Practica 1

# Introducción

Dentro de esta primera practica se abordaran practicas relacionadas con temas y aplicaciones previas que hemos visto a lo largo de la carrera, así como con las previas lecturas dictaminadas por la Tarea numero 1, la cual tiene relación a los Sistemas Operativos, clasificación, funcionamientos y la historia de los mismos, así como conceptos básicos que debemos entender para poder adentrarnos de una forma correcta y mas completa al mundo de la programación y el desarrollo.  
Comenzaremos con la definición de un Sistema Operativo el cual es un programa (software) que cuando arrancamos o iniciamos el ordenador se encarga de gestionar todos los recursos del sistema informático permitiendo así la comunicación entre el usuario y el ordenador.

Los recursos que gestiona son tanto del hardware (partes físicas, disco duro, pantalla, teclado, etc.) como del software (programas e instrucciones).  
Los otros programas, que no son SO, se llaman aplicaciones.  
Resumiendo, un SO es el encargado de gestionar el software y el hardware de un ordenador o computadora.   
En su sentido más general, un SO es un software que permite a un usuario ejecutar otras aplicaciones en un dispositivo informático (ordenador, Smartphone, Tablet, etc.).  
La mayoría de la gente usa el sistema operativo que viene en su ordenador o móvil, pero es posible actualizarlo o incluso cambiar el sistema operativo por otro diferente.

## ¿Para Qué sirve el Sistema Operativo?

Los sistemas operativos utilizan imágenes y botones para poder comunicarnos con el ordenador de forma sencilla y fácil para decirle lo que queremos hacer en cada momento a nuestro ordenador.  
Su función principal es la de darnos las herramientas necesarias para poder controlar nuestra computadora y poder hacer uso de ella, de la forma más cómoda y sencilla posible.

Las funciones básicas del Sistema Operativo son administrar los recursos del ordenador, coordinar el hardware y organizar archivos y directorios en los dispositivos de almacenamiento de nuestro ordenador.

Algunas cosas más concretas que puede realizar un Sistema Operativo son:

- El sistema operativo es la única gran pieza de software que ejecuta los programas y se encarga de todo lo demás.

Por ejemplo, el sistema operativo controla los archivos y otros recursos a los que pueden acceder estos programas una vez ejecutados.

- Cuando tenemos múltiples programas que se pueden ejecutar al mismo tiempo, el sistema operativo determina qué aplicaciones se deben ejecutar en qué orden y cuánto tiempo.

- Gestiona el intercambio de memoria interna entre múltiples aplicaciones.

- Se ocupa de la entrada y la salida de los datos desde y hacia los dispositivos de hardware conectados, tales como discos duros, impresoras, puertos de comunicación, teclado, etc.

Tipos de Sistemas Operativos

Ha nivel de usuario de un PC, podemos decir que hay dos tipos de Sistemas Operativos: SO para PC y SO para Teléfonos Móviles.  
Pero también hay una amplia familia de sistemas operativos categorizados según los tipos de ordenadores que controlan y el tipo de aplicaciones que admiten.  
Las 4 categorías para este tipo de SO son: **SO en tiempo real, SO Usuario Único Tarea Única, SO Usuario Único Multitarea y SO Multiusuario.**- Sistema operativo en tiempo real (RTOS): los sistemas operativos en tiempo real se utilizan para controlar maquinaria, instrumentos científicos y sistemas industriales.Una parte muy importante de un RTOS es administrar los recursos de la computadora para que una operación particular se ejecute exactamente en la misma cantidad de tiempo, cada vez que ocurre.

- Usuario único, tarea única: como su nombre lo indica, este sistema operativo está diseñado para administrar la computadora de modo que un usuario pueda hacer una cosa a la vez.

- Usuario único, multitarea: este es el tipo de sistema operativo que la mayoría de la gente usa en sus computadoras de escritorio y portátiles en la actualidad.

Windows de Microsoft y las plataformas MacOS de Apple son ejemplos de sistemas operativos que permitirán que un solo usuario tenga varios programas en funcionamiento al mismo tiempo.  
- Multiusuario: un sistema operativo multiusuario permite que muchos usuarios diferentes aprovechen los recursos de la computadora simultáneamente.

