

# 1.

**a. Mencione y explique las características más relevantes de GNU/Linux**  
**GNU/Linux se caracteriza por:**

- **Software libre:** se puede usar, copiar, modificar y distribuir.
- **Compatible con UNIX:** mantiene su diseño modular.
- **Portátil:** funciona en muchas arquitecturas y entornos.
- **Multitarea y multiusuario:** varios procesos y usuarios a la vez.
- **Núcleo modular:** permite cargar y descargar módulos en ejecución.
- **Variedad de distribuciones:** diferentes versiones adaptadas a cada necesidad.

**b. Mencione otros sistemas operativos y compárelos con GNU/Linux en cuanto a los puntos mencionados en el inciso a.**

- **Windows:** no es libre, no cumple filosofía UNIX, menos portable, multitarea y multiusuario limitados, núcleo no modular, pocas variantes oficiales.
- **macOS (basado en Unix/BSD):** no es libre, compatible con UNIX, portable solo a hardware Apple, multitarea y multiusuario, kernel híbrido pero cerrado, una sola distribución oficial.

**c. ¿Qué es GNU?**

GNU es un sistema operativo similar a UNIX, compuesto totalmente por software libre. Incluye programas y utilidades desarrollados bajo la filosofía de la Free Software Foundation (FSF).

Su objetivo es ser compatible con UNIX, usando piezas de software modulares y liberado bajo la Licencia GPL, que asegura que cualquiera pueda usarlo, copiarlo, modificarlo y distribuirlo.

**d. Indique una breve historia sobre la evolución del proyecto GNU.**

El proyecto GNU lo hizo Richard Stallman en 1983 para crear un sistema operativo tipo UNIX totalmente libre. En los 80s y 90s se desarrollaron muchas herramientas (compiladores, editores, shells), pero el kernel Hurd nunca terminó de madurar. En 1991 apareció Linux, y al combinarse con GNU en 1992 nació el sistema GNU/Linux, que funcionó completo.

**e. Explique qué es la multitarea, e indique si GNU/Linux hace uso de ella.**

Es la capacidad de un sistema operativo de ejecutar varios procesos al mismo tiempo, compartiendo el procesador. GNU/Linux usa multitarea, permitiendo usar muchos programas al mismo tiempo.

#### **f. ¿Qué es POSIX?**

POSIX es un estándar que define una interfaz común para los sistemas tipo UNIX, para que sean compatibles entre ellos.

## **2.**

#### **a. ¿Qué es una distribución de GNU/Linux? Nombre al menos 4 distribuciones de GNU/Linux y cite diferencias básicas entre ellas.**

Una distribución de GNU/Linux es un sistema operativo completo que combina el kernel Linux, las herramientas GNU y software adicional, empaquetados y mantenidos por una comunidad o empresa.

Ejemplos: Ubuntu, Debian, Red Hat, Arch Linux.

#### **b. ¿En qué se diferencia una distribución de otra?**

Las distribuciones se diferencian por el gestor de paquetes que utilizan, por el público al que apuntan, por el equilibrio entre estabilidad y actualización de software y por el nivel de facilidad de uso y soporte que ofrecen

#### **c. ¿Qué es Debian? Acceda al sitio <https://www.debian.org/> e indique cuáles son los objetivos del proyecto y una breve cronología del mismo.**

Debian es una distribución de GNU/Linux desarrollada por voluntarios desde 1993, quiere ofrecer un sistema completamente libre, gobernado democráticamente mediante un contrato social y normativas claras. Su inicio oficial fue el 16 de agosto de 1993, y en 1997 cristalizó sus valores a través del Social Contract.

## **3.**

#### **a y b. Nombre cuáles son los 3 componentes fundamentales de GNU/Linux.**

Los tres componentes fundamentales de GNU/Linux son el kernel, la shell y el filesystem.

- Kernel: núcleo del sistema, conecta hardware y software.
- Shell: intérprete de comandos, permite la interacción usuario-sistema.
- FileSystem: sistema de archivos, organiza y administra los datos.

## 4.

**Kernel:**

**a. ¿Cuáles son sus funciones principales?**

El kernel gestiona la comunicación entre hardware y software, administra los recursos como memoria procesos dispositivos y sistema de archivos y permite que los programas se ejecuten coordinando el uso del hardware

**b. ¿Cuál es la versión actual? ¿Cómo se definía el esquema de versionado del Kernel en versiones anteriores a la 2.4? ¿Qué cambió en el versionado que se impuso a partir de la versión 2.6?**

En las versiones anteriores a la 2.4, el esquema de versionado distinguía entre números pares y impares:

- Los pares indican versiones estables (ejemplo 2.4).
- Los impares indican versiones de desarrollo no destinadas a producción (ejemplo 2.5).

