|  |
| --- |
| Laboratorio de Computación  Salas A y B |
|  |

|  |  |
| --- | --- |
| Profesor: | Karina García Morales |
| Asignatura: | Fundamentos de Programación |
| Grupo: | 1121 |
| No de Práctica(s): | 02 |
| Integrante(s): | Valle Olivas Guillermo |
|  |  |
|  |  |
|  |  |
| No. de Equipo de cómputo empleado: |  |
| Semestre: | 2019-1 |
| Fecha de entrega: |  |
| Observaciones: |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Práctica** **02: GNU/Linux**

**Objetivo:**

Conocer la importancia del sistema operativo de una computadora, así como sus funciones. Explorar un sistema operativo GNU/Linux con el fin de conocer y utilizar los comandos básicos en GNU/Linux.

**Actividades:** ƒ

\*Iniciar sesión en un sistema operativo GNU/Linux y abrir una “terminal” ƒ

\*Utilizar los comandos básicos para navegar por el sistema de archivos. ƒ

\*Emplear comandos para manejo de archivos.

**Introducción**

El Sistema Operativo es el conjunto de programas y datos que administra los recursos tanto de hardware (dispositivos) como de software (programas y datos) de un sistema de cómputo y/o comunicación. Además funciona como interfaz entre la computadora y el usuario o aplicaciones.

En la actualidad existen diversos sistemas operativos; por ejemplo, para equipos de cómputo están Windows, Linux, Mac OS entre otros. Para el caso de dispositivos móviles se encuentran Android, IOS, Windows Phone entre otros. Cada uno de ellos tiene diferentes versiones y distribuciones que se ajustan a los diversos equipos de cómputo y comunicación en los que trabajan

Los componentes de un sistema operativo, de forma general, son:

\* Gestor de memoria,

\* Administrador y planificador de procesos,

\* Sistema de archivos y,

\*Administración de E/S.

Comúnmente, estos componentes se encuentran en el kernel o núcleo del sistema operativo.

En cuanto a la Interfaz con el usuario, las hay de tipo texto y de tipo gráfico. En la actualidad, es común trabajar con la interfaz gráfica ya que facilita mucho seleccionar la aplicación a utilizar; inclusive esta selección se hace “tocando la pantalla” (técnica touch).

Sin embargo, cuando se desarrollan proyectos donde se elaborarán documentos y programas es necesario el uso de dispositivos de entrada y salida (hardware) y aplicaciones en modo texto (software)

**Sistema Operativo Linux**

Linux es un sistema operativo tipo Unix de libre distribución para computadoras personales, servidores y estaciones de trabajo.

El sistema está conformado por el núcleo (kernel) y un gran número de programas y bibliotecas. Muchos programas y bibliotecas han sido posibles gracias al proyecto GNU, por lo mismo, se conoce a este sistema operativo como GNU/Linux.

**Software libre**

Un software libre es aquel que se puede adquirir de manera gratuita, es decir, no se tiene que pagar algún tipo de licencia a alguna casa desarrolladora de software por el uso del mismo.

Además, que un software sea libre implica también que el software viene acompañado del código fuente, es decir, se pueden realizar cambios en el funcionamiento del sistema si así se desea.

Linux se distribuye bajo la Licencia Pública General de GNU por lo tanto, el código fuente tiene que estar siempre accesible y cualquier modificación o trabajo derivado debe tener esta licencia.

**Licencia GNU**

La Licencia Pública General de GNU o GNU General Public License (GNU GPL) es una licencia creada por la Free Software Foundation en 1989 y está orientada principalmente a proteger la libre distribución, modificación y uso de software.

Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

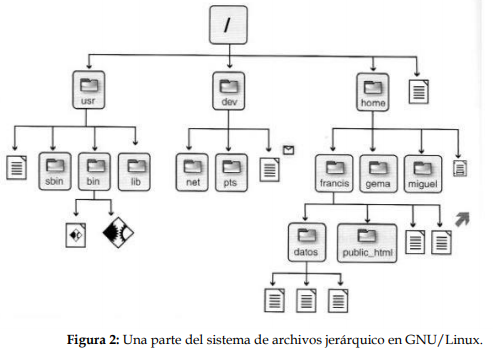
**Kernel de GNU/Linux**

El kernel o núcleo de linux se puede definir como el corazón del sistema operativo. Es el encargado de que el software y el hardware del equipo se puedan comunicar. Sus componentes son los que se mencionaron en la introducción de esta práctica.