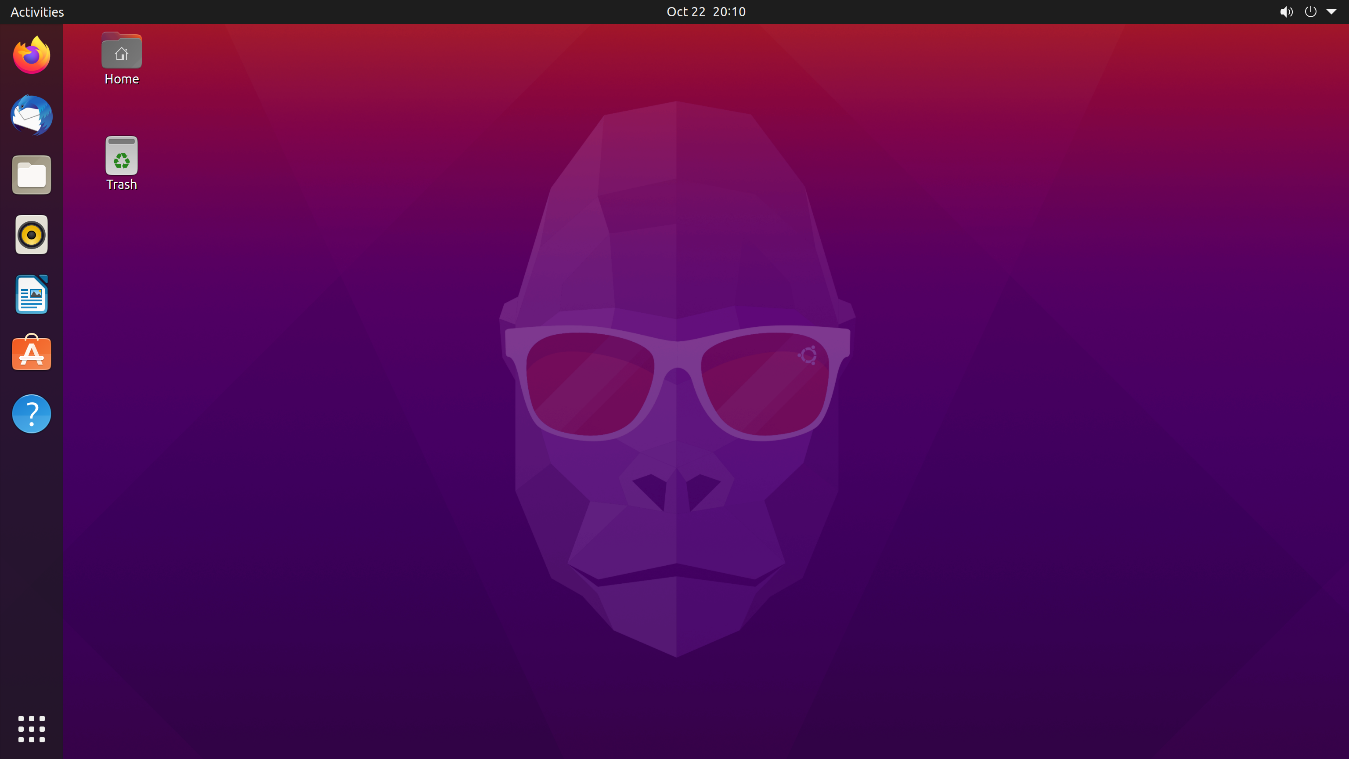
# Desarrollo Experimental

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.

## Sección Linux:

1. Verifique la existencia del sistema operativo Linux y su correcto funcionamiento con el entorno gráfico (si el entono 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.

Estamos usando la distribución de Ubuntu en su versión 20.04 ya que es el OS basado en Linux más utilizadas por la facilidad de uso además de ser amigable para principiantes en su instalación y uso

Algunas otras distribuciones de linux son:

Fedora: Distribución gratuita creada y mantenida por la empresa Red Hat que utiliza el sistema de paquetería RPM (Red Hat Package Manager). Tiene tres versiones diferentes para escritorio, servidores y sistemas en la nube, y destaca por su seguridad gracias al sistema SELinux ("Security-Enhanced Linux")

Arch Linux: Una distribución modular en la que empiezas desde cero y tienes que ir añadiéndole los componentes que quieras. No es muy apta para principiante, y utiliza pacman, su propio gestor de paquetes. Se trata de una Rolling Release, lo que quiere decir que todos sus componentes van actualizándose sin necesidad de instalar versiones nuevas del sistema operativo.

Linux Mint: Basado en Ubuntu, es uno de los más recomendados para todos aquellos que tocan Linux por primera vez. Su entorno de escritorio, Cinnamon, tiene muchas similitudes con el de Windows, y es también uno de los más personalizables

Peppermint OS: Distribución rápida y ligera basada en Ubuntu con entorno de escritorio LXDE. Utiliza la tecnología Prism de Mozilla para integrarse con las aplicaciones basadas en la nube, utilizando las webapps como si fueran nativas. Se presenta como una alternativa a otros sistemas basados en la nube como Chrome OS

Kali Linux: Distribución basada en Debian con una inmensa colección de herramientas para proteger nuestros equipos. Utiliza un kernel personalizado con parches de seguridad y tiene soporte para la arquitectura ARM.

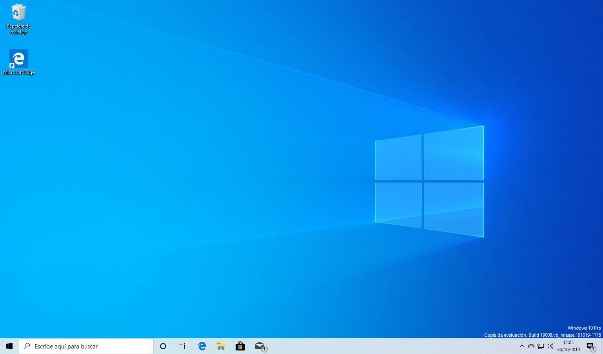
Puppy Linux: Una minúscula distribución que puede llevarse en un USB o CD, pero sorprendentemente rápida al cargarse enteramente en la memoria RAM del ordenador. Se carga en 30 o 40 segundos y ocupa sólo 100 MB.

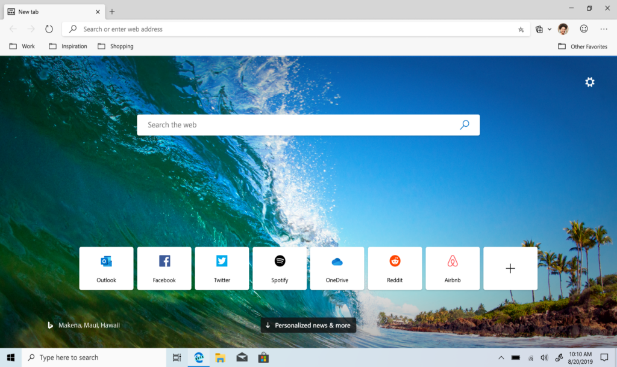
Debian: Muy estable y 100% libre, Debian destaca por su sistema de paquetería .deb y su gestión de paquetes APT. Es una de las distribuciones más importantes de GNU/Linux, ya que en ella se basan gigantes como Ubuntu.

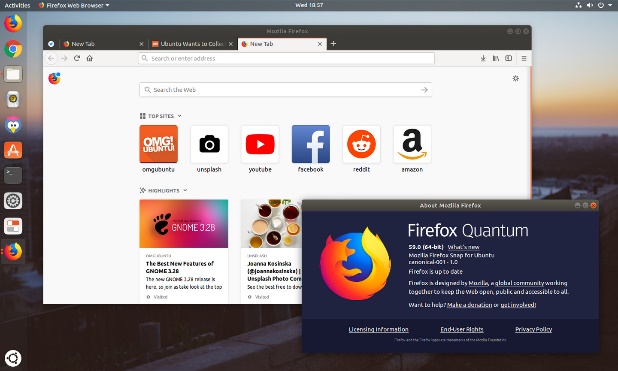
Red Hat Enterprise Linux: Distribución comercial de Linux desarrollada por Red Hat. Ofrece una estabilidad y flexibilidad punteras, lo que la coloca como una de las más recomendadas para empresas y servidores.

