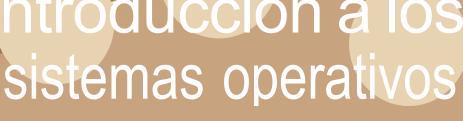


INSTITUTO POLITECTO NACIONAL ESCUELA SUPERIOR DE CÓMPUTO



Preties 1 Introducción a los











Profesor: Cortes Galicia Jorge

BEAUTIFUI

INTRODUCCION

Un sistema operativo (SO) es un conjunto de programas esenciales que gestionan la interacción entre el usuario y el hardware de una computadora, así como la ejecución de aplicaciones. En esencia, actúa como un intermediario vital entre ambos, simplificando el manejo y uso del sistema informático en su totalidad. Formando parte integral de todo sistema informático, el SO coordina los recursos de hardware, como la CPU, la memoria y los dispositivos de entrada/salida, para asegurar un funcionamiento eficiente.

Los SO se dividen en diversas categorías, según su capacidad y funcionalidad. Entre estas, destacan los sistemas monousuarios, que admiten un solo usuario a la vez, y los multiusuarios, capaces de servir a múltiples usuarios simultáneamente. Asimismo, se distinguen los sistemas monotarea, que permiten una sola tarea por usuario a la vez, de los multitareas, que posibilitan la realización de múltiples actividades simultáneas por parte del usuario. Además, existen sistemas operativos diseñados para gestionar un solo procesador (uniproceso) y otros para manejar varios procesadores simultáneamente (multiproceso).

Dentro del panorama de los SO más utilizados, destacan dos gigantes: Windows y Linux. Windows, desarrollado por Microsoft Corporation, es un sistema operativo gráfico que ha alcanzado una amplia popularidad debido a su versatilidad y facilidad de uso. Con soporte para múltiples usuarios y multitareas, Windows ofrece una amplia gama de funciones que incluyen desde la gestión de archivos hasta la conexión a Internet, adaptándose a diversas plataformas, como PC, dispositivos móviles y servidores.

Por otro lado, Linux es un sistema operativo de código abierto, cuyo desarrollo se basa en principios como la transparencia, la colaboración y la meritocracia inclusiva. Este sistema ofrece una amplia variedad de distribuciones, conocidas como "distros", que se adaptan a las necesidades de diferentes usuarios. Al igual que Windows, Linux es multiusuario y multitarea, y puede ejecutarse en una variedad de plataformas, desde computadoras de escritorio y portátiles hasta servidores y dispositivos móviles.

Ambos sistemas operativos tienen una presencia significativa en el mercado y ofrecen distintas opciones para satisfacer las necesidades de los usuarios en términos de funcionalidad, rendimiento y flexibilidad. Sin embargo, cada uno tiene sus propias características distintivas y comunidades de usuarios dedicadas que contribuyen a su desarrollo y evolución continua.



SECCIÓN LINUX

1.- 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.

Distribución de Linux Utilizada: Ubuntu

La distribución de Linux utilizada en este entorno es Ubuntu. La versión específica en uso es Ubuntu 22.04.4 LTS (Long Term Support). Ubuntu es una distribución de Linux basada en Debian y es conocida por su facilidad de uso, estabilidad y amplio soporte de la comunidad.

Otras distribuciones de Linux

Existen varias distribuciones de Linux disponibles, cada una con sus propias características y enfoques. Algunas de las distribuciones más conocidas incluyen:

FEDOMA: Es una distribución comunitaria respaldada por Red Hat y se centra en la innovación y la adopción temprana de nuevas tecnologías. Es especialmente popular entre los desarrolladores y usuarios avanzados.

DEBIAM: Es una distribución conocida por su estabilidad, robustez y su amplio conjunto de paquetes. Muchas otras distribuciones, incluyendo Ubuntu, se basan en Debian.

CENTOS/MHEL: Estas distribuciones son populares en entornos empresariales debido a su estabilidad y soporte a largo plazo. CentOS es una versión gratuita de Red Hat Enterprise Linux (RHEL).

ANCH LIMUX: Es una distribución minimalista y altamente personalizable dirigida a usuarios avanzados que prefieren un mayor control sobre su sistema.

CPENGUSE: Es conocida por su facilidad de uso y herramientas de administración del sistema, como YaST. Se adapta bien tanto a usuarios principiantes como a administradores de sistemas experimentados.

Diferencias Respecto a Otras Distribuciones

Ubuntu se destaca por su enfoque en la facilidad de uso, su amplia disponibilidad de software y su comunidad de usuarios activa. A diferencia de algunas otras distribuciones, Ubuntu ofrece versiones LTS (Long Term Support) con actualizaciones de seguridad y soporte extendido a largo plazo, lo que la hace especialmente adecuada para entornos de producción y empresas que valoran la estabilidad y la fiabilidad del sistema operativo.



Tabla comparativa de Ubuntu con respecto a otras distribuciones

| Caracteris tica | Ubuntu | Fedora | Debian | CentOS/R HEL | Arch Linux | openSUSE |
|-----------------------------|---|--|------------------------------|---|---|---|
| Base | Debian | Independie nte | Debian | RHEL | Independie nte | SUSE |
| Enfoque | Facilida d de uso, estabilid ad | Innovación, adopción temprana de tecnologías | Estabilid ad, robustez | Estabilidad , soporte a largo plazo | Minimalism O, personaliza Ción | Facilidad de uso, herramient as de administra ción |
| Gestor de Paquetes | APT (dpkg) | DNF (anteriorme nte yum) | APT (dpkg) | YUM/DNF (RHEL) | Pacman | Zypper |
| Ciclo de Lanzamien to | Regular (con versione s LTS) | Regular | Regular | Largo plazo (con versiones CentOS) | Rolling Release | Regular |
| Soporte | Comunid ad y Canonic al | Comunidad y Red Hat | Comunid ad | Red Hat | Comunidad | Comunidad y SUSE |
| Versiones LTS | Sí | No | No | Sí | No | No |
| Filosofía | Comunid ad y comerci al | Comunidad | Comunid ad | Comercial | Comunidad | Comunidad |
| Uso Principal | Escritori o y servidor | Escritorio y servidor | Escritori o y servidor | Servidor | Avanzado, servidores | Escritorio y servidor |

2.- 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.

Funcionalidades encontradas en Linux



PANE SUPERIOR: Proporciona acceso rápido a aplicaciones y herramientas importantes, así como notificaciones del sistema y la red.

ESCRITORIOS MÚLTIPLES: Permite trabajar en varios escritorios virtuales al mismo tiempo, lo que facilita la organización y la gestión de tareas.

CONFIGURACIÓN DEL SISTEMA: Ofrece una amplia gama de opciones de configuración para personalizar el entorno de Ubuntu según las preferencias del usuario, incluyendo ajustes de apariencia, hardware y sistema.

CARPETA PERSONAL: Almacena los archivos y datos del usuario, proporcionando un espacio de trabajo personalizado y seguro.

MULTI-TAMEA: Permite ejecutar varias aplicaciones simultáneamente y cambiar entre ellas fácilmente, lo que mejora la productividad del usuario.

COMEXIÓN A LA MED: Ubuntu está diseñado para funcionar con conexión a internet, lo que permite acceder a recursos en línea, actualizaciones de software y servicios en la nube.

CENTRO DE SOFTWARE: Proporciona acceso a una amplia variedad de aplicaciones, juegos, multimedia y otros recursos para facilitar la instalación y gestión de software.

HERRAMIENTAS DE OFFICE: Incluye suites de oficina como LibreOffice para la creación y edición de documentos, hojas de cálculo y presentaciones.

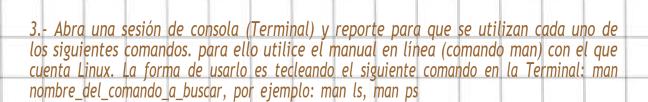
ACCESO A FUNCIONES AVANZADAS: La terminal proporciona acceso a funciones avanzadas y configuraciones que no siempre están disponibles a través de la interfaz gráfica de usuario. Esto permite a los usuarios realizar tareas específicas de manera más eficiente y personalizar su experiencia según sus necesidades.