De la figura 1, se puede observar que entre el kernel y las aplicaciones existe una capa que permite al usuario comunicarse con el sistema operativo y en general con la computadora, a través de programas que ya vienen instalados con la distribución de Linux (Debian, Ubuntu, Fedora, etc.) y trabajan ya sea en modo gráfico o en modo texto. Uno de estos programas es el Shell.

La estructura de Linux para el almacenamiento de archivos es de forma jerárquica; por lo que la carpeta o archivo base es “root” (raíz) la cual se representa con una diagonal (/). De este archivo raíz, parten todos los demás. Los archivos pueden ser carpetas (directorios), de datos, aplicaciones, programas, etc.



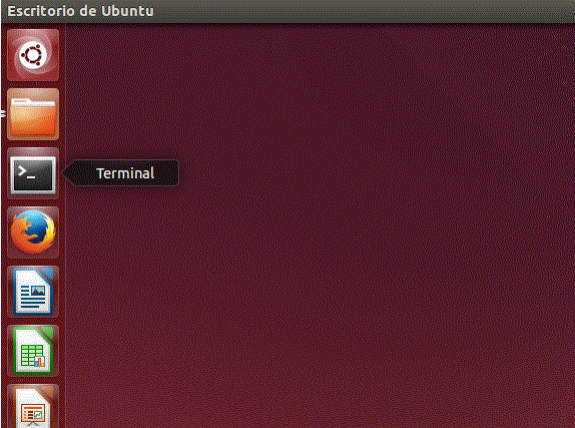
**Interfaz de línea de comandos (CLI) o shell de GNU/Linux**

El Shell de GNU/Linux permite introducir órdenes (comandos) y ejecutar programas en el sistema operativo. Todas las órdenes de UNIX/Linux son programas que están almacenados en el sistema de archivos y a los que llamamos comandos, por lo tanto, todo en GNU/Linux se puede controlar mediante comandos.

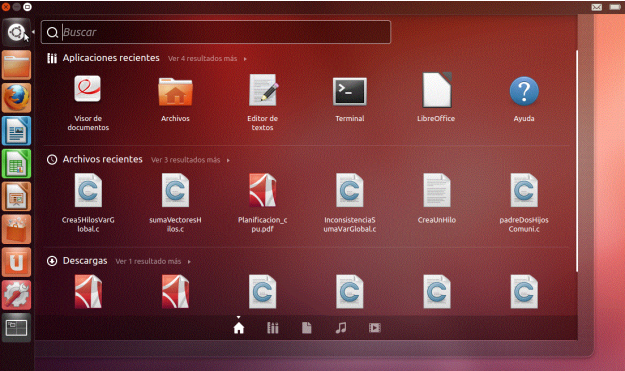
**Comandos básicos**

Para trabajar en Linux utilizando comandos, se debe abrir una “terminal” o “consola” que es una ventana donde aparece la “línea de comandos” en la cual se escribirá la orden o comando. La terminal permite un mayor grado de funciones y configuración de lo que queremos hacer con una aplicación o acción en general respecto a un entorno gráfico.

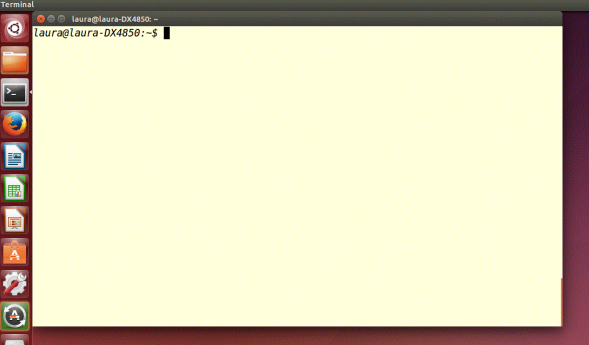
El proceso de abrir una terminal varía dependiendo del entorno gráfico. Por lo general hay un área de “aplicaciones” donde se selecciona terminal o consola.