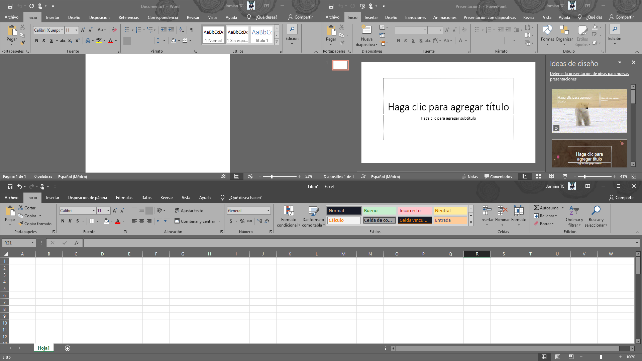
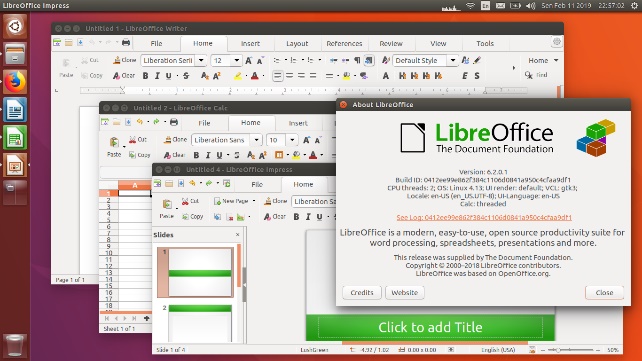
CentOS: Nació como un derivado gratuito de la distribución comercial Red Hat Enterprise Linux (RHEL) destinada al uso empresarial. Recientemente unió las fuerzas con el propio Red Hat, y sigue siendo una apuesta segura para los que busquen un código de gran calidad.

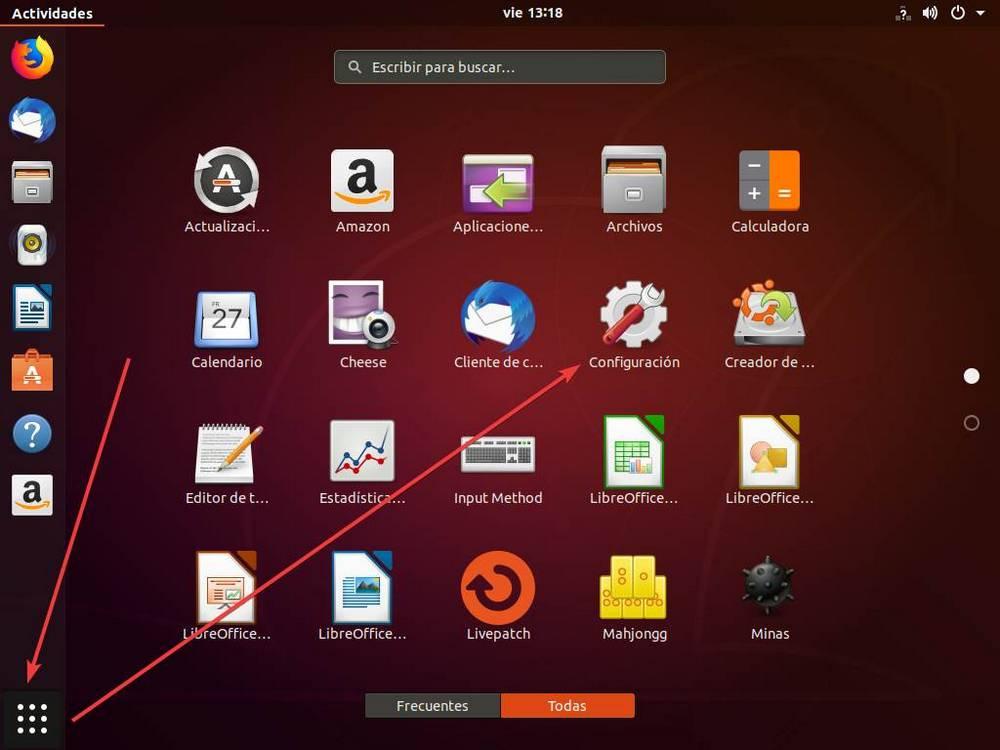
1. Revise el entorno Linux e identifique los componentes de menú con los que cuenta su sistema operativo. Entre en cada uno de los componentes del menú y explore algunas de sus funcionalidades. Reporte que funcionalidades encontró, además haga un comparativo de estas funcionalidades con respecto a las existentes en el sistema operativo Windows que usted utilice.

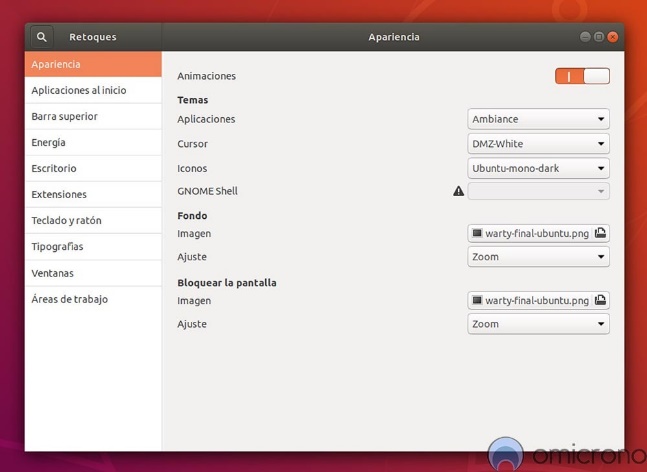
Las diferencias con los sistemas son bastantes, tanto que a simple vista cambia completamente el entorno, principalmente algunas características que debemos notar es que las barras de tarea están ubicadas y distribuidas de manera totalmente diferente