A partir de la versión 2.6, esa diferencia desapareció y ya no hay diferencia entre pares e impares, y las novedades y herramientas nuevas podían aparecer en cualquier número dentro de la misma serie.

**c. ¿Es posible tener más de un Kernel de GNU/Linux instalado en la misma máquina?**

Sí es posible tener más de un kernel de GNU/Linux instalado en la misma máquina ya que cada versión se guarda como un archivo aparte y el gestor de arranque permite elegir con cuál iniciar lo que permite mantener varias versiones instaladas

**d. ¿Dónde se encuentra ubicado dentro del File System?**

El kernel en GNU/Linux se guarda dentro del directorio /boot del sistema de archivos, donde están el archivo del núcleo y otros ficheros necesarios para que el sistema arranque

## 5.

**Intérprete de comandos (Shell):**

**a. ¿Qué es?**

El Shell es un programa que sirve de interfaz entre el usuario y el sistema operativo recibiendo los comandos que se escriben en la terminal traduciéndose en instrucciones que el sistema entiende y devolviendo la respuesta en la misma ventana

**b. ¿Cuáles son sus funciones?**

Las funciones del Shell son recibir los comandos que el usuario escribe, interpretarlos y pasarlos al sistema operativo para que se ejecuten, también muestra en pantalla la respuesta del sistema y permite la interacción continua entre el usuario y GNU/Linux

**c. Mencione al menos 3 intérpretes de comandos que posee GNU/Linux y compárelos entre ellos.**

En GNU/Linux se usan varios intérpretes de comandos como sh, que es básico y universal; csh, con sintaxis parecida a C; y bash, que es como sh pero con mejoras como historial y autocompletado.

**d. ¿Dónde se ubican (path) los comandos propios y externos al Shell?**

Los comandos propios del Shell se ejecutan directamente adentro, mientras que los comandos externos se buscan en las rutas definidas en la variable PATH, ubicándose normalmente en directorios como /bin, /usr/bin o /usr/local/bin

**e y f. ¿Por qué considera que el Shell no es parte del Kernel de GNU/Linux?**

La shell no forma parte del kernel básico del SO; sino que la misma “dialoga” con el kernel. La shell es iniciada por un proceso denominado “login”, y dado que cada usuario tiene asignado una shell por defecto, la misma se inicia cada vez que un usuario comienza a trabajar en su estación de trabajo (es decir se “loguea” en una terminal). Dentro del contenido del archivo /etc/passwd, se puede ver cual es la shell que cada usuario tiene asignada por defecto.

## 6.

**El sistema de Archivos (File System) en Linux:**

**a. ¿Qué es?**

El sistema de archivos en Linux es la forma en que el sistema operativo organiza y almacena la información en los dispositivos, permitiendo acceder a ella mediante directorios y rutas jerárquicas.

**b. ¿Cuál es la estructura básica de los File System en GNU/Linux? Mencione los directorios más importantes e indique qué tipo de información se encuentra en ellos. ¿A qué hace referencia la sigla FHS?**

El filesystem en GNU/Linux es jerárquico, con raíz /.

Principales directorios: /bin (comandos básicos), /sbin (administración), /etc (configuración), /home (usuarios), /usr (programas), /var (logs y datos variables), /tmp (temporales).

FHS = Filesystem Hierarchy Standard.

### c. Mencione sistemas de archivos soportados por GNU/Linux.

GNU/Linux soporta varios sistemas de archivos, entre ellos:

- ext3: estable y con journaling.
- ext4: más rápido y moderno, soporta archivos grandes.
- ReiserFS: eficiente con muchos archivos pequeños.
- XFS: alto rendimiento con archivos grandes.

### d. ¿Es posible visualizar particiones del tipo FAT y NTFS (que son de Windows) en GNU/Linux?

Sí, se puede. GNU/Linux puede montar y acceder a particiones FAT y NTFS, permitiendo leer y escribir en ellas mediante controladores y herramientas específicas.

## 7.

### Particiones:

#### a. Definición. Tipos de particiones. Ventajas y Desventajas.

Una partición es una división lógica de un disco físico que permite organizar y gestionar mejor el almacenamiento.

#### Tipos de particiones:

- Primarias: hasta 4 por disco; pueden usarse para instalar sistemas operativos.
- Extendida: permite crear múltiples particiones lógicas dentro de ella.
- Lógicas: se crean dentro de la extendida; útiles para guardar datos o sistemas adicionales.

**Ventajas:** mejor organización de datos, posibilidad de tener varios sistemas operativos, más seguridad ante fallos (no afecta a todo el disco).

**Desventajas:** espacio fijo (difícil de redimensionar sin herramientas), mala planificación puede desperdiciar espacio.