O bien en el ícono de aplicaciones en la línea de “buscar” escribir “terminal” si es que no está a la vista el ícono de terminal.



Una vez teniendo una terminal abierta, estamos listos para introducir comandos.

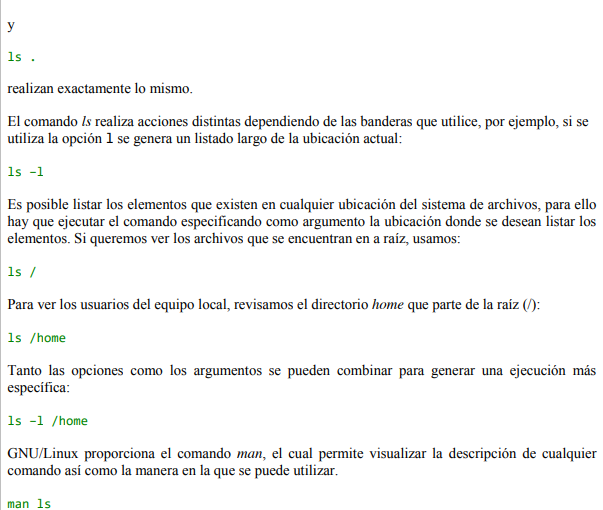
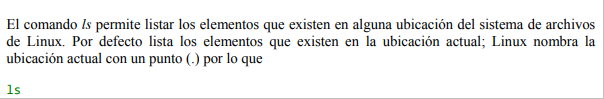


La sintaxis que siguen los comandos es la siguiente:

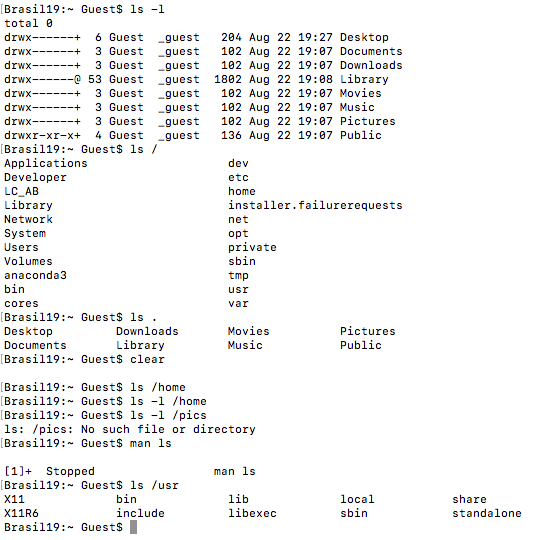
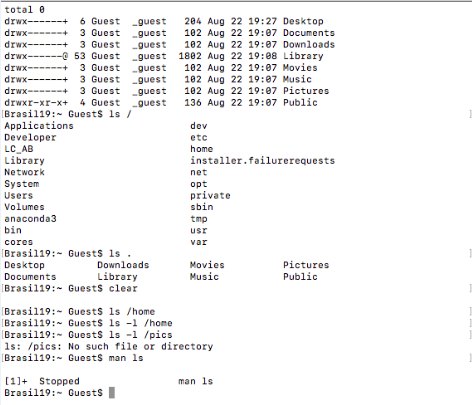
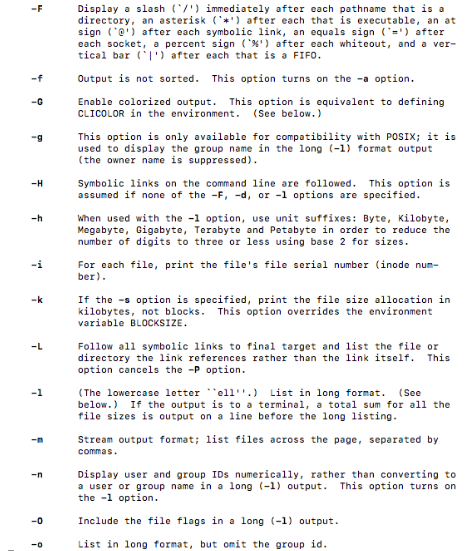
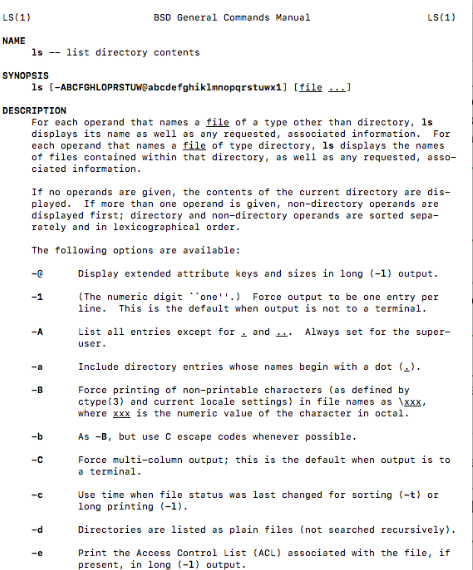
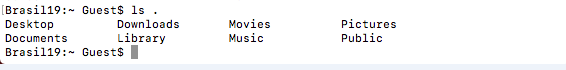
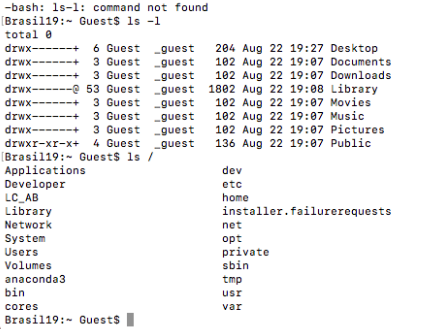
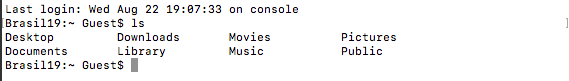
comando [-opciones] [argumentos]