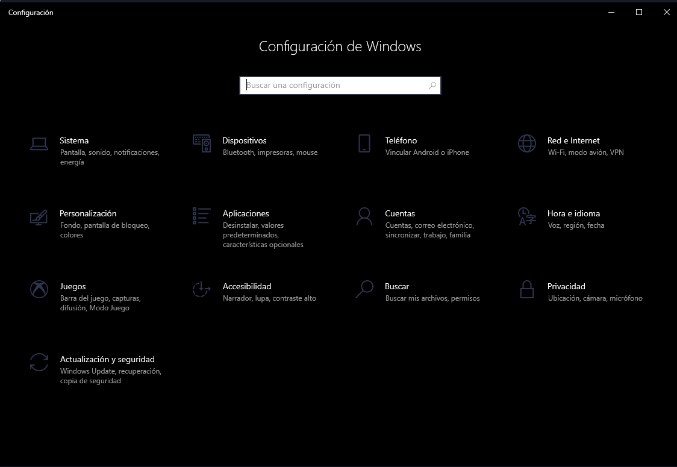
Viene instalado el explorador de internet Mozilla Firefox por defecto mientras que en Windows actualmente se opta por el nuevo Microsoft Edge



Las paqueterías de oficinas son diferentes en Ubuntu tenemos libre office mientras que en Windows tenemos Microsoft office

Los menús donde tenemos las aplicaciones son distintos el de Ubuntu recuerda un poco a lo que fue Windows 8 al tener un menú que se extiende por toda la pantalla

Los menús de configuraciones tienen una distribución distinta entre sí, siendo Ubuntu un poco más lite que Windows en el sentido de que no tiene tantos apartados

Otras características diferenciables es que sus paletas de colores, y distribuciones en distintas aplicaciones son tan opuestas en uno a la otra, los gestores de archivos, la cantidad de parámetros configurables en ambos sistemas y el software tan orientado a cada sistemas los hace bastante únicos además de que un gran punto a favor de Ubuntu es que es totalmente gratuito, a diferencia de Windows de que si queremos tener todas las funcionalidades desbloqueadas como personalización tenemos que adquirir una licencia

Otras diferencias que no son notables a nivel visual pero si internamente son que el código de Ubuntu es open source esto quiere decir que cualquiera puede modificarlo a su antojo sin ninguna restricción, las actualizaciones se hacen de manera manual, es decir cada cierto tiempo tienes que reinstalar Ubuntu a la nueva versión, para ello hay versiones que duran un año completo evitando estos problemas, tiene una mejor seguridad con respecto a Windows ya que su código desde sus cimientos fue creado atendiendo principios de seguridad y la optimización en Ubuntu es enormemente mayor a otros OS siendo más ligero y rápido al abrir aplicaciones o procesamiento de programas pesados

1. Abra una sesión de consola (Terminal) y reporte para que se utilizan cada uno de los siguientes comandos:

|  |  |  |
| --- | --- | --- |
| ls: | cd: | cp: |
| chmod: | cat: | mv: |
| vi: | grep: | mkdir: |
| pwd: | rm: | rmdir: |
| clear: | ps: | whoami: |

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.**

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

ls cd nombre\_directorio cp [ruta1]archivo1 [ruta2][archivo 2]

ls –l cat nombre\_archivo mv [ruta1]archivo1 [ruta2][archivo 2]

ls –la ls -la | more mkdir nombre\_directorio

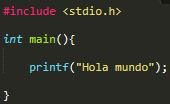
pwd rm nombre\_archivo rmdir nombre\_directorio

clear Ps whoami

|  |  |  |
| --- | --- | --- |
| ls: | cd nombre\_directorio: | cp [ruta1]archivo1 [ruta2][archivo 2]: |
| ls -l | cat nombre\_archivo: | Mv [ruta1]archivo1 [ruta2][archivo 2]: |
| ls -la | ls -la | more: | mkdir nombre\_directorio: |
| pwd: | rm nombre\_archivo: | rmdir nombre\_directorio: |
| clear: | ps: | whoami: |

Observe los resultados de la ejecución de cada comando.

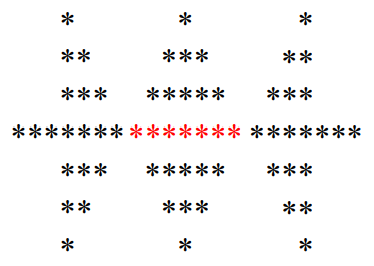
1. Algunos de los comandos anteriores pueden tener una serie de opciones para su ejecución (p.e. ls), utilice nuevamente el comando man nombre\_comando para ver detalles de información sobre cada uno de los comandos anteriores. Observe las diversas opciones que se utilizan en dichos comandos.
2. Abra 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.

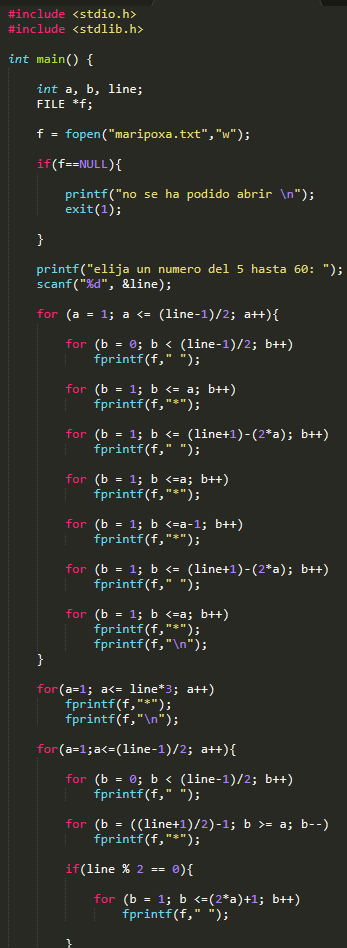
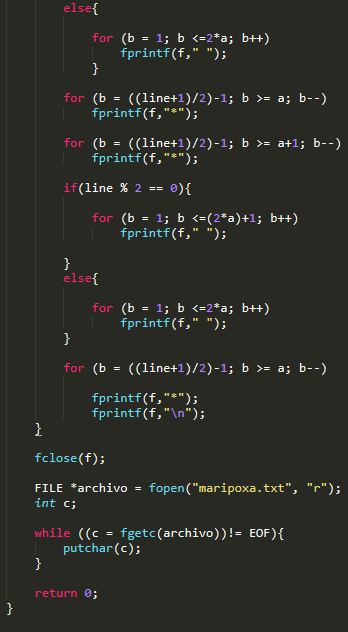
Código del programa

