



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:*

Alejandro Pimentel

*Asignatura:*

Fundamentos de Programación

*Grupo:*

Grupo 3 Bloque 135

*No de Práctica(s):*

Practica # 2

*Integrante(s):*

Muñoz Reyes Laura Vanessa - 2823

*No. de Equipo de  
cómputo empleado:*

Rumania 38

*No. de Lista o Brigada:*

No. de Cuenta: 3177522823

*Semestre:*

2020-1

*Fecha de entrega:*

26 de Julio del 2019

*Observaciones:*

Entregada por Github

Adjunto capturas de pantalla

**CALIFICACIÓN:** \_\_\_\_\_

## Practica #2: GNU/Linux

### Introducción:

En la vida cotidiana nos vemos enfrentados a diferentes sistemas operativos, sin embargo, pocas veces nos preguntamos cual sería el más funcional para nosotros o que diferente nos brinda cada uno. Es por eso que es de vital importancia para el profesional en Ingeniería conocer todas las herramientas de la computación.

### 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.

### Marco Referencial:

1. GNU/Linux es un sistema operativo libre multiplataforma, multiusuario y multitarea. El sistema es la combinación de varios proyectos, entre los cuales destacan GNU y el núcleo Linux. Su desarrollo es uno de los ejemplos más prominentes de software libre: todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera, bajo los términos de la GPL.

El software libre se refiere principalmente a cuatro libertades, protegido mediante licencias como la GPL que obliga a las versiones modificadas a mantener las mismas libertades:

- La libertad de utilizar el programa para cualquier fin, y sin restricciones.
- La libertad de estudiar cómo funciona el programa y modificarlo según necesidades o preferencias particulares.
- La libertad de compartir el programa.
- La libertad de hacer cambios al programa y distribuir la versión modificada.

El software libre implica tener un código abierto es decir:

- Colaboración
- Sin problemas de propiedad intelectual
- Sin contratos de compra de software.
- Retroalimentación.
- Desarrollo rápido y de calidad

A pesar de que «Linux» denomina en la jerga cotidiana al sistema operativo este es en realidad solo el kernel (núcleo) del sistema. El kernel es la parte del software de una computadora que no se ve, y que de hecho no debe ser vista. Es usado por los programas no por los usuarios para :

- Asigna recursos (memoria).
- Gestiona y vincula procesos.
- Comunica periféricos (drivers).
- Da acceso a los archivos.
- Gestiona el uso de red

## 2. ¿Que son los comandos?

Son las instrucciones que se pueden usar en la terminal, y generalmente tienen la siguiente estructura:

\$ **comando** -**opciones** **argumentos**

Es importante tener en cuenta los espacios que deben respetarse entre estructura y el signo \$ significa que la terminal esta esperando instrucciones

---

### Comandos Sencillos Linux

<b>\$date</b>	Manejo de fechas
<b>\$whoami</b>	Muestra nombre de usuario en el que estamos
<b>\$pwd</b>	Saber en qué parte estas situado
<b>\$cd</b>	Navegación entre directorios
<b>\$cd..</b>	Sube una carpeta
<b>\$touch</b>	Crear archivo nuevo
<b>\$rm</b>	Elimina o remueve archivo
<b>\$cp</b>	Copia la fuente al objetivo
<b>\$mv</b>	Mueve la fuente al objetivo removiendola
<b>\$mkdir</b>	Crear un nuevo directorio o carpeta
<b>\$rmdir</b>	Eliminar nueva carpeta
<b>\$ls</b>	Lista de archivos y carpetas del directorio actual
<b>\$ls - l</b>	Lista detallada
<b>\$ls - a</b>	Lista con archivos ocultos y no ocultos
<b>\$ls - al</b>	Lista completa de archivos y detallada
<b>\$man (comando)</b>	Muestra manual del comando
<b>\$(comando) --help</b>	Descripción del comando
<b>\$echo</b>	Impresión texto en pantalla
<b>\$cat</b>	Leer datos de archivo y mostrarlos
<b>\$more</b>	Ver el contenido de un archivo por partes o paginas
<b>\$less</b>	Muestra el archivo y permite navegación en el
<b>\$grep felicidad</b>	Imprime solo las líneas que contienen felicidad

---

Frecuentemente un programa necesitará obtener información desde, por ejemplo, un teclado, o bien enviará la información a un destino, por ejemplo a una pantalla. La comunicación entre el origen de cierta información y el destino se realiza mediante un flujo de información

### Entradas:

Estándar: Es el texto que reciben algunos comandos para procesarlos y después dar una salida.

