



Instituto Politecnico Nacional

Escuela Superior de Cómputo



Práctica 1

Introducción al sistema operativo Linux y Windows

Sistemas Operativos

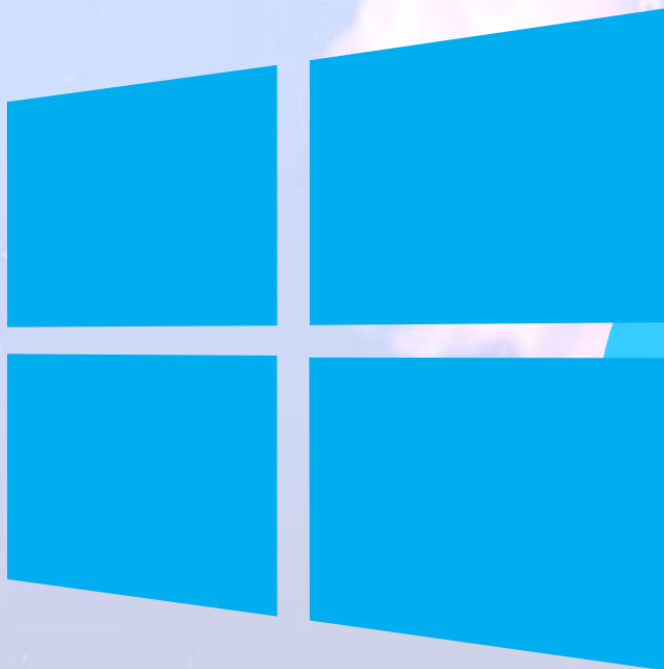
Grupo: 2CM12

Integrantes:

Baldovinos Gutiérrez Kevin
Bocanegra Heziquio Yestlanezi
Castañares Torres Jorge David
Hernández Hernández Rut Esther

Profesor

Jorge Cortes Galicia



Índice

Introducción	3
Sistema operativo	3
Windows	4
Linux	4
Competencias	5
Desarrollo.	5
Sección Linux	5
Sección Windows	23
Conclusión	39

Imagen 1 Linux desde VirtualBox	
Imagen 2 Versión de Linux desde VirtualBox	
Imagen 3 Entorno de Linux desde VirtualBox	
Imagen 4 VirtualBox Comando	
Imagen 5 VirtualBox comandos	
Imagen 6 VirtualBox comandos 1	
Imagen 7 VirtualBox comando man	
Imagen 8 VirtualBox "Hola mundo"	
Imagen 9 VirtualBox "Hola mundo" 1	
Imagen 10 VirtualBzox programa asteriscos	
Imagen 11 VirtualBox código fibonacci	
Imagen 12 CMD comando Dir	
Imagen 13 CMD comando ipconfig	
Imagen 14 CMD comando Cls	
Imagen 15 CMD comando ver	
Imagen 16 CMD comando tree	
Imagen 17 CMD comando Cd	
Imagen 18 CMD comando Type	
Imagen 19 CMD comando Mkdir	
Imagen 20 CMD comando rmdir	
Imagen 21 CMD comando Copy	
Imagen 22 CMD comando ren	
Imagen 23 CMD comando Chdir	
Imagen 24 CMD comando find	
magen 25 CMD Código asteriscos en Windows	
Imagen 26 CMD serie de Fibonacci desde windows	

Introducción

Sistema operativo

Los sistemas operativos son, hoy más que nunca, activos estratégicos de primer orden que pueden encumbrar a una firma (Apple con iOS, Samsung con Android) o condenarla al ostracismo o la venta (Nokia con su falta de reflejos con Symbian, o Blackberry y los errores estratégicos con BB OS).

En la practica 1 nos enfocaremos en hacer la comparación de dos sistemas operativos los cuales serán Windows y Linux.

Es el software que se sitúa entre la máquina y los programas. Básicamente su función es administrar los recursos del sistema.