Comparativo de funcionalidades encontradas entre Linux (Ubuntu) y Windows

| Funcionalidad | Ubuntu | Windows 10 / 11 | Imagen Ubuntu | Imagen Windows |
|---------------|--|--|---------------|-------------------|
| Inicio | Acceso rápido a aplicaciones, navegador, archivos, ayuda, almacenamiento, papelera, menú de aplicaciones | aplicaciones, documentos recientes, configuraciones, opciones de | | |



| X | | | | | | H |
|---|-------------------------|---|--|--|--|---|
| | Funcionalidad | Ubuntu | Windows 10 / 11 | Imagen Ubuntu | Imagen Windows | |
| | Barra de tareas | Barra de aplicaciones recientes y más utilizadas | Permite anclar aplicaciones, acceso a vista de tareas, ver aplicaciones abiertas, búsqueda integrada | | Windows 11 Or wante to the control of the control | |
| | Configuración rápida | Configuración centralizada, redes, estado de carga, gestión de recursos, configuración general, bloqueo y apagado | Configuración del sistema, configuración de dispositivos, cuentas de usuario, redes, etc. | Capleado conectado Cargado completamente > Cargado com | TOTAL COLLEGE TOTAL | _ |
| | Aplicaciones | Calculadora, editor de texto, System Monitor, terminal | Explorador de archivos, Editor de fotos, Reproductor de música, Centro de seguridad | | C tour relation and an experience of the control of | |



| ls: Lista la información |
|----------------------------|
| acerca de los archivos (la |
| opción por defecto es el |
| directorio actual). Ordena |
| alfabéticamente si no se |
| especifica ningún otro |
| parámetro. |
| • |

cd: Cambiar de directorio de trabajo.

cp: Copiar archivos odirectorios.



chmod: Cambia el modo de archivos y directorios, puede otorgar o revocar permisos de lectura, escritura y ejecución, para el propietario del archivo, el grupo al que pertenece y otros usuarios.

cat: Concatena archivos y los imprime en la misma terminal.

mv: *M*over (renombrar) archivos.

vi: Abre Vim que es un editor de texto para programadores.

grep: Busca patrones en cada archivo e imprime cada línea que coincide con el patrón.

mkdir: Crear directorios.

pwd: Imprime en pantalla la ruta completa del directorio sobre el que se está trabajando actualmente.

rm: Borrar archivos o directorios.

rmdir: Elimina directorios vaciones, aquellos que no contienen archivos.

clear: Limpia la pantalla de los comandos que se ejecutaron anteriormente.

ps: Muestra la información de una selección de procesos activos.

whoami: Imprimir el nombre del usuario relacionado con el ID del usuario activo.

4.- Ejecute desde la terminal los siguientes comandos (tenga cuidado al ejecutar los comandos de borrado de archivos y directorios):

artur@eulaubuntu:~\$ ls

Descargas Documentos Imágenes Plantillas snap Desktop Escritorio Música Público Vídeos





```
lartur@eulaubuntu:~$ ls -l
total 40
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Descargas
drwxr-xr-x 2 artur artur 4096 feb 10 23:03 Desktop
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Documentos
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Escritorio
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Imágenes
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Música
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Plantillas
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Público
drwxr-xr-x 2 artur artur 4096 feb 27 07:56 snap
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Vídeos
```

```
artur@eulaubuntu:~$ ls -la
total 108
drwxr-x--- 18 artur artur
                          4096 feb 27 22:23
                          4096 feb 6 19:08
drwxr-xr-x 3 root root
                          3497 feb 27 09:25 .bash_history
-rw----- 1 artur artur
                                  6 19:08 .bash logout
-rw-r--r-- 1 artur artur
-rw-r--r-- 1 artur artur
                          3771 feb 6 19:08 .bashrc
drwx----- 12 artur artur
                          4096 feb 24 13:05 .cache
drwx----- 14 artur artur 4096 feb 27 08:54 .config
                        4096 feb 27 08:54 Descargas
drwxr-xr-x 2 artur artur
drwxr-xr-x 2 artur artur 4096 feb 10 23:03 Desktop
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:54 Documentos
drwx----- 3 artur artur
                          4096 feb 10 22:56 .emacs.d
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Escritorio
drwx----- 2 artur artur
                        4096 feb 27 08:45 .gnupg
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:54 Imágenes
-rw----- 1 artur artur
                            65 feb 27 09:14 .lesshst
drwx----- 3 artur artur 4096 feb 6 19:45 .local
drwxr-xr-x 2 artur artur
                        4096 feb 27 08:54 Música
                        358 feb 27 08:53 .pam_environment
-rw-r--r-- 1 artur artur
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Plantillas
-rw-r--r-- 1 artur artur
                         807 feb 6 19:08 .profile
                        4096 feb 27 08:54 Público
drwxr-xr-x 2 artur artur
drwx----- 4 artur artur
                          4096 feb 27 07:56 snap
drwx----- 2 artur artur
                          4096 feb
                                   6 20:57 .ssh
                                   7 16:42 .sudo as admin successful
-rw-r--r-- 1 artur artur
                             0 feb
-rw----- 1 artur artur 12288 feb
                                  7 21:36 .swp
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:54 Videos
```

artur@eulaubuntu:~\$ pwd /home/artur

artur@eulaubuntu:~\$ clear



artur@eulaubuntu:~\$

artur@eulaubuntu:~\$ cd Escritorio/
artur@eulaubuntu:~/Escritorio\$ pwd
/home/artur/Escritorio

artur@eulaubuntu:~/Escritorio\$ ls
test.txt
artur@eulaubuntu:~/Escritorio\$ cat test.txt
Hola, este archivo es una prueba

```
artur@eulaubuntu:~$ ls -la|more
+total 108
drwxr-x--- 18 artur artur 4096 feb 27 22:23 .
drwxr-xr-x 3 root root
                          4096 feb 6 19:08 ...
           1 artur artur
                          3497 feb 27 09:25 .bash history
           1 artur artur
                          220 feb 6 19:08 .bash_logout
-rw-r--r-- 1 artur artur 3771 feb 6 19:08 .bashrc
drwx----- 12 artur artur 4096 feb 24 13:05 .cache
drwx----- 14 artur artur 4096 feb 27 08:54 .config
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Descargas
drwxr-xr-x 2 artur artur
                         4096 feb 10 23:03 Desktop
drwxr-xr-x 2 artur artur 4096 feb 27 08:54 Documentos
drwx----- 3 artur artur
                         4096 feb 10 22:56 .emacs.d
                          4096 feb 27 22:31 Escritorio
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:45 .gnupg
drwx----- 2 artur artur
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:54 Imágenes
-rw----- 1 artur artur
                           65 feb 27 09:14 .lesshst
drwx----- 3 artur artur
                          4096 feb 6 19:45 .local
                          4096 feb 27 08:54 Música
drwxr-xr-x 2 artur artur
           1 artur artur
                          358 feb 27 08:53 .pam environment
- FW- F-- F--
                          4096 feb 27 08:54 Plantillas
drwxr-xr-x 2 artur artur
-rw-r--r-- 1 artur artur
                         807 feb 6 19:08 .profile
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:54 Público
                          4096 feb 27 07:56 snap
drwx----- 4 artur artur
drwx-----
           2 artur artur
                          4096 feb 6 20:57 .ssh
                                   7 16:42 .sudo as admin successful
                             0 feb
-rw-r--r-- 1 artur artur
           1 artur artur 12288 feb
                                   7 21:36 .swp
drwxr-xr-x 2 artur artur
                          4096 feb 27 08:54 Vídeos
```

artur@eulaubuntu:~/Escritorio\$ ls

test1.txt test2.txt

artur@eulaubuntu:~/Escritorio\$ rm test1.txt

artur@eulaubuntu:~/Escritorio\$ ls

test2.txt



artur@eulaubuntu:~/Escritorio\$ ps

PID TTY TIME CMD 1979 pts/0 00:00:00 bash

2482 pts/0 00:00:00 ps

artur@eulaubuntu:~/Escritorio\$ ls

test2.txt

artur@eulaubuntu:~/Escritorio\$ cp test2.txt test3.txt

artur@eulaubuntu:~/Escritorio\$ ls

test2.txt test3.txt

artur@eulaubuntu:~/Escritorio\$ mv test3.txt ../Vídeos/

artur@eulaubuntu:~/Escritorio\$ cd ../Videos/

artur@eulaubuntu:~/Vídeos\$ ls

test3.txt

artur@eulaubuntu:~/Videos\$ ls

test3.txt

artur@eulaubuntu:~/Videos\$ mkdir Clases

artur@eulaubuntu:~/Vídeos\$ ls

clases test3.txt

artur@eulaubuntu:~/Vídeos\$ ls

Clases test3.txt

artur@eulaubuntu:~/Videos\$ rmdir Clases

artur@eulaubuntu:~/Vídeos\$ ls

test3.txt

artur@eulaubuntu:~\$ whoami

artur

5.- 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 consultar las diversas opciones que se utilizan en dichos comandos.