Argumentos o parámetros: Pueden ser archivos, rutas, variables, entre otros; y el comando los requiere antes de correr.

**Salidas:**

Siempre es estándar, y es lo que el programa devuelve después de procesar lo que se le pidió.

**Re direccionamiento:**

Lo que vas a hacer es redirigir lo que se muestra por pantalla a otro sitio. Pero no solo esto, sino que también, en lugar de tomar lo que entra directamente por el teclado, vas a tomar lo que proviene de otro archivo.

Por ejemplo, La ejecución de algunos comandos en el terminal da como resultado un texto. Por ejemplo, al ejecutar `ls`, el resultado es un texto que contiene el listado de archivos y directorios. Normalmente esto lo se muestra por pantalla, es decir, que la salida se dirige a la pantalla. Sin embargo, ¿qué sucede si quieres guardar este listado? La solución es redirigir la salida del comando, en lugar de a la pantalla a un archivo, y de esta manera guardarlo.

Hacia archivo ( `>` y `>>` )

`date > ahora.txt`

Reemplaza el contenido del archivo por la salida

`$ date >> ahora.txt`

**Procedimiento:**

1. Prueba los comandos sencillos ( `$date`, `$whoami`, `$pwd` )
2. Navega entre carpetas con `$cd`
3. Realiza operaciones con archivos ( `$touch`, `$rm`, `$cp`, `$mv`, `$mkdir` )
4. Realiza una lista de archivos
5. Busca el manual del comando `$ls`
6. Busca los detalles del comando `$ls`
7. Examina el archivo `Libro.txt` e interactúa con los comandos `$echo`, `$cat`, `$more`, `$less`, `$grep`

**Resultados:**

Fragmento de lo realizado en el laboratorio

Last login: Mon Aug 19 09:06:13 on console

[Rumania38:~ alumno\$ date

Mon Aug 19 09:19:29 CDT 2019

[Rumania38:~ alumno\$ whoami

alumno

[Rumania38:~ alumno\$ bwd

-bash: bwd: command not found

[Rumania38:~ alumno\$ pwd

/Users/alumno

[Rumania38:~ alumno\$ cd USER

-bash: cd: USER: No such file or directory

[Rumania38:~ alumno\$ cd /

[Rumania38:/ alumno\$

[Rumania38:/ alumno\$ cd/

-bash: cd/: No such file or directory

[Rumania38:/ alumno\$

[Rumania38:/ alumno\$ cd/

-bash: cd/: No such file or directory

[Rumania38:/ alumno\$ cd /

.DS\_Store

.DocumentRevisions-V100/

.PKInstallSandboxManager/

.PKInstallSandboxManager-SystemSoftware/

.Spotlight-V100/

.file

.fsevents/

.vol/

Applications/

Insight Files/

LC\_AB/

Library/

Network/

System/

Users/

Volumes/

anaconda3/

bin/

cores/

dev/

etc/

home/

installer.failurerequests

[net/

[opt/

[private/

[sbin/

[tmp/

[usr/

[var/

[Rumania38:/ alumno\$ cd / Users

[Rumania38:/ alumno\$ pwd

/

[Rumania38:/ alumno\$ cd /Users

[Rumania38:Users alumno\$ pdw

-bash: pdw: command not found

[Rumania38:Users alumno\$ pwd

/Users

[Rumania38:Users alumno\$ cd

Shared/ administrador/ alumno/ curso/

[Rumania38:Users alumno\$ cd shared/

[Rumania38:shared alumno\$ pwd

/Users/shared

[Rumania38:shared alumno\$ cd..

-bash: cd.: command not found

[Rumania38:shared alumno\$ cd ..

[Rumania38:Users alumno\$ cd ..