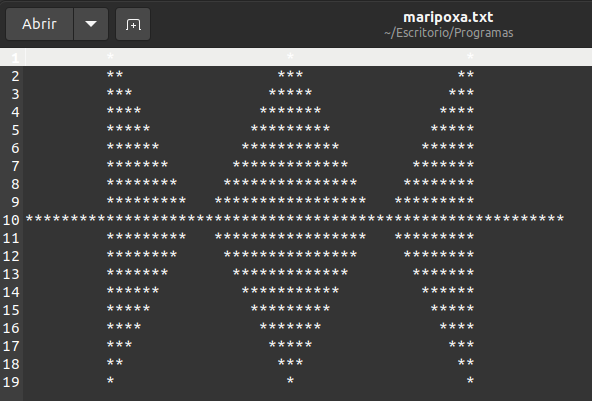
Ejecución en Ubuntu

1. Realice un programa que muestre la siguiente salida:

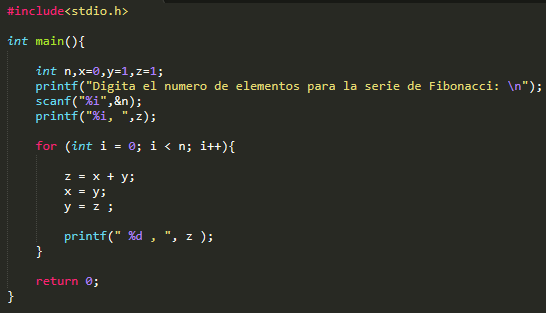
El dato de entrada al programa es el número de asteriscos de la línea horizontal media en la figura con la mayor cantidad de ellos (es decir la figura del rombo), en este caso el dato de entrada es 7 (línea horizontal de asteriscos en rojo). Este dato se pedirá al usuario y podrá ser cualquier valor 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.

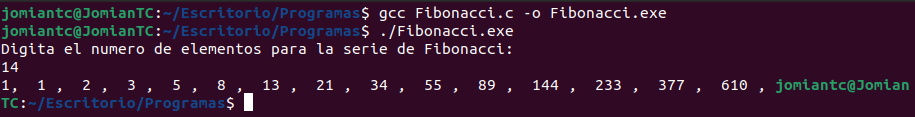


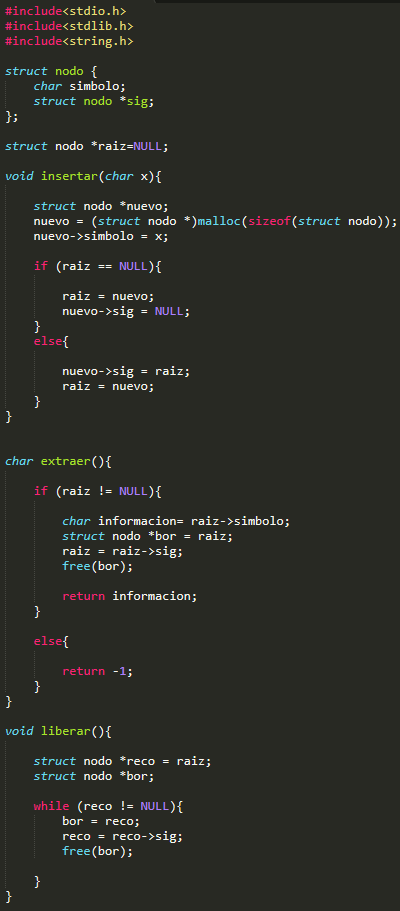
Código del programa

Ejecución en Ubuntu y fichero creado

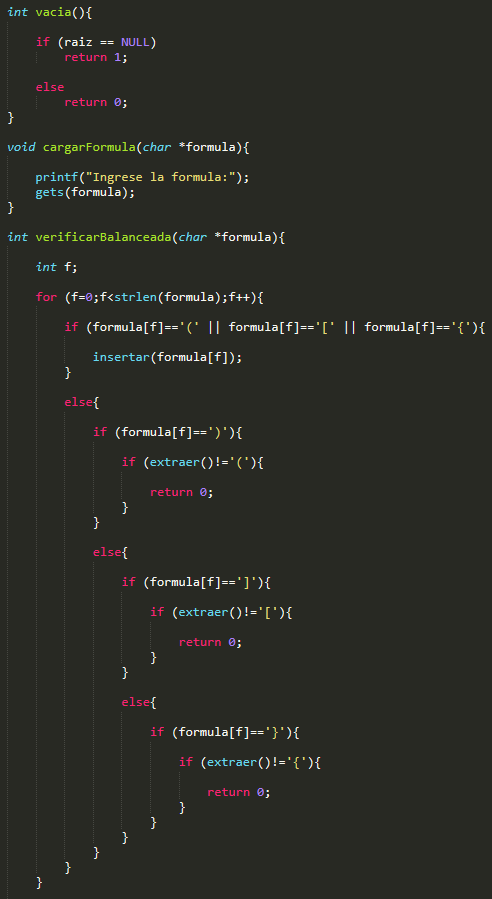
1. Realice un programa que calcule la serie de Fibonacci de un número dado como entrada. Compile y ejecute su programa.

