

1. En Linux, los archivos cuyo nombre comienzan con un punto (es decir, el caracter “.”) son considerados como “archivos ocultos”, que por defecto no son listados por el comando `ls` (ni por los administradores gráficos de archivos). Aparte de esta característica, son archivos normales, pero que son usados como archivos de configuración del sistema o de algunos programas. La opción `-a` del comando `ls` es usada para listar todos los archivos en una carpeta, incluyendo los ocultos. Usando comandos **Bash**, realice las siguientes tareas
 - (a) Liste todos los archivos de la carpeta principal en los computadores del laboratorio, incluyendo los ocultos.
 - (b) Filtre la lista de archivo, usando el comando `grep`, para listar sólo los archivos ocultos.
 - (c) Guarde la lista obtenida en el punto anterior en un nuevo archivo con nombre `ocultos.txt`.
 - (d) Cambie el nombre a algún archivo que haya creado para convertirlo en archivo oculto. Verifique que ahora `ls` no lo lista por defecto.
2. Investigue qué efecto tiene la opción `-o` en el comando `grep` y ejecute algunos comandos de prueba para verificar su funcionamiento. Luego de esto, ejecute comandos **Bash** que cuenten cuántas veces se repite la letra “a” en el extracto de el Quijote que usó en la [guía 01](#) (archivos `c1.txt` a `c5.txt`). Si usó las características de redireccionamiento de **Bash** (el caracter `|`) debiese realizar esta tarea con una única línea de comando.
3. El comando `echo` despliega en la salida principal (la pantalla) un mensaje de texto indicado.
 - (a) Pruebe qué hace el comando

```
echo 'Hola Mundo'
```

- (b) Con la opción `-e` el comando `echo` reconoce algunos caracteres especiales, por ejemplo `\n` es reconocido como un salto a una nueva línea. Verifique esto ejecutando

```
echo -e 'Hola\nMundo'
```

- (c) Investigue qué otros caracteres especiales son reconocidos por `echo -e`.
4. Además de los caracteres de redireccionamiento `>` y `|`, existe `>>`, que también redirecciona la salida de un comando a un archivo. La diferencia entre `>` y `>>` es que en el primer caso el nuevo archivo se crea desde cero (y si ya existe, se reemplaza por el nuevo), mientras que `>>` agrega la salida al final de un archivo preexistente.

Para ejercitar lo anterior, haga algunas pruebas simples con `>` y `>>`. Luego de esto escriba un comando **Bash** que junte todos los trozos del texto del Quijote (archivos `c1.txt` a `c5.txt`), lo guarde en el nuevo archivo `Quijote.txt`, y que luego le agregue al final una nueva línea con la palabra ‘FIN’. Nuevamente, puede realizar todo esto con una línea de comandos.