



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Karina García Morales

*Asignatura:* Fundamentos de programación

*Grupo:* 20

*No. de práctica(s):* 01

*Integrante(s):* Juan Manuel Cuellar Orbezo

*No. de lista o brigada:* 04

*Semestre:* 2024-2

*Fecha de entrega:* 20 de febrero del 2024

*Observaciones:* -

**CALIFICACIÓN:** \_\_\_\_\_

# Práctica 2 - GNU/Linux

## Objetivo

El alumno identificará al sistema operativo como una parte esencial de un sistema de cómputo. Explorará un sistema operativo GNU/Linux con el fin de conocer y utilizar sus comandos básicos.

## Desarrollo

### Sistema Operativo

Si la computadora fuera una película, el sistema operativo es el director que se encarga de que todo funcione correctamente. Es como el software maestro que controla la memoria, el disco duro, los programas y los dispositivos conectados a la computadora. Algunos ejemplos de sistemas operativos populares son Windows, macOS y Linux.

### El Kernel de GNU/Linux: El corazón del sistema

En el caso de GNU/Linux, el sistema operativo se compone de dos partes principales: el Kernel y las utilidades GNU. El Kernel es como el corazón del sistema, controlando el acceso al hardware y gestionando los recursos de la computadora. Las utilidades GNU son programas que le permiten interactuar con el sistema operativo y realizar tareas como navegar por archivos, editar textos y ejecutar programas.

### Comandos básicos de terminal

Aprender los comandos básicos de la terminal en Linux nos permite comunicarnos directamente con el sistema operativo, dando así un control más preciso sobre la computadora. Es como aprender un nuevo idioma que permite hablar directamente con el director de la película, en lugar de solo recitar el libreto.

Con los comandos básicos se pueden realizar tareas como:

- Navegar por archivos y carpetas
- Crear y eliminar archivos y carpetas
- Editar archivos de texto
- Controlar procesos
- Configurar el sistema

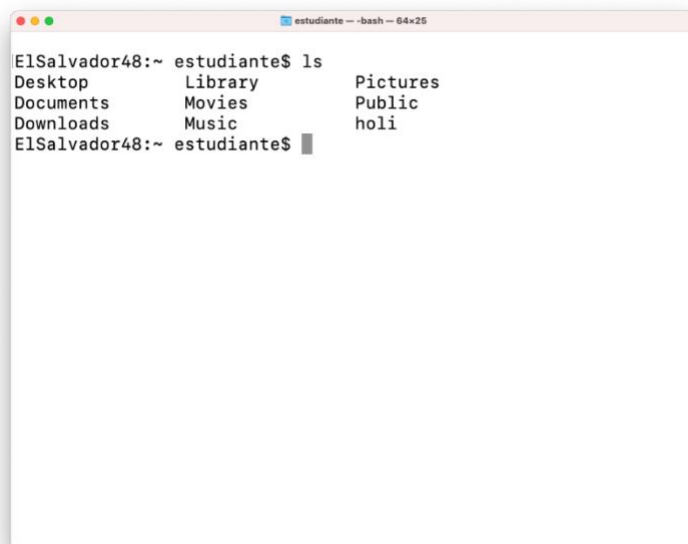
Aprender estos comandos abre un mundo de posibilidades y permite aprovechar al máximo la potencia y flexibilidad de la computadora.

## Glosario de comandos en Linux

A continuación, una pequeña lista de comandos importantes en Linux y algunos ejemplos de estos.