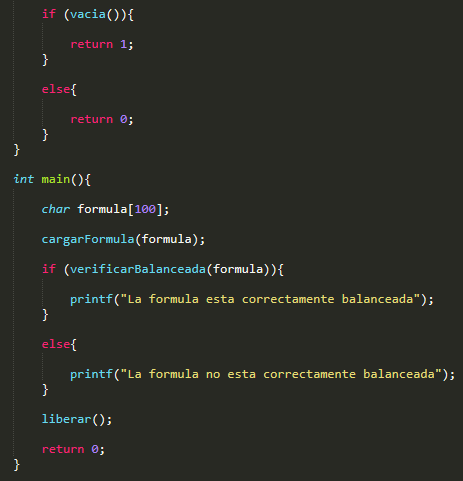
Código del programa

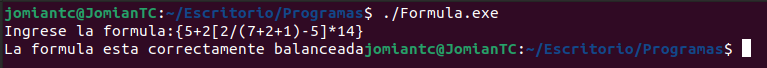
Ejecución en Ubuntu

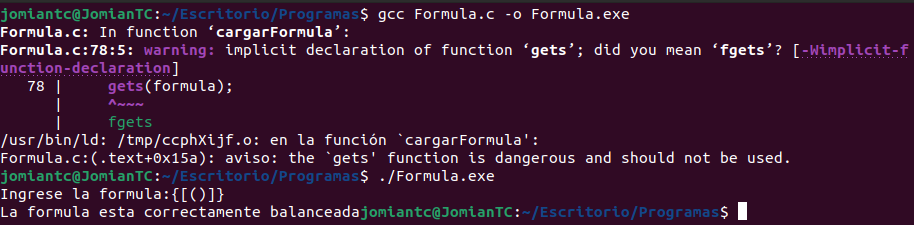
1. Realice un programa que determine si un grupo de paréntesis están balanceados utilizando pilas. Deberán considerarse tres tipos de paréntesis: ( ), { }, [ ]

Código del programa

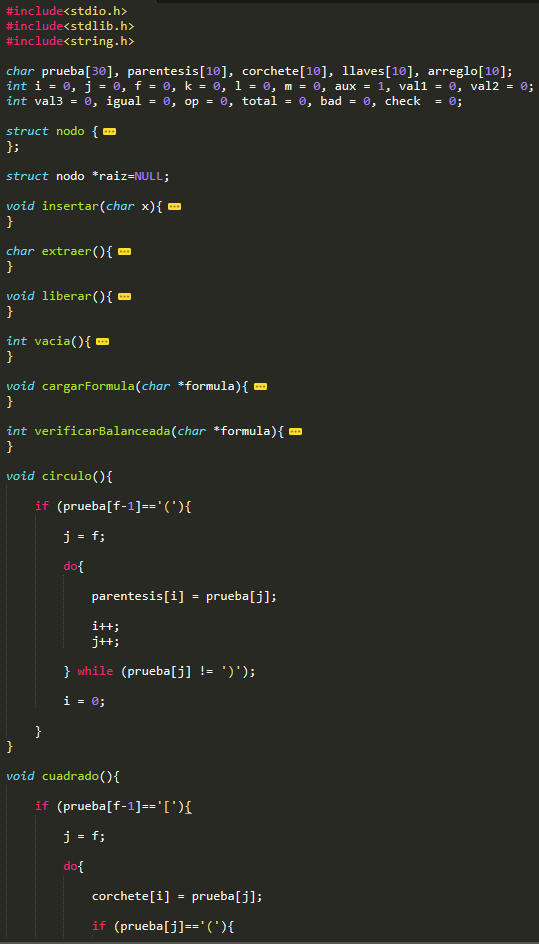


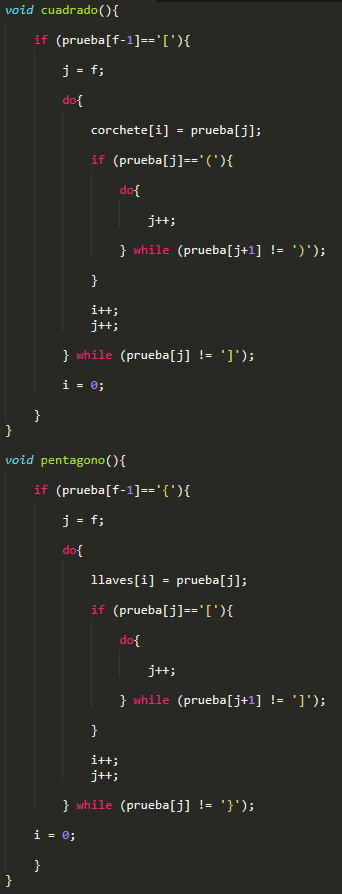
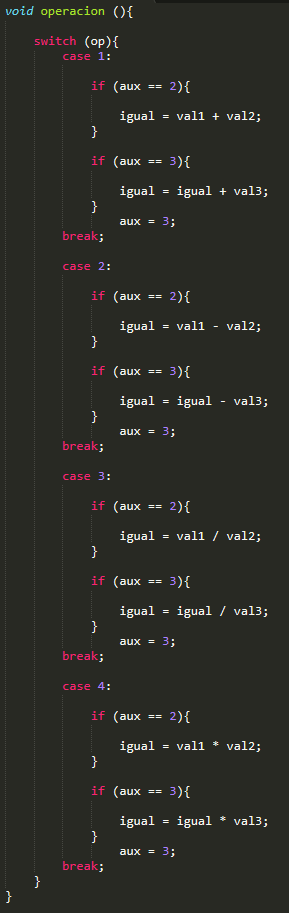