El sistema operativo es el software (programa o conjunto de programas) que en un sistema informático gestiona los recursos de la máquina y provee servicios básicos a los programas de aplicación. El sistema operativo siempre se ejecuta en modo privilegiado.

Windows

Windows es un sistema operativo desarrollado por la compañía Microsoft. Consiste en un software conformado por un conjunto de programas que permiten gestionar y controlar el funcionamiento de las partes de un ordenador, como la memoria, el disco de almacenamiento y los dispositivos periféricos, y la ejecución de otros programas y aplicaciones.



El software se acciona con el encendido del hardware, es decir, de un ordenador o dispositivo. Una vez iniciada la sesión, el usuario puede realizar múltiples tareas y acciones en el ordenador a través del sistema operativo, como el manejo de diferentes programas o la instalación de nuevos dispositivos.

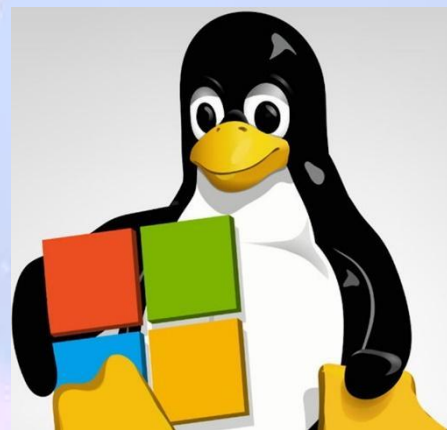
La palabra Windows, que proviene del inglés y significa *ventanas*, alude a la estructura del software que permite visualizar múltiples contenidos (como programas y archivos) organizados en compartimentos o ventanas diferentes.

El hecho de organizar los contenidos en diferentes ventanas permite que el usuario pueda visualizarlas en simultáneo, ya sea en formato de mosaico, minimizadas o superpuestas. Es decir, la interfaz de las ventanas permite ver y ejecutar varias acciones a la vez, como un procesador de datos (archivo de Word), un reproductor de video o un editor de gráficos, entre muchos otras.

Linux

Linux es un sistema operativo semejante a Unix, de código abierto y desarrollado por una comunidad, para computadoras, servidores, mainframes, dispositivos móviles y dispositivos embebidos. Es compatible con casi todas las principales plataformas informáticas, incluyendo x86, ARM y SPARC, por lo que es uno de los sistemas operativos más soportados.

Cada versión del sistema operativo Linux gestiona los recursos de hardware, lanza y gestiona las aplicaciones, y proporciona alguna forma de interfaz de usuario. La enorme comunidad de desarrollo y la amplia gama de distribuciones significa que una versión de Linux está disponible para casi cualquier tarea, y Linux ha penetrado en muchas áreas de la informática.



Competencias

El alumno aprende a familiarizarse con los sistemas operativos Linux y Windows mediante la exploración de sus ambientes de trabajo para desarrollar aplicaciones en lenguaje C. El alumno analiza el sistema operativo Linux y Windows mediante la comparación de sus características principales para diferenciarlos en su ambiente de trabajo.

Desarrollo.

Sección Linux

1. Verifique la existencia del sistema operativo Linux y su correcto funcionamiento con el entorno gráfico (si el entorno gráfico no funciona inicie sesión en modo texto).