Comando	Mnemotecnia	Explicación
<b>ls</b>	<b>LIST</b>	Lista los archivos y carpetas del directorio actual.
<b>touch</b>	-	Crea un archivo vacío.
<b>mkdir</b>	<b>MAKE DIRECTORY</b>	Crea una nueva carpeta.
<b>cd</b>	<b>CHANGE DIRECTORY</b>	Cambia el directorio actual.
<b>pwd</b>	<b>PRINT WORKING DIRECTORY</b>	Muestra la ruta del directorio actual.
<b>find</b>	<b>FIND</b> = ENCONTRAR	Busca archivos y carpetas en el sistema.
<b>clear</b>	<b>CLEAR</b> ≈ CLEAN	Limpia la pantalla.
<b>cp</b>	<b>COPY</b>	Copia archivos o carpetas.
<b>mv</b>	<b>MOVE</b>	Mueve archivos o carpetas.
<b>rm</b>	<b>REMOVE</b>	Elimina archivos o carpetas.
<b>man</b>	<b>MANUAL</b>	Muestra la página de manual de un comando, con información detallada sobre su uso y opciones.
<b>cal</b>	<b>CALENDAR</b>	Muestra un calendario del mes actual o del mes especificado.
<b>date</b>	<b>DATE</b> = FECHA	Muestra la fecha y hora actual, o permite configurarla.

A pesar de que la lista es bastante sencilla se incluyen a continuación varios ejemplos de ciertos comandos.

A screenshot of a terminal window titled 'estudiante -- -bash -- 64x25'. The prompt is 'ElSalvador48:~ estudiante\$'. The command 'ls' has been entered, and the output is a three-column list of directories: Desktop, Documents, Downloads, Library, Movies, Music, Pictures, Public, and holi. The prompt is now 'ElSalvador48:~ estudiante\$' with a cursor.

```
ElSalvador48:~ estudiante$ ls
Desktop      Library      Pictures
Documents    Movies       Public
Downloads    Music       holi
ElSalvador48:~ estudiante$
```

*Img. 01 – Captura de pantalla mostrando el comando ls. Se puede observar una lista de los archivos y directorios en esta ubicación.*

```
ElSalvador48:~ estudiante$ rm ma
manuel.png maquina48.txt
ElSalvador48:~ estudiante$ rm ma
manuel.png maquina48.txt
ElSalvador48:~ estudiante$ rm ma
rm: ma: No such file or directory
ElSalvador48:~ estudiante$ rm maquina48.txt
ElSalvador48:~ estudiante$ rm manuel.png
ElSalvador48:~ estudiante$ ls
Desktop      Library      Pictures      cuellar.png
Documents    Movies       Public
Downloads    Music        clase
ElSalvador48:~ estudiante$ rm cuellar.png
ElSalvador48:~ estudiante$ ls
Desktop      Library      Pictures
Documents    Movies       Public
Downloads    Music        clase
ElSalvador48:~ estudiante$ rm -r clase
ElSalvador48:~ estudiante$ ls
Desktop      Downloads    Movies      Pictures
Documents    Library      Music       Public
ElSalvador48:~ estudiante$
```

Img. 02 – Captura de pantalla mostrando el comando rm. Nótese como se debe de agregar la especificación -r para indicar la recursividad de la acción.

```
LS(1)                                General Commands Manual
LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [-@ABCFGHILOPRSTUWabcdefghiklmnopqrstuvwx1%,] [--color=when]
    [-D format] [file ...]

DESCRIPTION
  For each operand that names a file of a type other than directory
, ls displays its name as well as any requested, associated informatio
n. For each operand that names a file of type directory, ls displays the
names of files contained within that directory, as well as any requeste
d, associated information.

  If no operands are given, the contents of the current directory a
```

Img. 03 – Captura de pantalla mostrando el comando man. La información es muy completa. Se puede salir de esta vista con la tecla ESC.

```
ElSalvador48:~ estudiante$ ls /usr/
X11          lib          sbin
X11R6       libexec     share
bin         local       standalone
ElSalvador48:~ estudiante$ ls
Desktop      Library     Pictures
Documents    Movies      Public
Downloads    Music       holi
ElSalvador48:~ estudiante$ touch maquina48.txt
ElSalvador48:~ estudiante$ ls
Desktop      Library     Pictures      maquina48.txt
Documents    Movies      Public
Downloads    Music       holi
ElSalvador48:~ estudiante$ touch juan.png
ElSalvador48:~ estudiante$ touch manuel.png
ElSalvador48:~ estudiante$ touch cuellar.png
ElSalvador48:~ estudiante$
```