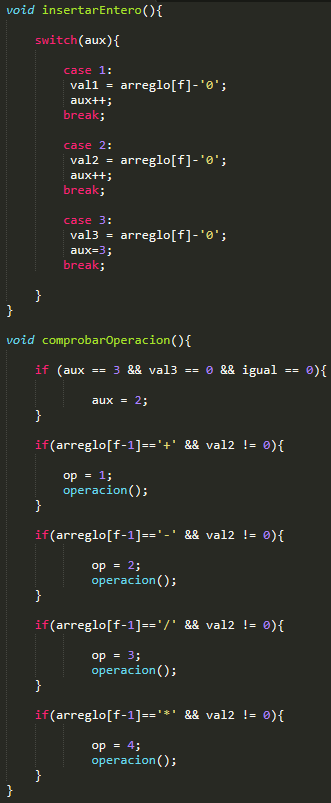
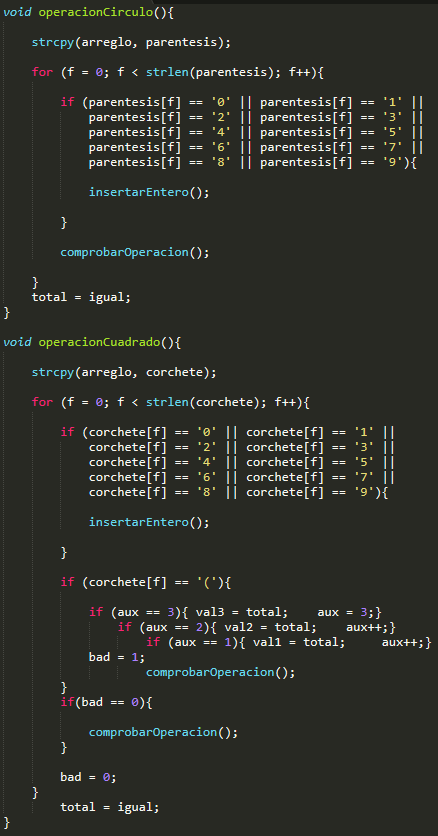
Ejecución en Ubuntu

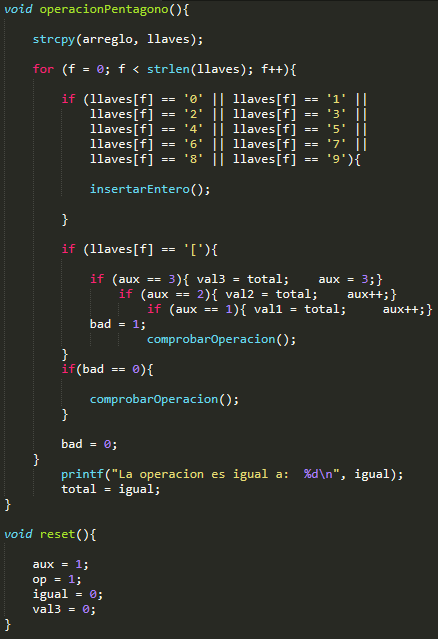


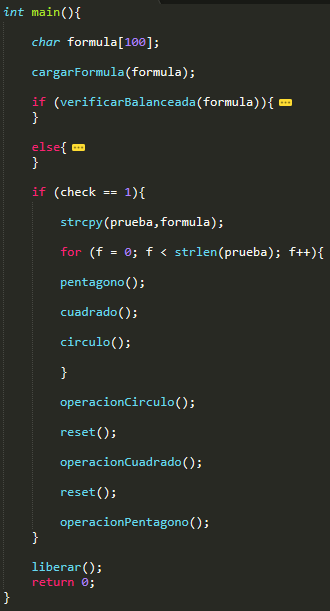
1. Realice un programa que evalúe una expresión aritmética delimitada completamente con paréntesis balanceados (por ejemplo: (((5-2)+4)\*10). Compile y ejecute su programa. Restricción: No utilizar transformación a notación prefija o posfija.

Código del programa



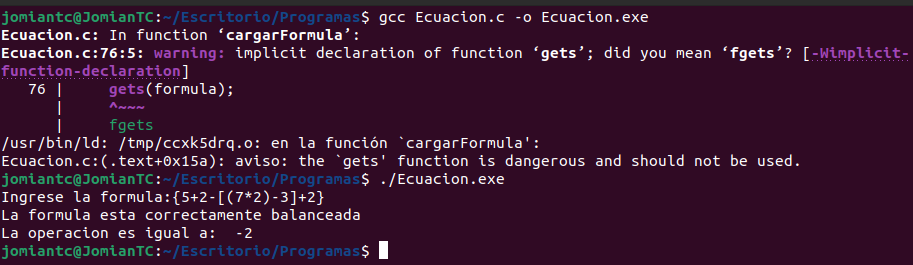


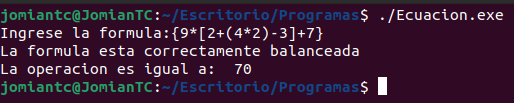




Las funciones que fueron minimizadas con el editor de texto son las mismas que el ejercicio anterior fueron cortadas con el fin de reducir el tamaño del archivo

El esquema resumido del código sin la comprobación de balance sería el siguiente

Ejecución en Ubuntu



1. 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).
4. Reporte la función de cada uno de los anteriores comandos.

**DIR**

El comando dir imprime la unidad en la que estas alojado

El número de serie del volumen de la unidad

Y también dependiendo de la ubicación donde estemos alojados por ejemplo en Desktop nos desplegara todos los archivos que tenemos en dicho directorio incluyendo la fecha en la que fue creado, el tamaño del archivo en bytes, el nombre del archivo además de su extensión

**IPCONFIG**

El comando IPCONFIG imprime todas las conexiones activas en ese momento por ejemplo tenemos la conexión de Ethernet es decir la conexión de internet por cable, las conexiones inalámbricas lan es decir conexiones en local, la conexión con nuestro Wi-Fi que imprime nuestra dirección IPv4, las mascara de subred y la puerta de enlace, también muestra la conexión de la red Bluetooth

**CLS**

Este comando limpia nuestra ventana de comandos dejando solo la dirección de la carpeta en la que estamos

**VER**

imprime la versión actual de Microsoft Windows

**TREE**

imprime una estructura de árbol de las carpetas en el directorio donde estamos alojados

**CD - Nombre\_Directorio**

Nos permite acceder a la carpeta del nombre del directorio que hayamos seleccionado

**TYPE - Nombre\_Archivo**

Nos permite ver el contenido del fichero que hayamos seleccionado, siempre y cuando sea un fichero plano (un archivo de texto, por ejemplo)

**MKDIR - Nombre\_Directorio**

Crea una carpeta en el directorio donde nosotros queramos asignándole un nombre

**RMDIR - Nombre\_Directorio**

Elimina la carpeta en el directorio que hayamos elegido con su contenido

**DEL - Nombre\_Archivo**

Elimina el archivo en el directorio donde estas o el contenido si el nombre que escribiste es una carpeta