[Rumania38:/ alumno\$ cd ..

```

[alumno] alumno — -bash — 80x67
/Users/shared
[Rumania38:shared alumno$ cd..
-bash: cd.: command not found
[Rumania38:shared alumno$ cd ..
[Rumania38:Users alumno$ cd ..
[Rumania38:/ alumno$ cd ..
[Rumania38:/ alumno$ /Users
-bash: /Users: is a directory
[Rumania38:/ alumno$ cd
.DS_Store
.DocumentRevisions-V100/
.PKInstallSandboxManager/
.PKInstallSandboxManager-SystemSoftware/
.Spotlight-V100/
.file
.fsevents/
.vol/
Applications/
Insight Files/
LC_AB/
Library/
Network/
System/
Users/
Volumes/
anaconda3/
bin/
cores/
dev/
etc/
home/
installer.failurerequests
[net/
[Rumania38:/ alumno$ cd
[Rumania38:~ alumno$ pwd
/Users/alumno
[Rumania38:~ alumno$ touch archivo1.doc
[Rumania38:~ alumno$ touch archivo5.txt
[Rumania38:~ alumno$ rm archivo1.doc
[Rumania38:~ alumno$ cp archivo5.txt archivo1.doc
[Rumania38:~ alumno$ mv archivo5.txt archivoe.txt
[Rumania38:~ alumno$ mkdir carpeta
[Rumania38:~ alumno$ ls
Desktop      Library      archivo1.doc  carpeta
Downloads    Pictures     archivoe.txt
[Rumania38:~ alumno$ ls -al
total 0
drwxr-xr-x+ 11 alumno  staff   352 Aug 19 09:55 .
drwxr-xr-x   6 root    admin   192 Jul 31 18:23 ..
drwx-----  2 alumno  staff    64 Aug 19 09:06 .Trash
drwx-----  3 alumno  staff    96 Aug 19 09:15 .bash_sessions
drwx-----  2 alumno  staff    64 Aug 19 09:06 Desktop
drwx-----  2 alumno  staff    64 Aug 19 09:06 Downloads
drwx-----@ 35 alumno  staff  1120 Aug 19 09:36 Library
drwx-----+  3 alumno  staff    96 Aug 13 10:53 Pictures
-rw-r--r--   1 alumno  staff     0 Aug 19 09:51 archivo1.doc
-rw-r--r--   1 alumno  staff     0 Aug 19 09:47 archivoe.txt
drwxr-xr-x   2 alumno  staff    64 Aug 19 09:55 carpeta
[Rumania38:~ alumno$ cp archivo5.txt carpeta/nombrecopiado.txt
cp: archivo5.txt: No such file or directory
[Rumania38:~ alumno$ cp archivoe.txt/nombrecopiado.txt
usage: cp [-R [-H | -L | -P]] [-fi | -n] [-apvXc] source_file target_file
        cp [-R [-H | -L | -P]] [-fi | -n] [-apvXc] source_file ... target_directory
[Rumania38:~ alumno$ mv archivo1.doc carpeta
[Rumania38:~ alumno$ cp archivoe.txt carpeta/nombrecambiado.txt
[Rumania38:~ alumno$ mv archivo1.doc carpeta

```

### **Conclusiones:**

Con esta práctica entendí mucho mejor los comandos y su estructura. Me llamo mucho la atención como desde la terminal se pueden hacer tantas operaciones como desde crear un archivo, modificarlo, moverlo entre carpetas hasta incluso leerlo en la misma terminal.

Comprendí la importancia de identificarte a la computadora con tu nombre de usuario. Y de escribir bien los comandos con el espacio donde debe ir y las letras tal cual.

Intentando hacerlo en mi casa entendí mucho más pero además descubrí que Windows no es tan práctico como Linux y no permite el ingreso de algunos comandos.

De igual manera me llamo mucho la atención el comando \$ls pues te muestra todas las carpetas a detalle.

En conclusión me gustó mucho esta práctica, le entendí a la terminal más y tuve menos errores.

### **Bibliografía:**

[https://drive.google.com/drive/folders/1FBjWL\\_WQnh8pDKR3t9uiWydUbs815y9s](https://drive.google.com/drive/folders/1FBjWL_WQnh8pDKR3t9uiWydUbs815y9s)