I C:

1. -l: Muestra los archivos en formato largo, mostrando detalles como permisos, propietario, grupo, tamaño, fecha de modificación, etc.

1 -a: Muestra todos los archivos, incluyendo los ocultos que empiezan con un punto.

-h: Muestra tamaños de archivos de manera humanamente legible (por ejemplo, en kilobytes, megabytes, gigabytes, etc.).



- -S: Ordena los archivos por tamaño.
- t: Ordena los archivos por fecha y hora de modificación, mostrando los más recientes primero.
- 6 -r: Muestra los archivos en orden inverso.
- 7. -R: Lista de forma recursiva los directorios y sus contenidos.
- 4 --color: Muestra la salida coloreada, resaltando diferentes tipos de archivos con colores.
- 4 -i: Muestra el número de inodo de cada archivo.
- 1. -d: Lista solo los nombres de directorios.
- ## -F: Añade un indicador al final de cada nombre de archivo para indicar el tipo de archivo (por ejemplo, "/" para directorios, "*" para archivos ejecutables, etc.).
- 1 --group-directories-first: Muestra primero los directorios, seguidos de los archivos.
- 13 --ignore: Omite archivos que coincidan con un patrón especificado.
- 4 -U: No ordena; lista los archivos en el orden en que están en el directorio.
- 5. -1: Lista un archivo por línea.

PWD:

- 1. -L: Si la variable de entorno PWD contiene un nombre de directorio simbólico, pwd imprimirá el nombre del directorio al que apunta, seguido de un signo de igualdad y el nombre del directorio de trabajo actual. Esto es útil si estás trabajando en un directorio simbólico y deseas ver la ruta real.
- -P: Obliga a pwd a imprimir la ruta física real del directorio de trabajo actual, omitiendo cualquier enlace simbólico.

CLEAR:

No tiene parámetros directos, pero a veces se puede usar en combinación con redirecciones de entrada y salida (> o >>) para limpiar un archivo o crear uno nuevo.

Por ejemplo:

clear > archivo.txt

Esto limpiará la pantalla y escribirá la salida vacía en un archivo llamado "archivo.txt".

CD:

No tiene parámetros como otros comandos. Simplemente se utiliza para cambiar de directorio, y puede tomar varios argumentos para especificar a qué directorio se desea cambiar.

- de cd [directorio]: Cambia al directorio especificado. Puedes proporcionar la ruta completa o relativa al directorio al que deseas cambiar.
- cd ..: Retrocede un nivel en la jerarquía de directorios, es decir, cambia al directorio padre del directorio actual.
- cd ~ o cd: Cambia al directorio de inicio del usuario actual.



- cd -: Cambia al directorio anterior, es decir, al directorio desde el que se cambió al directorio actual.
- cd /: Cambia al directorio raíz del sistema de archivos.

CAT:

- cat [archivo]: Muestra el contenido del archivo especificado en la salida estándar.
- cat [archivo1] [archivo2] ...: Muestra el contenido de múltiples archivos concatenados en la salida estándar.
- cat -n [archivo]: Muestra el contenido del archivo con números de línea.
- 4 cat -b [archivo]: Muestra el contenido del archivo con números de línea solo para líneas no vacías.
- cat -s [archivo]: Muestra el contenido del archivo con líneas en blanco condensadas a una sola línea.
- 6 cat -E [archivo]: Muestra el contenido del archivo con un signo de dólar (\$) al final de cada línea.
- cat > [archivo]: Crea un nuevo archivo y permite al usuario escribir en él desde la entrada estándar (se debe usar Ctrl + D para finalizar la entrada).
- cat >> [archivo]: Añade contenido a un archivo existente desde la entrada estándar (se debe usar Ctrl + D para finalizar la entrada).
- 9. cat -A [archivo]: Muestra el contenido del archivo con caracteres no imprimibles (como tabulaciones y retornos de carro) representados de manera visible.

M:

- 1. rm [archivo1] [archivo2] ...: Elimina los archivos especificados.
- rm -r [directorio]: Elimina un directorio y su contenido de forma recursiva.
- rm -f [archivo]: Élimina el archivo sin préguntar si estás seguro. Útil para evitar confirmaciones repetidas.
- 4 rm -i [archivo]: Solicita confirmación antes de eliminar cada archivo.
- 5 rm -v [archivo]: Muestra información detallada sobre cada archivo eliminado.
- f. rm -d [directorio]: Elimina solo directorios vacíos.
- 7. rm --preserve-root [archivo]: Evita que rm elimine archivos fuera del directorio raíz (/). Esta opción es para evitar la eliminación accidental de archivos del sistema.
- m --help: Muestra la ayuda del comando rm, proporcionando una lista de opciones y ejemplos de uso.

DC:

- 1. ps: Muestra una lista de tus procesos en ejecución en la terminal actual.
- 1. ps -e: Muestra todos los procesos del sistema.
- ps -u [usuario]: Muestra los procesos asociados con un usuario específico.
- 4 ps -aux o ps aux: Muestra todos los procesos del sistema en un formato detallado.
- ps -f: Muestra una salida más detallada que incluye información sobre el control de los procesos.
 - ps -l: Muestra una lista de procesos con información adicional en formato largo.



- ps -t [terminal]: Muestra los procesos asociados con una terminal específica.
- ps -p [PID]: Muestra información sobre un proceso específico, donde [PID] es el ID de proceso.
- 9. ps -N: Muestra solo los procesos cuyos nombres no contienen caracteres ASCII NUL.
- ps -M: Muestra la ruta completa del archivo ejecutable en lugar del nombre de comando truncado.

Ch:

- cp [archivo_origen] [archivo_destino]: Copia el archivo de origen al destino especificado. Si el destino es un directorio, el archivo se copiará en ese directorio con el mismo nombre. Si el destino es un archivo existente, se sobrescribirá.
- cp -r [directorio_origen] [directorio_destino]: Copia un directorio y su contenido de forma recursiva al directorio de destino.
- cp -i [archivo_origen] [archivo_destino]: Solicita confirmación antes de sobrescribir un archivo existente.
- 4 cp -v [archivo_origen] [archivo_destino]: Muestra información detallada sobre los archivos copiados.
- cp -u [archivo_origen] [archivo_destino]: Copia solo archivos que son más nuevos que los archivos existentes en el destino o archivos que no existen en el destino.
- cp -p [archivo_origen] [archivo_destino]: Conserva los atributos del archivo, como la fecha de modificación y los permisos.
- cp -a [archivo_origen] [archivo_destino]: Copia archivos y directorios de forma recursiva y conserva todos los atributos, como los permisos, propietario, grupo, fechas, etc.
- cp --backup[=CONTROL] [archivo_origen] [archivo_destino]: Realiza copias de seguridad de los archivos existentes antes de sobrescribirlos. Puedes especificar el tipo de copia de seguridad a través de CONTROL (por ejemplo, -- backup=numbered).