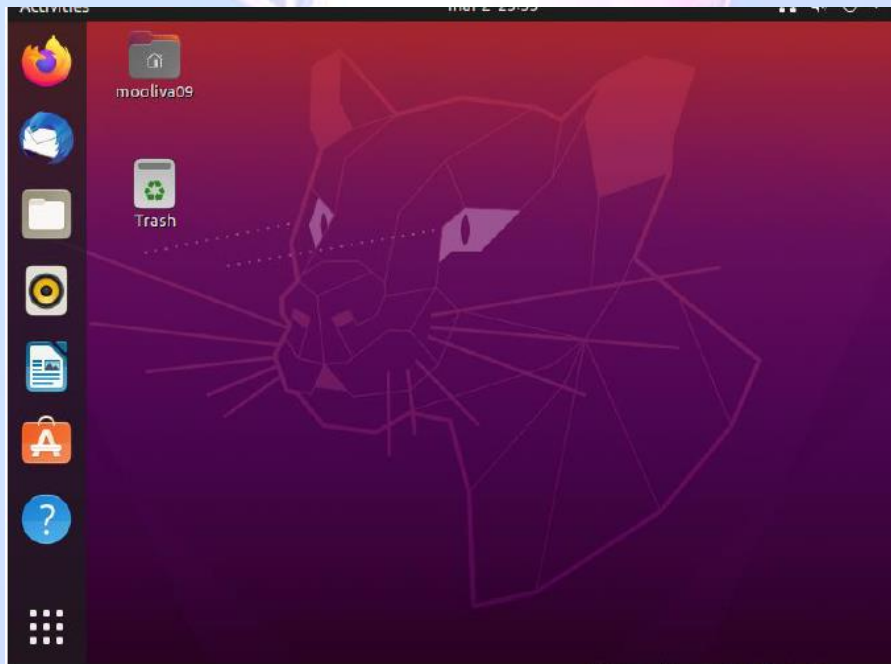


Imagen 1 Linux desde VirtualBox

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.



```
mooliva09@mooliva09-VirtualBox:~$ uname -a
Linux mooliva09-VirtualBox 5.8.0-44-generic #50~20.04.1-Ubuntu SMP Wed Feb 10 2
1:07:30 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
```

Imagen 2 Versión de Linux desde VirtualBox

Debian, Ubuntu, Linux Mint, Red Hat, Fedora, CentOS. Dependiendo de lo que usemos o nuestras necesidades nos servirán más unas u otras, ya que hay distribuciones que están más enfocadas a servidores o al usuario final.

3. Revise el entorno Linux e identifique los componentes de menú con los que cuenta su sistema operativo. Entre en cada uno de los componentes del menú 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.



The screenshot shows the Linux desktop environment with the Unity interface. The top panel includes the Dash menu (Activities) and the Dash to Search window. The Dash menu is open, showing a search bar and a list of applications. The Dash to Search window is also open, displaying a search bar and a list of applications. The desktop background is a dark blue and white pattern.

```
mooliva09@mooliva09-VirtualBox:~$ uname -a
Linux mooliva09-VirtualBox 5.8.0-44-generic #50~20.04.1-Ubuntu SMP Wed Feb 10 2
1:07:30 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
```

Imagen 2 Versión de Linux desde VirtualBox

Debian, Ubuntu, Linux Mint, Red Hat, Fedora, CentOS. Dependiendo de lo que usemos o nuestras necesidades nos servirán más unas u otras, ya que hay distribuciones que están más enfocadas a servidores o al usuario final.