Esto es, el nombre del comando, seguido de algunas banderas (opciones) para modificar la ejecución del mismo y, al final, se puede incluir un argumento (ruta, ubicación, archivo, etcétera) dependiendo del comando. Tanto las opciones como los argumentos son opcionales.

**Ejemplo (comando ls)**



Comandos enviados durante la práctica:



Antes de revisar otros comandos, es importante aprender a “navegar” por el sistema de archivos de Linux en modo texto. Basándonos en la Figura 2 de esta práctica, si deseamos ver la lista de los archivos del directorio usr, podemos escribir el comando:



Esto es, el argumento se inicia con / indicando que es el directorio raíz, seguido de usr que es el nombre del directorio. Cuando especificamos la ubicación de un archivo partiendo de la raíz, se dice que estamos indicando la “ruta absoluta” del archivo.



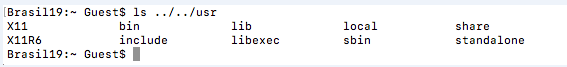
Existe otra forma de especificar la ubicación de un archivo, esto es empleando la “ruta relativa”. Si bien el punto (.) es para indicar la ubicación actual, el doble punto (..) se utiliza para referirse al directorio “padre”. De esta forma si deseamos listar los archivos que dependen de mi directorio padre se escribe el siguiente comando:



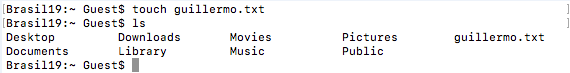
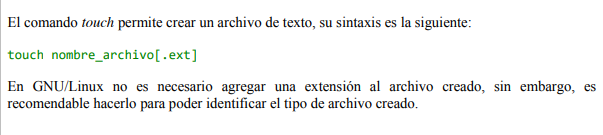
Se pueden utilizar varias referencias al directorio padre para ir navegando por el sistema de archivos, de tal manera que se realice la ubicación de un archivo a través de una ruta relativa. De la Figura 2, si nuestra cuenta depende de home, la ruta relativa para listar los archivos del directorio usr es:



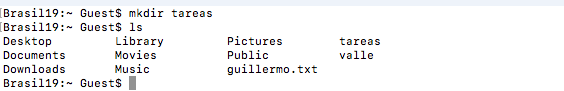
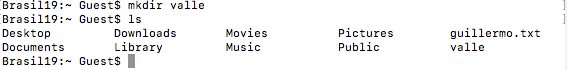
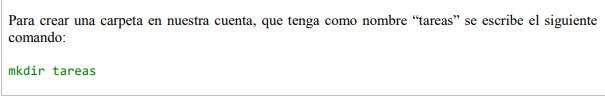
Con los primeros dos puntos se hace referencia al directorio home, con los siguientes dos puntos se refiere al directorio raíz, y finalmente se escribe el nombre del directorio usr.



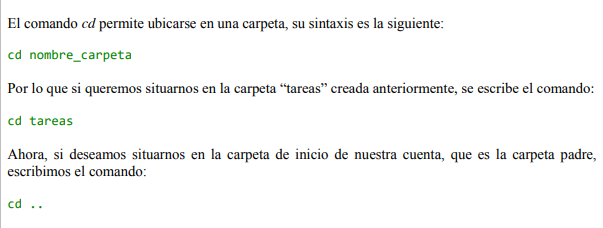
**Ejemplo (comando touch)**



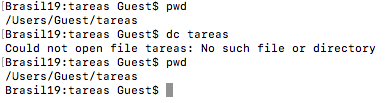
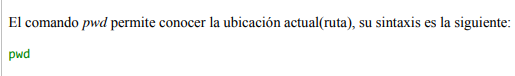
**Ejemplo (comando mkdir)**



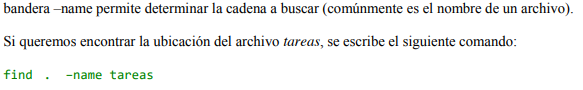
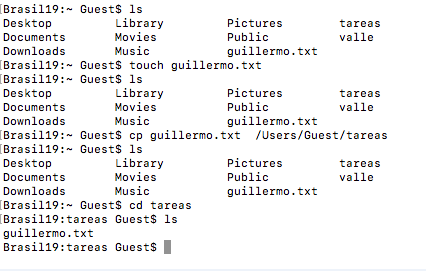
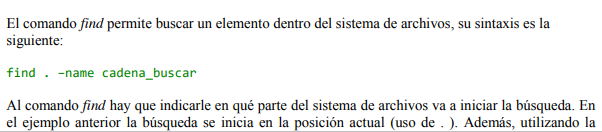
**Ejemplo (comando cd)**



**Ejemplo (comando pwd)**



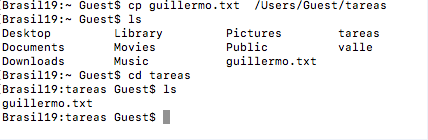
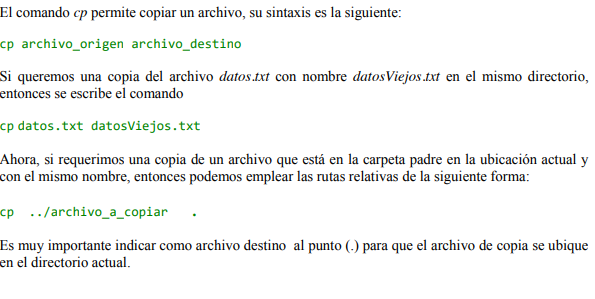
**Ejemplo (comando find)**



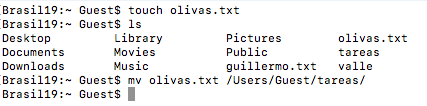
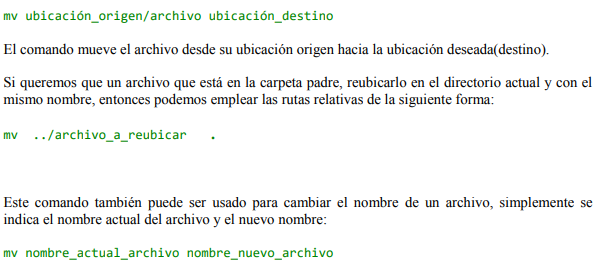
**Ejemplo (comando clear)**