MV:

- mv [archivo_origen] [archivo_destino]: Mueve el archivo de origen al destino especificado. Si el destino es un directorio, el archivo se moverá a ese directorio. Si el destino es un archivo existente, el archivo de origen se renombrará al nombre de destino.
- nw -i [archivo_origen] [archivo_destino]: Solicita confirmación antes de sobrescribir un archivo existente.
- mv -v [archivo_origen] [archivo_destino]: Muestra información detallada sobre los archivos movidos o renombrados.
- 4 mv -u [archivo_origen] [archivo_destino]: Mueve solo archivos que son más nuevos que los archivos existentes en el destino o archivos que no existen en el destino.
- mv -f [archivo_origen] [archivo_destino]: Sobrescribe el archivo de destino sin preguntar, incluso si el archivo de destino es de solo lectura o el archivo de origen no existe.



mv -b [archivo_origen] [archivo_destino]: Realiza copias de seguridad de los archivos existentes antes de sobrescribirlos.

mv --backup[=CONTROL] [archivo_origen] [archivo_destino]: Realiza copias de seguridad de los archivos existentes antes de sobrescribirlos. Puedes especificar el tipo de copia de seguridad a través de CONTROL (por ejemplo, -- backup=numbered).

MYDIK:

- mkdir [directorio]: Crea un directorio con el nombre especificado en el directorio actual.
- 1 mkdir -p [directorio]: Crea un directorio y, si es necesario, también crea cualquier directorio padre que falte en la ruta. Esta opción es útil cuando necesitas crear una estructura de directorios jerárquica.
- mkdir -m [modo] [directorio]: Especifica los permisos del nuevo directorio mediante el modo octal (por ejemplo, mkdir -m 755 carpeta).
- 4 mkdir -v [directorio]: Muestra información detallada sobre los directorios creados.
- 5. mkdir --help: Muestra la ayuda del comando mkdir, proporcionando una lista de opciones y ejemplos de uso.

EMDIK:

- rmdir [directorio]: Elimina el directorio especificado si está vacío. Si el directorio no está vacío, rmdir mostrará un mensaje de error.
- rmdir -p [directorio]: Elimina el directorio especificado y, de forma recursiva, también elimina cualquier directorio padre vacío que quede después de eliminar el directorio especificado. Esta opción es útil cuando deseas eliminar una estructura de directorios vacíos de forma eficiente.
- rmdir --ignore-fail-on-non-empty [directorio]: Elimina el directorio especificado incluso si no está vacío. Ten en cuenta que esta opción puede causar la eliminación de archivos y subdirectorios no deseados si el directorio no está vacío.
- 4 rmdir --help: Muestra la ayuda del comando rmdir, proporcionando una lista de opciones y ejemplos de uso.

WHOAMI: No tiene parámetros ni opciones es un comando muy simple que solo muestra el nombre asociado al usuario con el proceso que lo ejecuta.

Abra el editor de texto de su preferencia y realice un programa en lenguaje C que imprima en pantalla CHola mundo". Compile y ejecute su programa desde la Terminal, para ello utilice el comando siguiente: gcc nombre_programa.c -o nombre_salida para compilar, y ./nombre_salida para ejecutar el programa.





```
#include <stdio.h>

int main()

{
    printf("Hola mundo\n");
    return 0;
}
```

vboxuser@UBUNTU:~/Escritorio\$ gcc holamundo.c -o holamundo
vboxuser@UBUNTU:~/Escritorio\$./holamundo
Hola mundo

1. Realice un programa que muestre la siguiente salida:



El dato de entrada del programa será el número de asteriscos de la línea horizontal en la figura con la mayor cantidad de ellos (por ejemplo: para la figura dada es 14). Este dato se pedirá al usuario y podrá ser cualquier valor dentro de un rango que usted establezca, el cual permita visualizar correctamente la figura en la pantalla. 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 sólo bibliotecas estándares de ANSI C. Reporte tanto el código fuente como las pantallas de ejecución del programa.

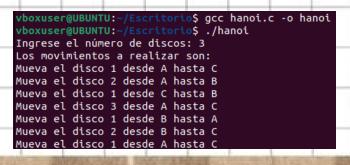




```
1 #include <stdio.h>
3
    // Función para imprimir un triángulo con la base dada
    void imprimir_figura(int base) {
        int i, j, espacios, asteriscos;
        // Imprimir el triángulo hacia arriba
        for (i = 0; i < base; i++) {
            espacios = base - i - 1;
9
            asteriscos = 2 * i + 1;
10
11
            // Imprimir espacios antes de los asteriscos
12
13
            for (j = 0; j < espacios; j++) {
14
               printf(" ");
15
            // Imprimir línea de asteriscos
18
            if (espacios > 0) {
                for (j = 0; j < asteriscos; j++) {
19
                   printf("*");
20
21
                }
22
23
                // Imprimir espacios después de los asteriscos
24
                for (j = 0; j < espacios + (base - 1); j++) {
25
                   printf(" ");
26
27
28
                printf("*");
29
            } else {
30
                // Si es la línea base, imprimir todos los asteriscos juntos
31
                for (j = 0; j < asteriscos * 2; j++) {
    printf("*");</pre>
32
33
34
35
            }
36
37
            printf("\n");
38
39
40
        // Imprimir el triángulo hacia abajo
        for (i = base - 1; i > 0; i--) {
41
            espacios = base - i;
42
            asteriscos = 2 * i - 1;
43
44
45
            // Imprimir espacios antes del primer asterisco
46
            for (j = 0; j < base - 1; j++) {
47
              printf(" ");
48
49
50
            printf("*");
51
            // Imprimir espacios entre asteriscos
52
53
            for (j = 0; j < base - 1 + espacios; j++) {
54
               printf(" ");
55
56
57
            // Imprimir línea de asteriscos
            for (j = 0; j < asteriscos; j++) {
               printf("*");
60
61
            printf("\n");
62
        }
63
64 }
65
66 int main() {
67
68
69
        printf("Ingrese la cantidad de asteriscos en la base del triángulo: ");
70
        scanf("%d", &base);
71
        // Si la base es par, incrementarla en 1 para asegurar un triángulo simétrico
72
        if (base % 2 == 0) {
73
74
            base++;
75
76
77
        // Llamar a la función para imprimir el triángulo
78
        imprimir_figura(((base + 1) / 2) / 2);
79
80
        return 0;
81 }
82
```

8 - Realice un programa que resuelva las torres de Hanoi para n discos de forma recursiva. Reporte tanto el código fuente como las pantallas de ejecución del programa.

```
#include <stdio.h>
    // Función para resolver el problema de las Torres de Hanoi
    void hanoi(int n, char origen, char destino, char auxiliar) {
        // Caso base: solo hay un disco para mover
        if (n == 1) {
           printf("Mueva el disco 1 desde %c hasta %c\n", origen, destino);
10
        // Mover n-1 discos de la torre origen a la torre auxiliar, usando la torre destino como auxiliar
11
12
        hanoi(n - 1, origen, auxiliar, destino);
13
        // Mover el disco n de la torre origen a la torre destino
14
15
        printf("Mueva el disco %d desde %c hasta %c\n", n, origen, destino);
16
17
        // Mover los n-1 discos restantes de la torre auxiliar a la torre destino, usando la torre origen como auxiliar
18
        hanoi(n - 1, auxiliar, destino, origen);
19 }
20
21 int main() {
22
        int num_discos;
23
24
        // Solicitar al usuario el número de discos
25
        printf("Ingrese el número de discos: ");
26
        scanf("%d", &num_discos);
27
28
        // Mostrar los movimientos a realizar
        printf("Los movimientos a realizar son:\n");
30
31
        // Llamar a la función Hanoi para resolver el problema
        hanoi(num_discos, 'A', 'C', 'B');
33
34
        return 0;
35 }
```