**COPY - Nombre\_Archivo Nombre\_NuevoArchivo**

Sirve para copiar un archivo y ponerle un nuevo nombre al archivo copiado

**REN - Nombre\_Archivo Nombre\_NuevoArchivo**

Renombra el archivo con un nuevo nombre que nosotros elegimos

**CHDIR - Nombre\_Directorio**

Tiene la misma función que CD permite ir o imprimir la dirección que queremos o estamos respectivamente

**ECHO - Cadena**

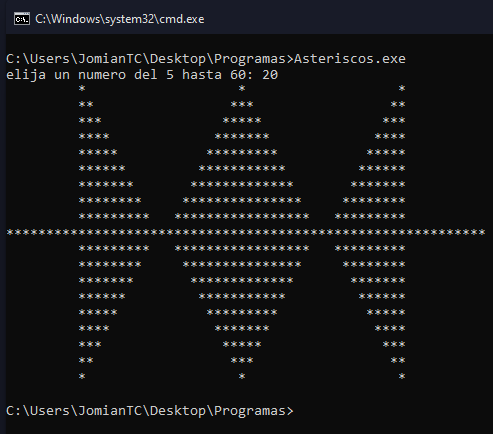
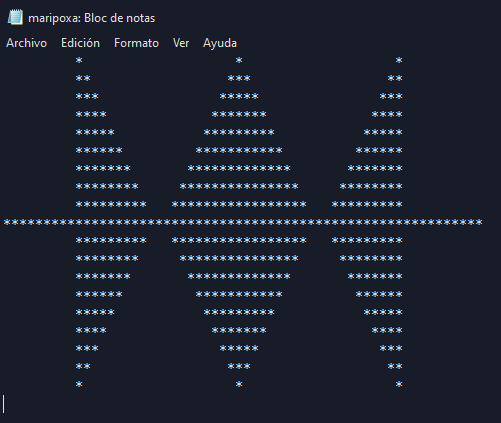
Imprime la cadena que nosotros escribimos después del comando o también lo podemos desactivar para quitar nuestra barra de dirección actual

**FIND - Cadena\_Buscar Nombre\_Archivo**

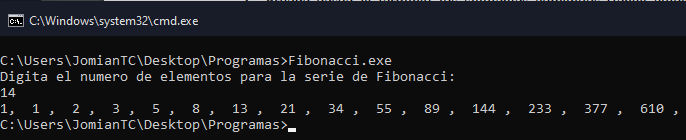
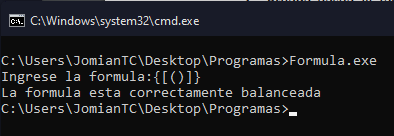
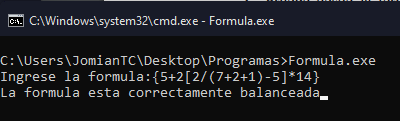
Imprime la línea en la cual se encuentra el texto que nosotros queremos buscar en el archivo que especificamos

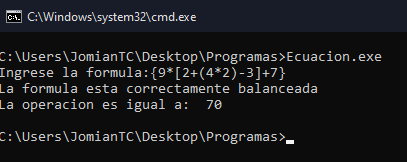
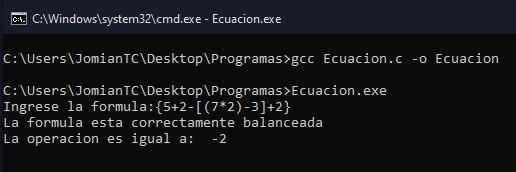
1. Abra la carpeta de su memoria usb donde tiene los programas que desarrolló en la sección de Linux y ejecútelos. Dé sus observaciones sobre la ejecución de los mismos.
2. Busque el directorio donde está instalado Dev C.
3. Desde la consola ubíquese en el directorio de Dev C, y cambie al directorio “bin”
4. 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.

Programa Hola mundo

Programa Asteriscos

Programa Fibonacci

Programa Balance de corchetes

Programa Expresiones aritméticas

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

Como pudimos observar las diferencias entre estos sistemas en cuanto a la compilación y ejecución de los programas anteriormente escritos no fue nada diferente a excepción de que la compilación en Ubuntu dio mensajes de avisos por funciones que no pertenecían al compilador de C si no al de C++ pero que de igual forma funcionan en C sin ningún problema, nada mas resulto diferente, los programas arrojaron los mismos resultados en igualdad de circunstancias, además de que en Ubuntu es mas sencillo compilar ya que tiene un compilador integrado para lenguaje C cosa que en Windows no, en cuanto los comandos sus objetivos son los mismo pero la forma de escribirlos es muy diferente, por la costumbre de usar Windows como sistema operativo principal al momento de escribir estos comandos es algo nuevo y tuvimos muchos fallos de escritura al momento de ejecutarlos, pero no hay mucha diferencia en los comandos con excepción de la sintaxis que como es de gran obviedad es muy distinta siendo ambas igualmente fáciles de aprender y de utilizar

# Conclusiones

Me parece que la practica asignada presento un buen reto para el equipo y a nivel personal, pues me hizo volver a temas que había dejado un poco abandonados por cuestiones académicas de tener que comenzar a manejar otro tipo de lenguajes de programación, de igual forma esta practica me hizo darme cuenta del enorme panorama que hay presente con Linux así como la gran diferencia que tiene con Windows, pues personalmente he estado trabajando siempre con Windows y ya estoy mucho más familiarizado con dicho Sistema Operativo, aunque me agrado mucho como es que Linux te proporciona un manual integrado en su misma consola que te proporciona información acerca de los distintos comandos que es posible usar, la realización del código presentó en ciertos momentos algunas trabas, principalmente de lógica, lo cual hizo que el equipo se esforzara aún más para poder proporcionar la salida esperada y por último la lectura de los documentos proporcionados para la elaboración de los mapas me pareció de suma utilidad, pues me proporcionan más conocimiento acerca del equipo con el cual trabajo día a día y el cual en un futuro seguirá siendo mi principal herramienta de trabajo, es por eso que me parece muy importante tener conocimiento del funcionamiento interno que este posee.

***-Mora Ayala José Antonio***