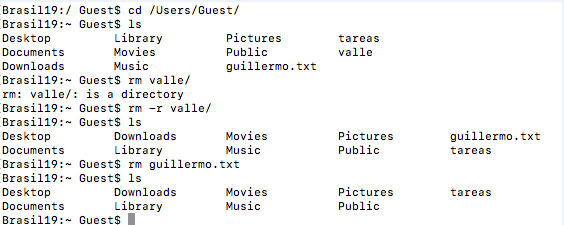
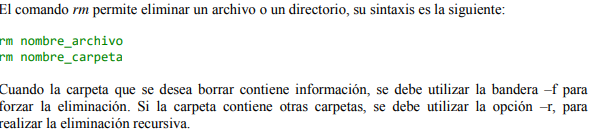
**Ejemplo (comando cp)**



**Ejemplo (comando mv)**



**Ejemplo (comando rm)**

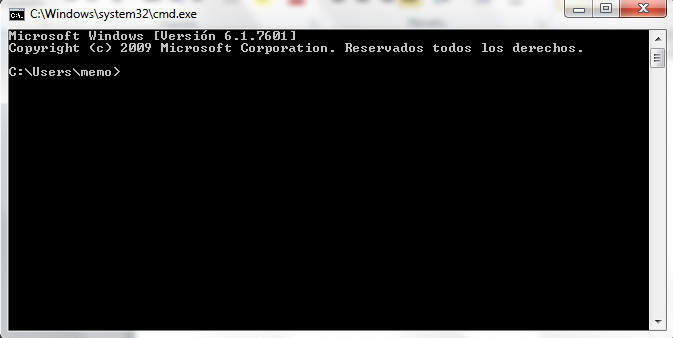


**Ejercicios de tarea**

Debido a que el sistema operativo que manejo en casa es Windows, no era posible enviar los comandos trabajados en el laboratorio, así que fue necesario auxiliarme de una tabla de equivalencias de comandos entre Linux y Windows.



Para abrir la consola de Windows en el ícono de aplicaciones, en la línea de “buscar” se escribe “cmd”



Actividades:

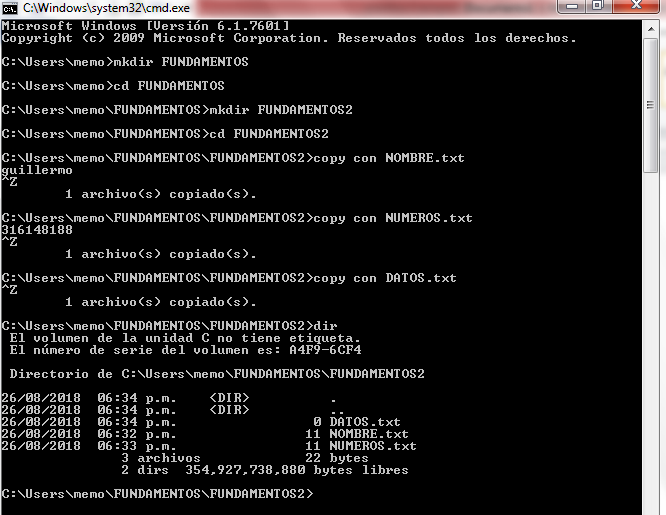
1. Crea un directorio que se llame "FUNDAMENTOS" y entra a el.