9.- Realice un programa que determine si un grupo de paréntesis están balanceados utilizando pilas. Deberán considerarse tres tipos de paréntesis: (), { }, []. Reporte tanto el código fuente como las pantallas de ejecución del programa.

```
#include <stdlib.h>
    // Definición de la estructura de la pila
    typedef struct {
        char *elementos;
        int tope;
        unsigned int capacidad;
   } Pila;
    // Función para inicializar la pila
    void crearPila(Pila *pila, unsigned int capacidad) {
   pila->capacidad = capacidad;
14
        pila->tope = -1;
        pila->elementos = (char*) malloc(pila->capacidad * sizeof(char));
16
   }
   // Función para verificar si la pila está vacía
18
   int estaVacia(Pila *pila) {
        return pila->tope == -1;
21 }
23 // Función para verificar si la pila está llena
    int estallena(Pila *pila) {
        return pila->tope == pila->capacidad - 1;
   // Función para agregar un elemento a la pila
    void apilar(Pila *pila, char elemento) {
       if (estallena(pila)) {
           printf("La pila está llena\n");
32
        pila->elementos[++pila->tope] = elemento;
35
37
    // Función para eliminar un elemento de la pila
    char desapilar(Pila *pila) {
        if (estaVacia(pila)) {
           printf("La pila está vacía\n");
41
            return '\0';
        return pila->elementos[pila->tope--];
43
44
   }
    // Función para verificar si los paréntesis están balanceados
46
    int estanBalanceados(char *expresion) {
48
        Pila pila;
        crearPila(&pila, 100); // Capacidad arbitraria
        for (int i = 0; expresion[i] != '\0'; i++) {
51
            if (expresion[i] == '(' || expresion[i] == '{' || expresion[i] == '[') {
53
                apilar(&pila, expresion[i]);
                continue;
55
            if (estaVacia(&pila)) {
57
58
            switch (expresion[i]) {
60
                case ')':
   x = desapilar(&pila);
62
                    if (x != '(') {
                        return 0;
63
                    break:
                case '}':
x = desapilar(&pila);
67
                    if (x != '{') {
                        return 0;
69
                break;
case ']':
                     x = desapilar(&pila);
                    if (x != '[') {
                        return 0;
                    break;
```

```
80     return estaVacia(&pila);
81    }
82
83    int main() {
84         char expresion[100];
85         printf("Ingrese una expresión con paréntesis, corchetes o llaves: ");
86         scanf("%s", expresion);
87         if (estanBalanceados(expresion)) {
88               printf("Los paréntesis están balanceados.\n");
89         } else {
90               printf("Los paréntesis no están balanceados.\n");
91         }
92         return 0;
93    }
94
```

vboxuser@UBUNTU:~/Escritorio\$ gcc balanceo.c -o balanceo
vboxuser@UBUNTU:~/Escritorio\$./balanceo
Ingrese una expresión con paréntesis, corchetes o llaves: ()[{}]
Los paréntesis están balanceados.

10.- Realice un programa que evalúe cualquier expresión aritmética delimitada completamente con paréntesis balanceados (por ejemplo, entrada: (((5-2)+4)*10), salida expresión evaluada: 70). Como restricción del programa se tiene que no deberá de utilizar notación prefija ni posfija para la evaluación. Reporte tanto el código fuente como las pantallas de ejecución del programa.

```
2 #include <stdlib.h>
 3 #include <string.h>
 4 #include <ctype.h>
    // Función para obtener la precedencia de los operadores
   int precedencia(char op);
10 // Función para aplicar el operador
int aplicarOperador(int a, int b, char op);
13
14
15 // Estructura para la pila
   struct Pila {
17
       int tope;
18
       unsigned capacidad;
19
       int* arreglo;
20 };
21
22 // Crear una pila vacía
   struct Pila* crearPila(unsigned capacidad) {
      struct Pila* pila = (struct Pila*)malloc(sizeof(struct Pila));
        pila->capacidad = capacidad;
25
26
       pila->tope = -1;
       pila->arreglo = (int*)malloc(pila->capacidad * sizeof(int));
28
       return pila:
29 }
30
   // Verificar si la pila está vacía
   int estaVacia(struct Pila* pila) {
32
33
      return pila->tope == -1;
34
35
   // Empujar un elemento a la pila
36
37
    void push(struct Pila* pila, int elemento) {
       pila->arreglo[++pila->tope] = elemento;
39
40
```

```
// Sacar un elemento de la pila
42
    int pop(struct Pila* pila) {
43
        if (!estaVacia(pila))
44
            return pila->arreglo[pila->tope--];
45
        return -1; // Valor sentinela para errores
46
   }
47
48
    // Obtener el tope de la pila sin eliminarlo
    int peek(struct Pila* pila) {
49
50
        if (!estaVacia(pila))
51
            return pila->arreglo[pila->tope];
52
        return -1; // Valor sentinela para errores
53
54
55
    // Evaluar una expresión aritmética con paréntesis balanceados
56
    int evaluarExpresion(const char* expresion) {
57
        struct Pila* pila = crearPila(strlen(expression));
58
59
        for (i = 0; expresion[i]; ++i) {
60
61
           if (expresion[i] == '(')
                push(pila, expresion[i]);
62
63
            else if (expresion[i] == ')') {
64
                if (estaVacia(pila)) {
65
                    printf("Error: Paréntesis desbalanceados\n");
66
                    return -1;
67
68
                pop(pila);
            }
69
70
        }
71
72
        if (!estaVacia(pila)) {
73
            printf("Error: Paréntesis desbalanceados\n");
74
            return -1;
75
76
77
        // Evaluación de la expresión sin los paréntesis
78
        struct Pila* operandos = crearPila(strlen(expresion));
        struct Pila* operadores = crearPila(strlen(expresion));
79
80
```

```
81
         for (i = 0; expresion[i]; ++i) {
 82
             if (isdigit(expresion[i])) {
                 int operando = 0;
 83
 84
                 while (isdigit(expresion[i])) {
 85
                     operando = operando * 10 + (expresion[i] - '0');
                 i--;
             push(operandos, operando);
} else if (expresion[i] == '+' || expresion[i] == '-' || expresion[i] == ''') {
 89
                 while (!estaVacia(operadores) && precedencia(expresion[i]) <= precedencia(peek(operadores))) {</pre>
                     int val2 = pop(operandos);
                     int val1 = pop(operandos);
 93
                     char op = pop(operadores);
 94
                     push(operandos, aplicarOperador(val1, val2, op));
 97
                 push(operadores, expresion[i]);
 98
             } else if (expresion[i] == '(') {
                 push(operadores, expresion[i]);
             } else if (expresion[i] == ')') {
101
                 while (!estaVacia(operadores) && peek(operadores) != '(') {
102
                     int val2 = pop(operandos);
                     int val1 = pop(operandos);
194
                     char op = pop(operadores);
105
                     push(operandos, aplicarOperador(val1, val2, op));
106
107
                 pop(operadores); // Pop '('
108
109
         }
110
         while (!estaVacia(operadores)) {
             int val2 = pop(operandos);
113
             int val1 = pop(operandos);
114
             char op = pop(operadores);
             push(operandos, aplicarOperador(val1, val2, op));
115
```

```
int resultado = pop(operandos);
118
119
         free(pila->arreglo):
120
         free(pila);
         free(operandos->arreglo);
122
         free(operandos);
123
         free(operadores->arreglo);
124
         free(operadores);
125
        return resultado;
127
128 // Función para obtener la precedencia de los operadores
129 int precedencia(char op) {
        if (op == '+' || op == '-')
130
            return 1;
         if (op == '*' || op == '/')
132
133
            return 2;
        return 0;
134
135 }
137 // Función para aplicar el operador
    int aplicarOperador(int a, int b, char op) {
139
      switch (op) {
140
            case '+':
141
                return a + b;
                return a - b;
143
144
            case '*':
               return a * b:
145
            case '/':
146
               if (b != 0)
                    return a / b;
149
                     printf("Error: División por cero\n");
150
151
                     exit(EXIT_FAILURE);
152
154
        return 0; // No debería llegar aquí
155 }
156
157 int main() {
        char expresion[100]; // Suponemos que la expresión no excederá los 100 caracteres
158
         printf("Ingrese la expresión aritmética: ");
160
         fgets(expresion, sizeof(expresion), stdin); // Lee la expresión desde la entrada estándar
161
        // Elimina el salto de línea final agregado por fgets si está presente
162
        if ((strlen(expression) > 0) && (expression[strlen(expression) - 1] == '\n'))
163
             expresion[strlen(expresion) - 1] = '\0';
165
166
        int resultado = evaluarExpresion(expresion);
167
        if (resultado != -1)
168
169
            printf("Resultado de la expresión: %d\n", resultado);
170
171
172 }
```

vboxuser@UBUNTU:~/Escritorio\$ gcc expresion.c -o expresion
vboxuser@UBUNTU:~/Escritorio\$./expresion
Ingrese la expresión aritmética: (((5-2)+4)*10)
Resultado de la expresión: 70

11.- Guarde todos sus programas (fuente y ejecutable) en una memoria usb.

