende de /home"
```

El desarrollo de este script puede ser un poco largo, motivo por el que te recomendamos que te centres en resolver lo que se pide y no inviertas tiempo en validar los argumentos. Además, podrás alcanzar el 80% de la puntuación si tu script comprueba todos los archivos que son propiedad del usuario sin tener en cuenta su ubicación. Para alcanzar el 100% de la puntuación deberás comprobar que los archivos pertenezcan al usuario y estén en un directorio que descienda de “/home”.