2. Crea un directorio que se llame "FUNDAMENTOS2" y entra a el.

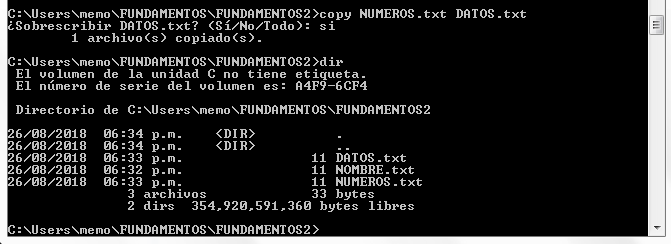
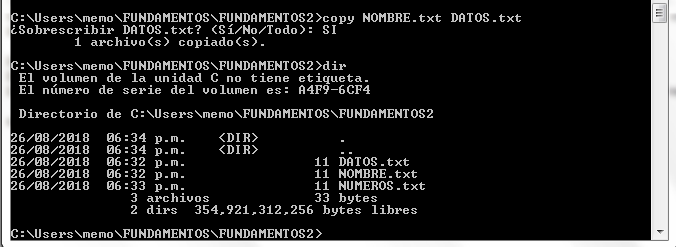
3. Crea un archivo llamado "NOMBRE" con tu primer nombre (puedes utilizar algÃºn editor grÃ¡fico de Linux en terminal).

4. Crea un archivo llamado "NUMEROS" con tu nÃºmero de cuenta.

5. Crea un archivo nuevo llamado "DATOS" y copia los datos de los archivos "NOMBRE" y "NUMEROS".



6. Muestra en pantalla el contenido de ambos archivos, uno a la vez y con que comando lo muestras.



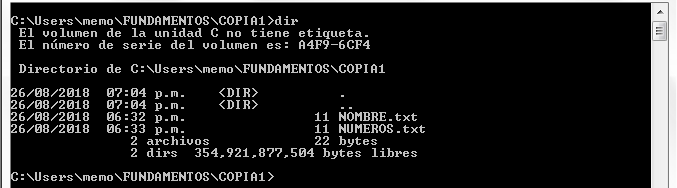
7. Salir del directorio FUNDAMENTOS2.

8. Crea un directorio que se llame COPIA1.



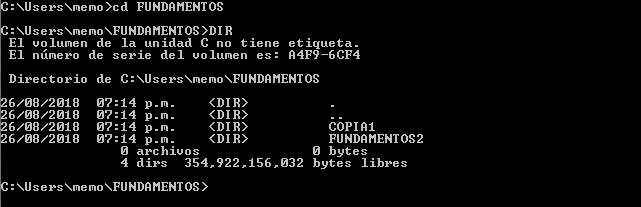
9. Copia los archivos NOMBRE y NUMEROS que creaste, al directorio COPIA1.

10. Muestra el contenido de tu carpeta.



13. Muestra los permisos de los archivos contenidos en el directorio FUNDAMENTOS y en DATOS, explica cada uno de ellos.

14. Indica el directorio en el que te encuentras y con que comando lo muestras.



15. Teclea el comando cal y escribe lo que muestra.

16. Teclea el comando date y escribe la salida.

El comando cal en Windows no muestra nada, en Linux muestra un calendario.

El comando date muestra la fecha del día en que se envió el comando



.

**Conclusiones**

Honestamente me resultó muy difícil llevar a cabo la práctica debido a que nunca había trabajado en la terminal, no sabía con certeza lo que estaba haciendo, metía un comando y no sabía cómo darme cuenta si lo había hecho bien o mal, me tomó algo de tiempo darme cuenta cuando los comandos son ejecutados de forma exitosa, pero al finalizar la práctica pude ser capaz de familiarizarme un poco con la terminal.

Con los ejercicios de tarea no me fue tan fácil, debido a que los comandos en Linux no son iguales a los de Windows, y la terminal no es tan amigable, de hecho tuve que invertir algo de tiempo en buscar los comandos necesarios para poder llevar a cabo los ejercicios en Windows.

**Bibliografía** ƒ

\*Óscar Vicente Huguet Soriano, Sonia Doménech Gómez. Introducción a Linux. [Figura 1]. Consulta: Junio de 2015. Disponible en: <http://mural.uv.es/oshuso/81_introduccin_a_linux.html>

\*Pablo Delgado. Integración de sistemas. Linux y su sistema gestor de ficheros (descripciones).[Figura 2]. Consulta agosto de 2016. Disponible en: <http://todobytes.es/2014/09/integracion-de-sistemas-linux-y-su-sistema-gestor-deficheros-descripciones/>