**b. ¿Cómo se identifican las particiones en GNU/Linux? (Considere discos IDE, SCSI y SATA).**

En GNU/Linux las particiones aparecen como archivos dentro de /dev; en discos IDE se nombran como /dev/hdX con letras para el disco y números para la partición por ejemplo /dev/hda1; en discos SCSI y SATA se nombran como /dev/sdX siguiendo el mismo criterio por ejemplo /dev/sda1

**c. ¿Cuántas particiones son necesarias como mínimo para instalar GNU/Linux? Nómbralas indicando tipo de partición, identificación, tipo de File System y punto de montaje.**

Para instalar GNU/Linux se necesitan como mínimo dos particiones una es la raíz / que se crea como partición primaria o lógica identificada como Linux generalmente con sistema de archivos ext4 y con punto de montaje en / y la otra es la swap que también puede ser primaria o lógica identificada como Linux swap no usa un sistema de archivos tradicional sino que funciona como memoria de intercambio y no tiene un punto de montaje en directorios porque el sistema la utiliza directamente como apoyo de la RAM

**d. Dar ejemplos de diversos casos de particionamiento dependiendo del tipo de tarea que se deba realizar en su sistema operativo.**

El particionamiento en GNU/Linux depende del uso si es un equipo de escritorio básico alcanza con la raíz y la swap si es para uso personal con muchos archivos conviene separar /home para guardar datos de usuario si es un servidor se recomienda separar /var para los registros y archivos que crecen mucho y en sistemas críticos también se suele separar /boot para asegurar los archivos de arranque

**e. ¿Qué tipo de software para particionar existe? Menciónelos y compare.**

En GNU/Linux se usan programas como **fdisk**, herramienta en modo texto simple y confiable **parted**, más flexible y con soporte para GPT y discos grandes y **GParted**, su versión gráfica que muestra las particiones visualmente y facilita crear mover o redimensionar en resumen fdisk es básico pero seguro parted es más completo y GParted el más amigable para el usuario

## 8.

### a. ¿Qué es el BIOS? ¿Qué tarea realiza?

El BIOS es un firmware almacenado en un chip de la placa base que se ejecuta apenas se enciende la computadora su tarea es realizar una verificación básica del hardware llamada POST y luego buscar un dispositivo de arranque para entregar el control al gestor de arranque o al sistema operativo.

### b. ¿Qué es UEFI? ¿Cuál es su función?

UEFI es el reemplazo moderno del BIOS con más capacidad y mejor compatibilidad con hardware nuevo su función es inicializar los componentes del sistema de manera más rápida y segura además de soportar discos grandes con tabla GPT y permitir interfaces gráficas y opciones avanzadas en el arranque.

### c. ¿Qué es el MBR? ¿Qué es el MBC?

El MBR o Master Boot Record es el primer sector de un disco que contiene la tabla de particiones y un pequeño código para iniciar el arranque en discos tradicionales el MBC o Master Boot Code es ese código de arranque que se encuentra en el MBR y que transfiere el control al gestor de arranque o a la partición activa.

### d. ¿A qué hacen referencia las siglas GPT? ¿Qué sustituye? Indique cuál es su formato.

GPT significa GUID Partition Table y es un esquema de particionamiento moderno que sustituye al MBR permite manejar discos de más de 2 TB y un número mucho mayor de particiones su formato está basado en identificadores globales únicos (GUID) que hacen más segura y flexible la organización de las particiones.

### e. ¿Cuál es la funcionalidad de un “Gestor de Arranque”? ¿Qué tipos existen? ¿Dónde se instalan? Cite gestores de arranque conocidos.

Un gestor de arranque es un programa que se ejecuta después del BIOS o UEFI y antes del sistema operativo su función es cargar el kernel elegido en memoria existen gestores simples que arrancan un único sistema y gestores múltiples que permiten elegir entre varios se instalan generalmente en el MBR o en la partición de arranque y algunos conocidos son GRUB LILO y systemd-boot.

**f. ¿Cuáles son los pasos que se suceden desde que se prende una computadora hasta que el Sistema Operativo es cargado (proceso de bootstrap)?**

Al encender la PC primero el BIOS o UEFI inicializa el hardware luego se ejecuta el POST que verifica los dispositivos básicos después se localiza el sector de arranque en el disco y se carga el MBR o GPT que contiene el gestor de arranque este carga el kernel en memoria y finalmente el control pasa al sistema operativo que empieza a ejecutarse.

**g. Analice el proceso de arranque en GNU/Linux y describa sus principales pasos.**

En GNU/Linux el gestor de arranque como GRUB carga el kernel en memoria junto con la imagen de inicio el kernel se encarga de detectar y configurar los dispositivos básicos y montar el sistema de archivos raíz luego se ejecuta el proceso init o systemd que inicia los servicios y demonios necesarios y finalmente se presenta la interfaz de usuario ya sea gráfica o en modo consola.