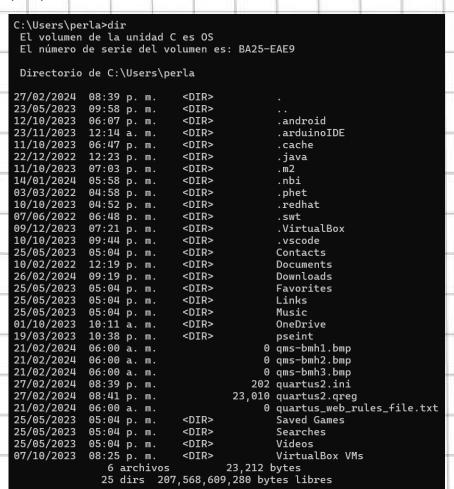
SECCIÓN WINDOWS

- I Inicie sesión en Windows.
- 1 Abra una consola.
- Investigue en Internet 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).

| NOTE OF THE PARTY | | | |
|---|----------------------------|-------------------------|--|
| Dir | Cd nombre_directorio | Copy nombre_archivo | |
| | | nuevo_nombre_archivo | |
| ipconfig | Type nombre_archivo | Ren nombre_archivo | |
| | | nuevo_nombre_archivo | |
| Cls | Mkdir nombre_directorio | Chdir nombre_directorio | |
| | | | |
| Ver | rmdir nombre_directorio | echo "Hola Mundo" | |
| | | | |
| Tree | del nombre_archivo | Find "cadena_buscar" | |
| | | nombre_archiv | |

Reporte la función de cada uno de los anteriores comandos ejecutados.

Muestra una lista de los archivos y subdirectorios de un directorio. Si se usa sin parámetros, este comando muestra la etiqueta de volumen y el número de serie del disco, seguido de una lista de directorios y archivos del disco (como los nombres y la fecha y hora en que se modificó por última vez). En el caso de archivos, este comando muestra la extensión de nombre y el tamaño en bytes. Este comando también muestra el total de archivos y directorios que aparecen, el tamaño acumulado y el espacio libre (en bytes) que queda en el disco.



con una letra de unidad (por ejemplo, cd C:), cd muestra los nombres del directorio actual en la unidad especificada. Si se usa sin parámetros, cd muestra la unidad y el directorio actuales.

```
C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1\Prueba>cd..
```

- C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1>cd..
- C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS>cd..
- C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE>cd SISTEMAS OPERATIVOS
- C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS>

la configuración del Protocolo de configuración dinámica de host (DHCP) y del Sistema de nombres de dominio (DNS). Si se usa sin parámetros, ipconfig muestra la versión 4 (IPv4) del protocolo de Internet y las direcciones IPv6, la máscara de subred y la puerta de enlace predeterminada para todos los adaptadores.

```
C:\Users\perla>ipconfig
Configuración IP de Windows
Adaptador de Ethernet Ethernet 2:
   Sufijo DNS específico para la conexión. . :
   Vínculo: dirección IPv6 local. . . : fe80::53a0:ec82:389e:336d%15
   Dirección IPv4. . . . . . . . . . . . . . . 192.168.56.1
   Puerta de enlace predeterminada . . . .
Adaptador de LAN inalámbrica Conexión de área local* 1:
   Estado de los medios. . . . . . . . . : medios desconectados
   Sufijo DNS específico para la conexión. . :
Adaptador de LAN inalámbrica Conexión de área local* 2:
   Estado de los medios. . . . . . . . . . . medios desconectados
   Sufijo DNS específico para la conexión. . :
Adaptador de LAN inalámbrica Wi-Fi:
   Sufijo DNS específico para la conexión. . :
   Dirección IPv6 . . . . . . . . : 2806:2f0:9180:ff9a:2c1d:6e2e:280f:aea6
Dirección IPv6 temporal . . . . : 2806:2f0:9180:ff9a:d42c:8e32:4b0c:df57
Vínculo: dirección IPv6 local . . : fe80::fa6c:7f09:6110:57bc%17
   Dirección IPv4. . . . . . . . . . . . . . . . . . 192.168.1.21
   Adaptador de Ethernet Conexión de red Bluetooth:
                                       . . . : medios desconectados
   Estado de los medios. .
   Sufijo DNS específico para la conexión. . :
```

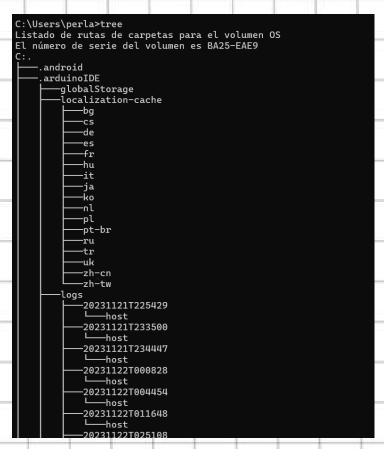
CLS: Borra la ventana del símbolo del sistema.

C:\Users\perla>

VEN: Muestra el número de versión del sistema operativo. Este comando se admite en el símbolo del sistema de Windows (Cmd.exe), pero no en ninguna versión de PowerShell.

C:\Users\perla>ver Microsoft Windows [Versión 10.0.22621.3155]

TIPEE: Muestra la estructura de directorios de una ruta de acceso o del disco en una unidad gráficamente. La estructura mostrada por este comando depende de los parámetros que especifique en el símbolo del sistema. Si no especifica una unidad o ruta de acceso, este comando muestra la estructura de árbol que comienza con el directorio actual de la unidad actual.







con una letra de unidad (por ejemplo, cd C:), cd muestra los nombres del directorio actual en la unidad especificada. Si se usa sin parámetros, cd muestra la unidad y el directorio actuales.

C:\Users\perla\OneDrive>cd\

C:\>cd users

C:\Users>cd perla

C:\Users\perla>cd OneDrive

C:\Users\perla\OneDrive>

En el shell de comandos de Windows, type es un comando integrado que muestra el contenido de un archivo de texto. Use el comando type para ver un archivo de texto sin modificarlo. En PowerShell, type es un alias integrado para el Get-Content cmdlet, que también muestra el contenido de un archivo, pero con una sintaxis diferente

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1>type prueba.docx
D:%/,f[Content_Types].xml ó(á|ö╦n¬0E,ò·ûÀUbÞó¬*ï>û-RÚ{V³Æm³=lü;¬ÇH6æÆÖ{'Ö\$×½hm4YBê½½è÷╦\$`àô½½*·5y+)ëë[¡fá|PĐ
C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1>type Prueba.txt
Hola Mundo! Estas usando type!

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>type con > Prueba.txt
Hago prueba

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>type Prueba.txt
Hago prueba

habilitadas de manera predeterminada, permiten usar un comando mkdir para crear directorios intermedios en una ruta de acceso especificada.

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1>MKDIR Prueba\1

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1>dir

El volumen de la unidad C es OS

El número de serie del volumen es: BA25-EAE9

Directorio de C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SIST EMAS OPERATIVOS\PRACTICA1

```
29/02/2024 07:45 p. m. <DIR> .
29/02/2024 06:03 p. m. <DIR> ..
29/02/2024 07:49 p. m. <DIR> Prueba
29/02/2024 06:49 p. m. 14,770 Prueba.docx
29/02/2024 06:50 p. m. 30 Prueba.txt
2 archivos 14,800 bytes
3 dirs 207,619,219,456 bytes libres
```





Consola de recuperación de Windows mediante distintos parámetros.