*Img. 04 – Captura de pantalla mostrando el comando touch. El usuario puede elegir el nombre y la extensión del archivo.*

```
ElSalvador48:~ estudiante$ touch juan.png
ElSalvador48:~ estudiante$ touch manuel.png
ElSalvador48:~ estudiante$ touch cuellar.png
ElSalvador48:~ estudiante$ ls
Desktop      Music        holi
Downloads    Movies       cuellar.png  maquina48.txt
Documents    Music        holi
Downloads    Pictures     juan.png
Library      Public       manuel.png
ElSalvador48:~ estudiante$ touch 1.gif 2.gif 3.gif 4.gif 5.gif
ElSalvador48:~ estudiante$ ls
1.gif        Desktop      Music        juan.png
2.gif        Documents    Pictures     manuel.png
3.gif        Downloads    Public       maquina48.txt
4.gif        Library      cuellar.png
5.gif        Movies       holi
ElSalvador48:~ estudiante$ mkdir tareas
ElSalvador48:~ estudiante$ ls
1.gif        Desktop      Music        juan.png
2.gif        Documents    Pictures     manuel.png
3.gif        Downloads    Public       maquina48.txt
4.gif        Library      cuellar.png  tareas
5.gif        Movies       holi
ElSalvador48:~ estudiante$
```

```
3.gif        Downloads    Public       manuel.png
4.gif        Library      clase        maquina48.txt
5.gif        Movies       cuellar.png  tareas
ElSalvador48:~ estudiante$ pwd
/Users/estudiante
ElSalvador48:~ estudiante$ cp 1.gif /Users/estudiante/tareas/
ElSalvador48:~ estudiante$ ls
1.gif        Desktop      Music        holi
2.gif        Documents    Pictures     juan.png
3.gif        Downloads    Public       manuel.png
4.gif        Library      clase        maquina48.txt
5.gif        Movies       cuellar.png  tareas
ElSalvador48:~ estudiante$ ls tareas/
1.gif
ElSalvador48:~ estudiante$ mv 1.gif 2.gif 3.gif 4.gif 5.gif /Users/estudiante/clase
ElSalvador48:~ estudiante$ ls
Desktop      Movies       clase        manuel.png
Documents    Music        cuellar.png  maquina48.txt
Downloads    Pictures     holi        tareas
Library      Public       juan.png
ElSalvador48:~ estudiante$ ls clase/
1.gif 2.gif 3.gif 4.gif 5.gif
ElSalvador48:~ estudiante$
```

Img. 05 & 06 – Capturas de pantalla mostrando una combinación de comandos para poder controlar las funciones básicas del sistema LINUX.

## Tarea

### Comandos básicos de LINUX

Para simplicidad en el formato de esta práctica y para que el lector pueda entender el documento de una mejor manera, se adjuntan capturas de pantalla de cada uno de los siguientes pasos en el documento anexo 1 al final de este reporte.

- Se abre un simulador de terminal en una página web para poder realizar los pasos desde casa sin necesidad de arriesgar nuestro equipo en casa.
- Se crea un directorio nuevo (LAB.2024.JMCO) con el comando `mkdir` y entramos a él con el comando `cd`.
- Dentro de nuestro nuevo directorio se crean cinco nuevas carpetas con el comando `mkdir`. Estas 5 carpetas representan 5 cursos en este semestre. AlgLi – Álgebra Lineal, Calln – Cálculo Integral,

DibMe – Dibujo Mecánico, Estat – Estática & Progm – Programación. Nótese como pueden crearse varios directorios con el mismo comando.