**h. ¿Cuáles son los pasos que se suceden en el proceso de parada (shutdown) de GNU/Linux?**

En el apagado primero se notifica a los procesos para que se cierren ordenadamente luego se desmontan los sistemas de archivos para evitar pérdida de datos y finalmente se envía la señal al kernel para que detenga la CPU o reinicie según corresponda todo esto lo maneja el comando shutdown o systemctl en las distribuciones modernas.

**i. ¿Es posible tener en una PC GNU/Linux y otro Sistema Operativo instalado? Justifique.**

Sí es posible se llama sistema de arranque dual o dual boot y funciona porque cada sistema puede estar en una partición diferente y un gestor de arranque como GRUB permite elegir cuál iniciar al encender la computadora de esta forma pueden convivir GNU/Linux con Windows u otros sistemas sin problemas mientras cada uno tenga su espacio en disco.

## 9. Completar

**a. ¿Cómo se identifican los archivos en GNU/Linux?**

En gnu linux un archivo se reconoce por su ruta dentro del filesystem y por su tipo real, no solo por el nombre, la ruta es la cadena que ubica al archivo dentro de directorios separados por “/”, y para saber qué es cada archivo podés usar el comando file que te dice su tipo



**b. Investigue el funcionamiento de los editores vim, nano y mcedit, y los comandos cat, more y less.**

Cat muestra el contenido de un archivo y con -n numera las líneas, more lo muestra de forma paginada para leerlo por partes, hasta ahí llega el material que tenemos

**c. Cree un archivo llamado “prueba.exe” en su directorio personal usando el vim. El mismo debe contener su número de alumno y su nombre.**

**d. Investigue el funcionamiento del comando file. Pruébalo con diferentes archivos. ¿Qué diferencia nota?**

El comando file sirve para que el sistema te diga de qué tipo es un fichero por ejemplo si es texto, binario, etc, la idea es que lo ejecute sobre distintos archivos y veas que reporta tipos distintos según el contenido y no por cómo se llame, esto encaja con que los nombres pueden ser solo cadenas y los metadatos van aparte en el filesystem del que habla el apunte

## 10.

**a. Cree la carpeta ISOCSO**

“mkdir ISOCSO”: make directory “ISOCSO”

**b. Acceda a la carpeta**

“cd ISOCSO”: change directory “ISOCSO”

**c. Cree dos archivos con los nombres isocso.txt e isocso.csv**

“touch isocso.txt” , “touch isocso.csv”

**d. Liste el contenido del directorio actual**

“ls”: list

**e. Visualizar la ruta donde estoy situado**

“pwd”: print working directory

**f. Busque todos los archivos en los que su nombre contiene la cadena “iso\*”**

“find . -name “iso\*” ”

**g. Informar la cantidad de espacio libre en disco**

“df -h”: con df solo alcanza, df -h lo hace más legible.

#### **h. Verifique los usuarios conectados al sistema**

“who”

#### **i. Editar a el archivo isocso.txt e ingresar Nombre y Apellido**

“nano isocso.txt”: parado en la carpeta ISOCSO/

#### **j. Mostrar en pantalla las últimas líneas de un archivo.**

“tail isocso.txt”: parado en la carpeta ISOCSO/

## 11.

**Investigue el funcionamiento, parámetros y ubicación (directorio) de los siguientes comandos:**

→ **man**: Muestra el manual de ayuda de un comando. Ubicación: /usr/bin/man. Ejemplo: “man ls”

→ **shutdown**: Apaga o reinicia el sistema. Ejemplo: shutdown -h now. Ubicación: /sbin/shutdown. Ejemplo: “shutdown -h now”

→ **reboot**: Reinicia la máquina. Ubicación: /sbin/reboot. Ejemplo: “reboot”

→ **halt**: Detiene el sistema. Ubicación: /sbin/halt. Ejemplo: “halt”

→ **uname**: Muestra información del sistema y kernel. Ubicación: /bin/uname. Ejemplo: “uname -a”

→ **dmesg**: Muestra mensajes del kernel (ej. detección de hardware). Ubicación: /bin/dmesg. Ejemplo: “dmesg | grep usb”

→ **lspci**: Lista dispositivos PCI conectados. Ubicación: /usr/bin/lspci. Ejemplo: “lspci”

→ **at**: Programa la ejecución de comandos en un momento determinado. Ubicación: /usr/bin/at. Ejemplo: “echo “reboot” | at now + 2 minutes”

→ **netstat**: Muestra conexiones de red y puertos abiertos. Ubicación: /bin/netstat o /usr/bin/netstat. Ejemplo: “netstat -tulnp”

→ **head**: Muestra las primeras líneas de un archivo. Ubicación: /usr/bin/head. Ejemplo: “head -n 5 /etc/passwd”

→ **tail**: Muestra las últimas líneas de un archivo. Ubicación: /usr/bin/tail. Ejemplo: “tail -f /var/log/syslog”