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS \PRACTICA1>rmdir Prueba

El directorio no está vacío.

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1>mkdir Prueba\vacio

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1>cd Prueba

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>rmdir vacio

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>mkdir NoVacio

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS \PRACTICA1\Prueba>mkdir NoVacio\uno NoVacio\dos

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>rmdir /S Prueba
Prueba, ¿Está seguro (S/N)? S

DEL: Elimina uno o varios archivos. Este comando realiza las mismas acciones que el comando erase. El comando del también se puede ejecutar desde la Consola de recuperación de Windows mediante parámetros diferentes.

Directorio de C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SIST EMAS OPERATIVOS\PRACTICA1\Prueba

```
29/02/2024
              08:04 p. m.
                                  <DIR>
29/02/2024 08:01 p. m.
                                 <DIR>
29/02/2024 07:49 p. m.
29/02/2024 07:56 p. m.
29/02/2024 08:03 p. m.
29/02/2024 06:50 p. m.
                                 <DIR>
                                 <DIR>
                                                     NoVacio
                                          325,815 PRUEBA.docx
                                                 30 Prueba.txt
29/02/2024 08:02 p. m.
                                                  0 Prueba2.txt
                                        325,845 bytes
                   3 archivos
                  4 dirs 207,605,743,616 bytes libres
```

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS \PRACTICA1\Prueba>del *.txt

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>del *

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS \PRACTICA1\Prueba*, ¿Está seguro (S/N)? S

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS \PRACTICA1\Prueba>dir

El volumen de la unidad C es OS

El número de serie del volumen es: BA25-EAE9

Directorio de C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SIST EMAS OPERATIVOS\PRACTICA1\Prueba

```
29/02/2024 08:05 p. m. <DIR> .
29/02/2024 08:01 p. m. <DIR> ..
29/02/2024 07:49 p. m. <DIR> 1
29/02/2024 07:56 p. m. <DIR> NoVacio
0 archivos 0 bytes
4 dirs 207,604,510,720 bytes libres
```



Copia uno o varios archivos de una ubicación a otra.

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>type con > Prueba.txt
Hago prueba

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>type Prueba.txt
Hago prueba

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1\Prueba>dir

El volumen de la unidad C es OS

El número de serie del volumen es: BA25-EAE9

Directorio de C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SIST EMAS OPERATIVOS\PRACTICA1\Prueba

```
29/02/2024 08:11 p. m.
                          <DIR>
29/02/2024 08:01 p. m.
                          <DIR>
29/02/2024 07:49 p. m.
                          <DIR>
29/02/2024 07:56 p. m.
                          <DIR>
                                         NoVacio
29/02/2024 08:09 p. m.
                                      13 Prueba.txt
29/02/2024 08:09 p. m.
                                      13 PruebaCopia.txt
                                     26 bytes
              2 archivos
              4 dirs 207,610,576,896 bytes libres
```

LEM: Cambia el nombre de archivos o directorios.

```
C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1\Prueba>ren Prueba.txt PruebaRename.txt
El volumen de la unidad C es OS
El número de serie del volumen es: BA25-EAE9
 Directorio de C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1\Prueba
29/02/2024 08:01 p. m.
29/02/2024 07:49 p. m.
                             <DIR>
                                          NoVacio
13 PruebaCopia.txt
13 PruebaRename.txt
29/02/2024 07:56 p. m.
29/02/2024 08:09 p. m.
                             <DIR>
29/02/2024 08:09 p.m. 13 Pruebakene
29/02/2024 08:09 p.m. 26 bytes
2 archivos 26 bytes
4 dirs 207,607,263,232 bytes libres
C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO_SEMESTRE\STSTEMAS_OPERATIVOS\PRACTICA1\Prueba>ren_1_RenameDirec
El volumen de la unidad C es OS
El número de serie del volumen es: BA25-EAE9
 Directorio de C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1\Prueba
29/02/2024 08:14 p. m.
29/02/2024 08:01 p. m.
29/02/2024 07:56 p. m.
                             <DIR>
                                             NoVacio
                             <DIR>
                                          13 PruebaCopia.txt
13 PruebaRename.txt
29/02/2024 08:09 p. m.
            08:09 p. m.
                            <DIR>
                                             RenameDirec
                2 archivos 26 bytes
4 dirs 207,610,155,008 bytes libres
C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1\Prueba>
```

CHDIM: Muestra el nombre del directorio actual o cambia el directorio actual. Si se usa solo con una letra de unidad (por ejemplo, chdir C:), chdir muestra los nombres del directorio actual en la unidad especificada. Si se usa sin parámetros, chdir muestra la unidad y el directorio actuales.



C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS \PRACTICA1\Prueba>cd..

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
\PRACTICA1>cd..

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS
>cd..

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE>cd..

C:\Users\perla\OneDrive\Documentos\ESCOM>cd..

C:\Users\perla\OneDrive\Documentos>cd..

C:\Users\perla\OneDrive>cd..

C:\Users\perla>chdir

C:\Users\perla

C:\Users\perla>chdir C:

C:\Users\perla

C:\Users\perla>chdir C:\

C:\>chdir Users

C:\Users>chdir perla

C:\Users\perla>

ECHO: Muestra mensajes o activa o desactiva la característica de eco de comandos. Si se usa sin parámetros, echo muestra la configuración de eco actual.

C:\Users\perla>echo Hola, mundo!
Hola, mundo!

C:\Users\perla>echo %PATH%

C:\Program Files\Eclipse Adoptium\jdk-21.0.0.35-hotspot\bin;C:\Program Files \Common Files\Oracle\Java\javapath;C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\OpenSSH\;C:\Program Files\MySQL\MySQL Shell 8.0\bin\;C:\Users\perla\AppData\Local\Microsoft\WindowsApps;C:\Users\perla\AppData\Local\Programs\Microsoft VS Code\bin;C:\Program Files\JetBrains\IntelliJ IDEA 2023.2.3\bin;;C:\altera\13.0sp1\modelsim_ase\win32aloem

C:\Users\perla>echo on

C:\Users\perla>echo off

FIMD: Busca una cadena de texto en un archivo o archivos y muestra líneas de texto que contienen la cadena especificada.

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1>CD PRUEBA

C:\Users\perla\OneDrive\Documentos\ESCOM\CUARTO SEMESTRE\SISTEMAS OPERATIVOS\PRACTICA1\Prueba>find "Hago" PruebaCopia.tx

----- PRUEBACOPIA.TXT

Hago prueba



<u>B E A U T I F U L</u>

Conecte su memoria usb donde tiene los programas que desarrolló en la sección de Linux y ejecútelos directamente. Reporte sus observaciones sobre la ejecución de estos.

Al intentar ejecutar las salidas compiladas en linux (ubuntu) desde windows. Windows no logra reconocer el archivo.

Microsoft Windows [Versión 10.0.22621.3155]

(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>./astericos
"." no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>asteriscos "asteriscos" no se reconoce como un comando interno o externo, programa o archivo por lotes ejecutable.

- Busque el directorio donde está instalado Dev C o algún IDE de C que tenga como compilador a gcc.
- l Desde la consola ubíquese en el directorio de Dev C, y cambie al directorio Cbin".
- Desde ese directorio compile en la consola como se indica en el punto 6 de la sección Linux cada uno de los programas creados en esa sección y que guardó en la memoria usb. Reporte la ejecución de todos los programas.

Microsoft Windows [Versión 10.0.22621.3155]

(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>gcc holamundo.c -o holamundo

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>holamundo Hola mundo

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>gcc asteriscos.c -o asteriscos

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>asteriscos Ingrese la cantidad de asteriscos en la base del tri¦íngulo: 14

* *

*** *

***** *

* *****

* ***

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>gcc hanoi.c -o hanoi

Ingrese el nimero de discos: 3
Los movimientos a realizar son:
Mueva el disco 1 desde A hasta C
Mueva el disco 2 desde A hasta B
Mueva el disco 1 desde C hasta B
Mueva el disco 3 desde A hasta C
Mueva el disco 1 desde B hasta A
Mueva el disco 2 desde B hasta C
Mueva el disco 1 desde A hasta C





C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>gcc balanceo.c -o balanceo

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>balanceo Ingrese una expresi¦|n con par¦®ntesis, corchetes o llaves: ()[{}] Los par¦®ntesis est¦ín balanceados.

