

# Sistemas Operativos

## Práctica 1. Introducción al sistema operativo Linux y Windows (1)

Prof. Jorge Cortés Galicia

### Competencias.

El alumno aprende a familiarizarse con los sistemas operativos Linux y Windows mediante la exploración de sus ambientes de trabajo para desarrollar aplicaciones en lenguaje C.

El alumno analiza el sistema operativo Linux y Windows mediante la comparación de sus características principales para diferenciarlos en su ambiente de trabajo.

### Desarrollo.

#### Sección Linux:

1. Verifique la existencia del sistema operativo Linux y su correcto funcionamiento con el entorno gráfico (si el entorno gráfico no funciona inicie sesión en modo texto).
2. Reporte la distribución que usted está utilizando. Mencione que otras distribuciones de Linux existen y cuáles son las diferencias respecto a la distribución que usted está utilizando.
3. Revise el entorno Linux e identifique los componentes de menú con los que cuenta su sistema operativo. Entre en cada uno de los componentes del menú y explore algunas de sus funcionalidades. Reporte que funcionalidades encontró, además haga un comparativo de estas funcionalidades con respecto a las existentes en el sistema operativo Windows que usted utilice.
4. Abra una sesión de consola (Terminal) y reporte para que se utilizan cada uno de los siguientes comandos:

<b>ls</b>	<b>cd</b>	<b>cp</b>
<b>chmod</b>	<b>cat</b>	<b>mv</b>
<b>vi</b>	<b>grep</b>	<b>mkdir</b>
<b>pwd</b>	<b>rm</b>	<b>rmdir</b>
<b>clear</b>	<b>ps</b>	<b>whoami</b>

para ello utilice el manual en línea (**man**) con el que cuenta linux. La forma de usarlo es tecleando el siguiente comando en la Terminal:

**man nombre\_del\_comando\_a\_buscar, por ejemplo: man ls, man ps.**

5. Pruebe desde la terminal los siguientes comandos (tenga cuidado al ejecutar los comandos de borrado de archivos y directorios):

<b>ls</b>	<b>cd</b> nombre_directorio	<b>cp</b> [ruta1]archivo1 [ruta2][archivo 2]
<b>ls -l</b>	<b>cat</b> nombre_archivo	<b>mv</b> [ruta1]archivo1 [ruta2][archivo 2]
<b>ls -la</b>	<b>ls -la   more</b>	<b>mkdir</b> nombre_directorio
<b>pwd</b>	<b>rm</b> nombre_archivo	<b>rmdir</b> nombre_directorio
<b>clear</b>	<b>Ps</b>	<b>whoami</b>

Observe los resultados de la ejecución de cada comando.

- Algunos de los comandos anteriores pueden tener una serie de opciones para su ejecución (p.e. ls), utilice nuevamente el comando **man nombre\_comando** para ver detalles de información sobre cada uno de los comandos anteriores. Observe las diversas opciones que se utilizan en dichos comandos.
- Abra el editor de texto de su preferencia y realice un programa en lenguaje C que imprima en pantalla “Hola mundo”. Compile y ejecute su programa, para ello utilice el comando siguiente en la Tarminal:  
**gcc nombre\_programa.c -o nombre\_salida** para compilar, y  
**./nombre\_salida** para ejecutar el programa.
- Realice un programa que muestre la siguiente salida:

```

      *      *      *
     **     ***     **
    ***    *****    ***
 *****  *****  *****
    ***    *****    ***
     **     ***     **
      *      *      *

```

El dato de entrada al programa es el número de asteriscos de la línea horizontal media en la figura con la mayor cantidad de ellos (es decir la figura del rombo), en este caso el dato de entrada es 7 (línea horizontal de asteriscos en rojo). Este dato se pedirá al usuario y podrá ser cualquier valor dentro de un rango que usted establezca, el cual permita construir correctamente la figura completa en la pantalla de la computadora. A partir de este único dato, se construirá la figura con la restricción de que sólo deberá imprimirse un asterisco a la vez. Adicionalmente, dé la opción de guardar la figura en un archivo. Utilice solo bibliotecas estándares de ANSI C. Compile y ejecute su programa.

- Realice un programa que calcule la serie de Fibonacci de un número dado como entrada. Compile y ejecute su programa.
- Realice un programa que determine si un grupo de paréntesis están balanceados utilizando pilas. Deberán considerarse tres tipos de paréntesis: ( ), { }, [ ]. Compile y ejecute su programa.
- Realice un programa que evalúe una expresión aritmética delimitada completamente con paréntesis balanceados (por ejemplo: (((5-2)+4)\*10). Compile y ejecute su programa. Restricción: No utilizar transformación a notación prefija o posfija.
- Guarde sus programas (fuente y ejecutable) en memoria usb.

### Sección Windows:

1. Inicie sesión en Windows.
2. Abra una consola.
3. Investigue para que sirve cada uno de los siguientes comandos y ejecútelos en la consola (tenga cuidado al ejecutar los comandos de borrado de archivos y directorios).

<b>Dir</b>	<b>Cd</b> nombre_directorio	<b>Copy</b> nombre_archivo nuevo_nombre_archivo
<b>ipconfig</b>	<b>Type</b> nombre_archivo	<b>Ren</b> nombre_archivo nuevo_nombre_archivo
<b>Cls</b>	<b>Mkdir</b> nombre_directorio	<b>Chdir</b> nombre_directorio
<b>Ver</b>	<b>rmdir</b> nombre_directorio	<b>echo</b> "Hola mundo"
<b>Tree</b>	<b>del</b> nombre_archivo	<b>Find</b> "cadena_buscar" nombre_archivo

4. Reporte la función de cada uno de los anteriores comandos.
5. Abra la carpeta de su memoria usb donde tiene los programas que desarrolló en la sección de Linux y ejecútelos. Dé sus observaciones sobre la ejecución de los mismos.
6. Busque el directorio donde está instalado Dev C.
7. Desde la consola ubíquese en el directorio de Dev C, y cambie al directorio "bin"
8. Desde ese directorio compile en la consola como se indica en el punto 10 de la sección Linux cada uno de los programas creados en esa sección y que guardó en la memoria usb, ejecútelos y observe su funcionamiento.
9. Reporte las diferencias y similitudes entre los comandos de Linux y Windows, así como la compilación y ejecución de los programas realizados.