- D. Se utiliza el comando touch para generar varios archivos (JMCO.ToDo.Mtria.txt) estos documentos son generados en sus respectivos directorios, nótese que uno de los parámetros de este comando es indicar en qué directorio debe ser creado el archivo.
- E. En esta imagen comprobamos la existencia de los archivos en cada una de las carpetas que creamos. Esto con la ayuda del comando ls
- F. En esta imagen volvemos al directorio root; y esto puede ser comprobado con el comando pwd. Nótese que en este sólo existe la nueva carpeta que fue creada en el paso B.
- G. En este paso creamos un directorio nuevo (JMCO.Copy) donde trabajaremos en los siguientes pasos.
- H. En este paso se utiliza el comando cp para copiar los archivos de cada una de nuestras carpetas, es importante resaltar que se utiliza el modificador de directorio aquí para indicar donde se encuentra el archivo a copiar. Finalmente la última línea de comando le pide al sistema hace una lista (ls) de los documentos en el directorio.
- I. En este paso se mueve el directorio lleno de copias (JMCO.Copy) a nuestra carpeta de trabajo, esto gracias al comando mv.
- J. En este inciso se utiliza el modificador -l del comando ls para listar los permisos de cada uno de los componentes. Se utiliza el comando chown para cambiar el propietario del documento; sin embargo al ser este un simulador esta acción es imposible.
- K. En esta imagen solamente se muestra el uso del comando pwd para indicar al usuario la lista de directorios a seguir para poder localizarse en el sistema.
- L. En este inciso el comando cal es utilizado para mostrar un calendario del día y el mes en curso, nótese el formato que se le da al calendario
- M. Aquí se puede ver el uso del comando date, el cual nos indica la fecha, hora exacta y zona horaria de nuestro dispositivo.
- N. Finalmente en esta captura de pantalla mostramos el uso del comando man donde nos muestra una pequeña guía del comando elegido.

## Github

A continuación, la liga al repositorio de Github - [https://github.com/CuellarJM/practica1\\_fdp](https://github.com/CuellarJM/practica1_fdp)

## Conclusión

Aprender comandos básicos de Linux es importante ya que esto nos da una base sólida para poder familiarizarnos con herramientas útiles. Y esto a su vez nos permite comprender cómo funciona un sistema operativo, nos ofrece mayor control y flexibilidad.

En conclusión, conocer estos fundamentos fortalece las habilidades técnicas, abre oportunidades laborales y proporciona una base sólida para el desarrollo de software.

## Bibliografía

El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Edu

Google LLC. (2023). Refine Google searches. Google Search Help.  
<https://support.google.com/websearch/answer/2466433?hl=en>

GitHub, Inc. (2024). Hello world. GitHub Docs. <https://docs.github.com/en/get-started/start-your-journey/hello-world>

Solano, J. A., García, E. E., & Montaña, L. S. (2022, February 21). Manual de prácticas del laboratorio de Fundamentos de programación. CDMX; Facultad de Ingeniería.



```
Loading...
Welcome to Fedora 33 (riscv64)
[root@localhost ~]#
```

A

```
Loading...
Welcome to Fedora 33 (riscv64)
[root@localhost ~]# mkdir LAB.2024.JMCO
[root@localhost ~]# cd LAB.2024.JMCO
[root@localhost LAB.2024.JMCO]#
```

B

```
[root@localhost LAB.2024.JMCO]# mkdir Progm AlgLi DibMe CalIn Estat
[root@localhost LAB.2024.JMCO]# ls
AlgLi CalIn DibMe Estat Progm
[root@localhost LAB.2024.JMCO]#
```

C

```
[root@localhost LAB.2024.JMCO]# touch ./AlgLi/JMCO.ToDo.AlgLi.txt
[root@localhost LAB.2024.JMCO]# touch ./CalIn/JMCO.ToDo.CalIn.txt
[root@localhost LAB.2024.JMCO]# touch ./DibMe/JMCO.ToDo.DibMe.txt
[root@localhost LAB.2024.JMCO]# touch ./Estat/JMCO.ToDo.Estat.txt
[root@localhost LAB.2024.JMCO]# touch ./Progm/JMCO.ToDo.Progm.txt
[root@localhost LAB.2024.JMCO]#
```