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>gcc expresion.c -o expresion

C:\Users\jarro\OneDrive\Escritorio\SISTEMAS OPERATIVOS\UBUNTU>expresion Ingrese la expresi||n aritm|@tica: (((5-2)+4)*10)
Resultado de la expresi||n: 70

Reporte las diferencias y similitudes entre los comandos de Linux y Windows, así como la compilación y ejecución de los programas realizados, que haya encontrado.

DIFERENCIAS:

Sistema Operativo:

- <u>Linux</u>: Se utiliza una distribución específica de Linux, como Ubuntu, Fedora, CentOS, etc. Cada distribución puede tener diferentes configuraciones y herramientas preinstaladas.
- <u>Windows</u>: Se utiliza el sistema operativo Windows, que puede ser Windows 7, 8, 10, etc. Cada versión de Windows tiene sus propias características y herramientas únicas.

Entorno de línea de comandos:

- <u>Linux</u>: Utiliza la Terminal, que es una interfaz de línea de comandos donde los usuarios interactúan con el sistema operativo utilizando comandos de texto.
- <u>Windows</u>: Utiliza el símbolo del sistema (Command Prompt) o PowerShell, que son interfaces similares a la Terminal de Linux, pero tienen diferentes comandos y características.

Comandos del sistema:

- <u>Linux:</u> Los comandos del sistema son diferentes de los de Windows. Por ejemplo, para listar archivos y directorios en Linux se utiliza el comando ls, mientras que en Windows se utiliza dir.
- Windows: Los comandos del sistema en Windows varían de los de Linux.
 Por ejemplo, para cambiar de directorio en Windows se utiliza el comando cd, al igual que en Linux, pero la sintaxis puede ser diferente en algunos casos.

Compilación y ejecución de programas en C:

- <u>Linux</u>: Se utiliza el compilador GCC (GNU Compiler Collection) para compilar programas en C. Para ejecutar los programas compilados, se utiliza el comando ./nombre programa.
- <u>Windows</u>: Se puede utilizar Dev C++ u otro IDE con el compilador GCC integrado, o bien, el compilador GCC instalado directamente Los programas compilados se ejecutan con el nombre del archivo ejecutable, como nombre_programa.exe.

Estructura de directorios:



<u>Linux</u>: Utiliza una estructura de directorios diferente a la de Windows. Por ejemplo, en Linux, el directorio raíz se representa como /, mientras que en Windows se representa como **C:**\.

<u>Windows</u>: Utiliza una estructura de directorios diferente, con unidades de disco como C:\, D:\, etc., y directorios como Users, Program Files, etc.

SIMILITUDES:

Algunos comandos son similares:

 Aunque los comandos del sistema son diferentes en Linux y Windows, algunos tienen funcionalidades similares. Por ejemplo, ambos sistemas tienen comandos para listar archivos, cambiar de directorio, copiar archivos, etc.

1. Compilación y ejecución de programas en C:

 Ambos sistemas utilizan GCC como compilador de C, lo que significa que el proceso de compilación es similar en ambos sistemas.

• Una vez compilados, los programas se ejecutan de manera similar en ambos sistemas, aunque la sintaxis del comando puede variar ligeramente (./nombre_programa en Linux, nombre_programa.exe en Windows).

Capacidad para ejecutar programas compilados:

• Tanto en Linux como en Windows, los programas compilados pueden ejecutarse una vez que se han otorgado los permisos adecuados en el caso de Linux, o se han configurado los permisos de ejecución en Windows.

Uso de pilas y evaluación de expresiones:

 Ambos sistemas pueden realizar tareas como evaluar expresiones aritméticas o verificar el balanceo de paréntesis utilizando pilas.

 Aunque los comandos específicos pueden ser diferentes, los conceptos subvacentes y las técnicas utilizadas son similares en ambos sistemas.

Almacenamiento de programas:

 En ambos sistemas, los programas y archivos pueden almacenarse en unidades externas como memorias USB, y pueden ser ejecutados desde allí.

Acceso a Internet:

 Tanto en Linux como en Windows, se puede acceder a Internet para buscar información sobre comandos o realizar otras tareas relacionadas con el desarrollo de software.

Conclusiones

Nombre:

Considero que tanto Linux como Windows tienen sus ventajas y desventajas en el ámbito del desarrollo de software. En mi experiencia, he observado que ambos sistemas operativos ofrecen un rendimiento similar al compilar códigos, lo que los hace viables para trabajar en proyectos de programación. Aunque los comandos de Windows me resultan más familiares y rápidos de usar, personalmente prefiero Windows debido a su interfaz más amigable, especialmente para aquellos con menos experiencia en



computación. Sin embargo, reconozco que Linux puede ser más beneficioso para usuarios con conocimientos avanzados, ya que permite modificaciones según las necesidades del usuario. En resumen, mi elección entre Linux y Windows se basa en la accesibilidad y la experiencia del usuario, considerando las necesidades específicas de cada proyecto.

Nombre:

Mi análisis de los sistemas operativos Linux y Windows revela que ambos tienen sus propias fortalezas y debilidades. Aprecio la capacidad de Linux para aprovechar al máximo los recursos del hardware y obtener un rendimiento óptimo en la compilación de programas. Por otro lado, reconozco que Windows se destaca por su facilidad de uso y accesibilidad para los usuarios menos experimentados en computación. No tengo una preferencia clara por uno u otro sistema, ya que reconozco la importancia de evaluar cada herramienta según las necesidades específicas del proyecto o la tarea en cuestión. En última instancia, mi elección entre Linux y Windows se basa en encontrar el equilibrio adecuado entre eficiencia y comodidad de uso.

Nombre:

En mi experiencia, he notado que el sistema operativo Linux ofrece una mayor estabilidad y velocidad en la compilación de programas en comparación con Windows. Aunque reconozco que la interfaz visual de Linux puede resultar más compleja para algunos usuarios, valoro su eficiencia y rendimiento en el desarrollo de software. Sin embargo, también comprendo la importancia de la facilidad de uso y la accesibilidad que ofrece Windows, especialmente para usuarios menos experimentados en tecnología. No tengo una preferencia clara por uno u otro sistema, ya que creo que es crucial evaluar cada uno según las necesidades específicas del proyecto y las habilidades del usuario. En última instancia, mi elección entre Linux y Windows se basa en encontrar el equilibrio entre funcionalidad y facilidad de uso.

Nombre:

Mi experiencia con los sistemas operativos Linux y Windows me ha permitido familiarizarme con la terminal y los comandos en ambas plataformas. Reconozco la importancia de considerar el propósito de uso al elegir un sistema operativo, ya que cada uno tiene sus propias ventajas y desventajas. No tengo una preferencia clara por uno u otro sistema, ya que creo en la adaptabilidad y la capacidad de ajustarme a diferentes entornos tecnológicos según las necesidades del proyecto. Mi análisis equilibrado de las fortalezas y debilidades de Linux y Windows me ha llevado a comprender que la elección entre ambos depende de varios factores, incluidas las preferencias personales y las demandas específicas del trabajo en cuestión. En resumen, estoy abierto a trabajar con cualquier sistema operativo que mejor se adapte a las necesidades del proyecto en ese momento.

Referencia

Britannica, Los editores de la enciclopedia. "Microsoft Windows". Enciclopedia Británica, 31 de marzo de 2021, https://www.britannica.com/technology/Windows-OS. Consultado el 2 de marzo de 2024.

Carretero, J; Anasagasti, P de M; García, F; Pérez, F.. (2001). Sistemas Operativos: Una visión aplicada. Madrid: McGraw-Hill .

Wolf, G; Ruiz, E; Bergero, F; Meza, E. . (2015). Fundamentos de Sistemas Operativos. México: Universidad Nacional Autónoma de México

