



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación  
Salas A y B

*Profesor:* Alejandro Pimentel

*Asignatura:* Laboratorio de programación

*Grupo:* 135

*No de Práctica(s):* Práctica 2

*Integrante(s):* Pimentel Maldonado Adriana Itzel

*No. de Equipo de cómputo  
empleado:* 13

*No. de Lista o Brigada:* 42

*Semestre:* Primer semestre

*Fecha de entrega:* 25/agosto/2019

*Observaciones:*

CALIFICACIÓN: \_\_\_\_\_

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

## **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. 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 x Administración de E/S.

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

## ***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

## ***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.

Además de conocer de otros sistemas operativos, continuamos con el uso de Linux.

```
MINGW64/c/Users/Itzel Pm
Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ comando -opciones argumentos
bash: comando: command not found

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ date
do., 25 de ago. de 2019 23:53:30

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ whoami
Itzel Pm

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ pwd
/c/Users/Itzel Pm

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ cd /home/
bash: cd: /home/: No such file or directory

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ cd Users
bash: cd: Users: No such file or directory

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ cd USER
bash: cd: USER: No such file or directory

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ cd ..

Itzel PmLAPTOP-888FAJ3D MINGW64 /c/Users
$ cd

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ touch archivol.txt

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ touch archivov2.txt

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ rm archivol.txt

Itzel PmLAPTOP-888FAJ3D MINGW64 ~ (master)
$ cp archivol.txt ...
```

```
MINGW64/c/Users/Itzel Pm
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ comando -opciones argumentos
bash: comando: command not found

Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ date
do., 25 de ago. de 2019 23:53:30

Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ whoami
Itzel Pm
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ pwd
/c/Users/Itzel Pm
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd /home/
bash: cd: /home/: No such file or directory
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd Users
bash: cd: Users: No such file or directory
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd USER
bash: cd: USER: No such file or directory
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd ..
Itzel Pm@LAPTOP-888FAJJD MINGW64 /c/Users
$ cd
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ touch archivol.txt
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ touch archivo2.txt
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ rm archivol.txt
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cp archivol.txt ..
```

```
MINGW64/c/Users/Itzel Pm
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ comando -opciones argumentos
bash: comando: command not found

Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ date
do., 25 de ago. de 2019 23:53:30

Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ whoami
Itzel Pm
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ pwd
/c/Users/Itzel Pm
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd /home/
bash: cd: /home/: No such file or directory
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd Users
bash: cd: Users: No such file or directory
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd USER
bash: cd: USER: No such file or directory
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cd ..
Itzel Pm@LAPTOP-888FAJJD MINGW64 /c/Users
$ cd
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ touch archivol.txt
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ touch archivo2.txt
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ rm archivol.txt
Itzel Pm@LAPTOP-888FAJJD MINGW64 ~ (master)
$ cp archivol.txt ..
```

```
MINGW64/c/Users/Itzel Pm
Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ comando -opciones argumentos
bash: comando: command not found

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ date
do., 25 de ago. de 2019 23:53:30

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$
Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ whoami
Itzel Pm

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ pwd
/c/Users/Itzel Pm

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ cd /home/
bash: cd: /home/: No such file or directory

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ cd Users
bash: cd: Users: No such file or directory

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ cd USER
bash: cd: USER: No such file or directory

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ cd ..

Itzel Pm@LAPTOP-888AFAJD MINGW64 /c/Users
$ cd

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ touch archivo1.txt

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ touch archivo2.txt

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ rm archivo1.txt

Itzel Pm@LAPTOP-888AFAJD MINGW64 ~ (master)
$ cp archivo1.txt ..
```

```
MINGW64/c/Users/Itzel Pm
--indicator-style=WORD (default if omitted), 'auto', or 'never'
                        append indicator with style WORD to entry names:
                        none (default), slash (-p),
                        file-type (--file-type), classify (-F)
-l, --inode             print the index number of each file
-I, --ignore-ePattern  do not list implied entries matching shell PATTERN
-k, --kibibytes         default to 1024-byte blocks for disk usage;
                        used only with -s and per directory totals
-l                     use a long listing format
-L, --dereference       when showing file information for a symbolic
                        link, show information for the file the link
                        references rather than for the link itself
-m                     fill width with a comma separated list of entries
                        like -l, but list numeric user and group IDs
-N, --literal           print entry names without quoting
-o                     like -l, but do not list group information
-p, --indicator-style=slash
                        append / indicator to directories
-q, --hide-control-chars
                        print ? instead of nongraphic characters
--show-control-chars   show nongraphic characters as-is (the default,
                        unless program is 'ls' and output is a terminal)
-Q, --quote-name        enclose entry names in double quotes
--quoting-style=WORD   use quoting style WORD for entry names:
                        literal, locale, shell, shell-always,
                        shell-escape, shell-escape-always, c, escape
                        (overrides QUOTING_STYLE environment variable)
-r, --reverse           reverse order while sorting
-R, --recursive         list subdirectories recursively
-s, --size              print the allocated size of each file, in blocks
-S, --sort=WORD         sort by file size, largest first
                        sort by WORD instead of name: none (-U), size (-S),
                        time (-t), version (-V), extension (-X)
--time=WORD            with -l, show time as WORD instead of default
                        modification time: atime or access or use (-u);
                        ctime or status (-o); also use specified time
                        as sort key if --sort=time (newest first)
--time-style=TIME_STYLE
                        time/date format with -l; see TIME_STYLE below
-t                     sort by modification time, newest first
-T, --tabsize=COLS     assume tab stops at each COLS instead of 8
-u                     with -lt: sort by, and show, access time;
                        with -li: show access time and sort by name;
                        otherwise: sort by access time, newest first
-U                     do not sort; list entries in directory order
-v                     natural sort of (version) numbers within text
-w, --width=COLS       set output width to COLS. 0 means no limit
-x                     list entries by lines instead of by columns
-X                     sort alphabetically by entry extension
-Z, --context           print any security context of each file
-l                     list one file per line. Avoid '\n' with -q or -b
--append-exe           append .exe if cygwin magic was needed
```

## CONCLUSIONES:

Git como controlador de versiones es una herramienta muy útil, y la importancia de conocer sus comandos para que tengamos un mejor uso del mismo es imprescindible. Conocer además de otros sistemas operativos y sus terminales amplia nuestros conocimientos y nos permite elegir la opción que nos ayudaría a realizar mejores entregas para nuestros trabajos y en vida laboral.

Hay tanto diferencias como similitudes entre los sistemas operativos Windows, Mac y Linux, pero al ser Linux disponible para todos de manera gratuita, hace que

manejarlo no implique ningún tipo de gasto, y su uso simple facilita el entendimiento del programa