D

```
[root@localhost LAB.2024.JMCO]# ls AlgLi
JMCO.ToDo.AlgLi.txt
[root@localhost LAB.2024.JMCO]# ls CalIn
JMCO.ToDo.CalIn.txt
[root@localhost LAB.2024.JMCO]# ls DibMe
JMCO.ToDo.DibMe.txt
[root@localhost LAB.2024.JMCO]# ls Estat
JMCO.ToDo.Estat.txt
[root@localhost LAB.2024.JMCO]# ls Progm
JMCO.ToDo.Progm.txt
[root@localhost LAB.2024.JMCO]#
```

E

```
[root@localhost LAB.2024.JMCO]# cd
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
bench.py hello.c LAB.2024.JMCO
[root@localhost ~]#
```

F

```
[root@localhost ~]# mkdir JMCO.Copy
[root@localhost ~]#
```

G

```
[root@localhost ~]# cd JMKO.Copy/
[root@localhost JMKO.Copy]# cp ../LAB.2024.JMKO/Algli/*.txt .
[root@localhost JMKO.Copy]# cp ../LAB.2024.JMKO/CalIn/*.txt .
[root@localhost JMKO.Copy]# cp ../LAB.2024.JMKO/DibMe/*.txt .
[root@localhost JMKO.Copy]# cp ../LAB.2024.JMKO/Estat/*.txt .
[root@localhost JMKO.Copy]# cp ../LAB.2024.JMKO/Progm/*.txt .
[root@localhost JMKO.Copy]# ls
JMKO.ToDo.Algli.txt  JMKO.ToDo.DibMe.txt  JMKO.ToDo.Progm.txt
JMKO.ToDo.CalIn.txt  JMKO.ToDo.Estat.txt
[root@localhost JMKO.Copy]#
```

H

```
[root@localhost ~]# mv JMKO.Copy ../LAB.2024.JMKO/
[root@localhost ~]# ls
bench.py  hello.c  LAB.2024.JMKO
[root@localhost ~]# ls ../LAB.2024.JMKO/
Algli  CalIn  DibMe  Estat  JMKO.Copy  Progm
[root@localhost ~]#
```

I

```
[root@localhost ~]# cd ../LAB.2024.JMKO/JMKO.Copy/
[root@localhost JMKO.Copy]# ls -l
total 0
-rw-r--r-- 1 root root 0 Feb 19 00:49 JMKO.ToDo.Algli.txt
-rw-r--r-- 1 root root 0 Feb 19 00:50 JMKO.ToDo.CalIn.txt
-rw-r--r-- 1 root root 0 Feb 19 00:50 JMKO.ToDo.DibMe.txt
-rw-r--r-- 1 root root 0 Feb 19 00:50 JMKO.ToDo.Estat.txt
-rw-r--r-- 1 root root 0 Feb 19 00:51 JMKO.ToDo.Progm.txt
[root@localhost JMKO.Copy]# touch test.jpg
[root@localhost JMKO.Copy]# chown ANON test.jpg
lschown: invalid user: 'ANON'
[root@localhost JMKO.Copy]#
```

J

```
[root@localhost JMKO.Copy]# pwd
/root/LAB.2024.JMKO/JMKO.Copy
[root@localhost JMKO.Copy]#
```

K

```
[root@localhost JMKO.Copy]# cal
February 2024
Su Mo Tu We Th Fr Sa
                1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29
[root@localhost JMKO.Copy]#
```

L

```
[root@localhost JMKO.Copy]# date
Mon Feb 19 12:54:32 AM UTC 2024
[root@localhost JMKO.Copy]#
```

M

```
CP(1)                                User Commands                                CP(1)
NAME
cp - copy files and directories

SYNOPSIS
cp [OPTION]... [-T] SOURCE DEST
cp [OPTION]... SOURCE... DIRECTORY
cp [OPTION]... -t DIRECTORY SOURCE...

DESCRIPTION
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.

Mandatory arguments to long options are mandatory for short options
too.

-a, --archive
    same as -dR --preserve=all

--attributes-only
    don't copy the file data, just the attributes

--backup[=CONTROL]
    make a backup of each existing destination file

-b
    like --backup but does not accept an argument

--copy-contents
    copy contents of special files when recursive

Manual page cp(1) line 1 (press h for help or q to quit)
```

N