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

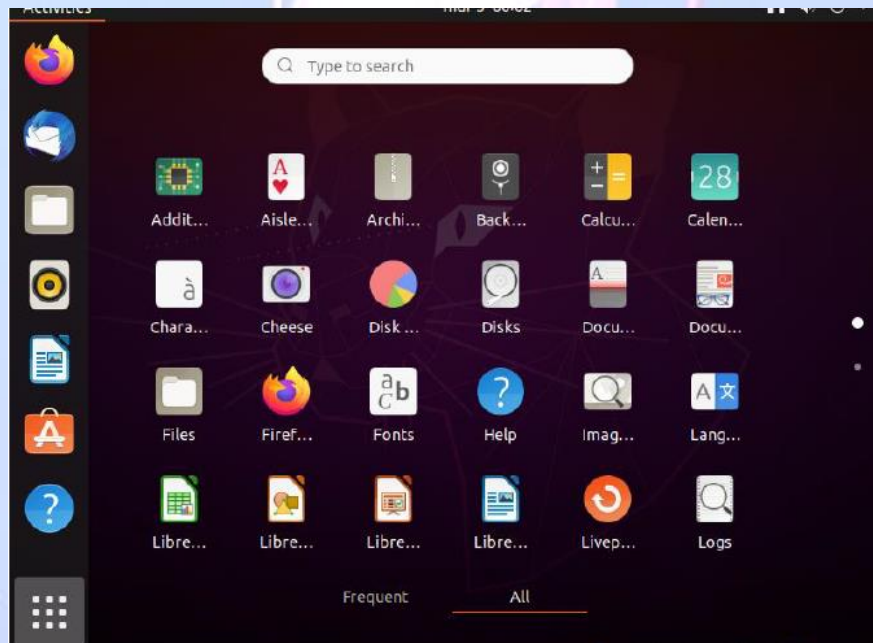


Imagen 3 Entorno de Linux desde VirtualBox

Funciones y características:

- Tanto ubuntu como Windows ya tienen un explorador de internet predeterminado
- Ubuntu ya viene con un editor de office de uso libre, en Windows podemos instalar el mismo editor o comprar office.

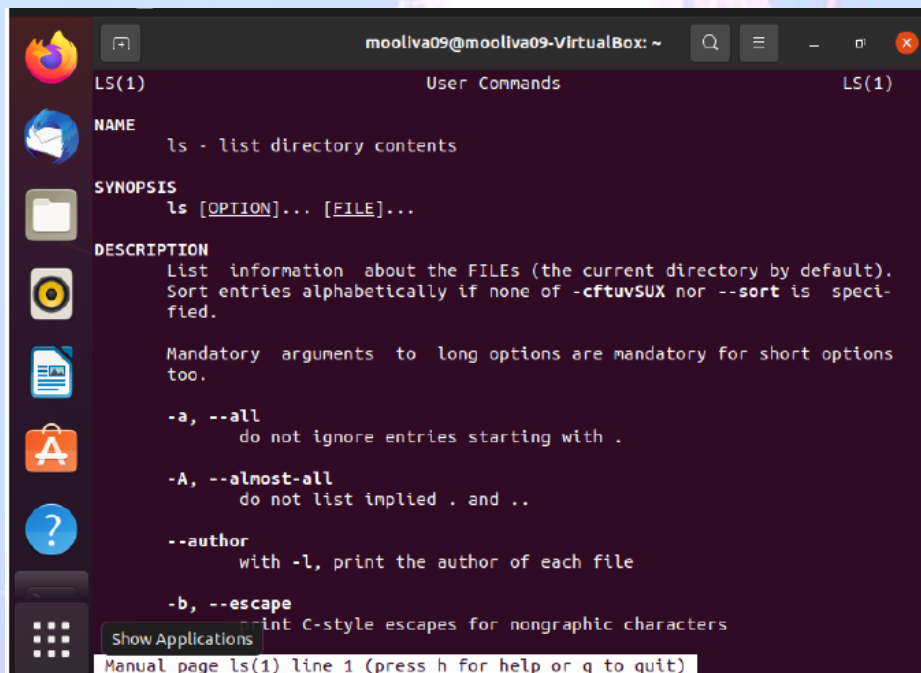
-Ubuntu trae preinstalado 35 aplicaciones para empezar con el uso del sistema de entre las cuales muchas son de software libre, mientras que en windows son desarrolladas exclusivamente para el sistema

4. 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**.



```
mooliva09@mooliva09-VirtualBox: ~
LS(1)                                User Commands                                LS(1)
NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default).
  Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
  fied.

  Mandatory arguments to long options are mandatory for short options
  too.

  -a, --all
      do not ignore entries starting with .

  -A, --almost-all
      do not list implied . and ..

  --author
      with -l, print the author of each file

  -b, --escape
      print C-style escapes for nongraphic characters

Show Applications
Manual page ls(1) line 1 (press h for help or q to quit)
```

Imagen 4 VirtualBox Comando

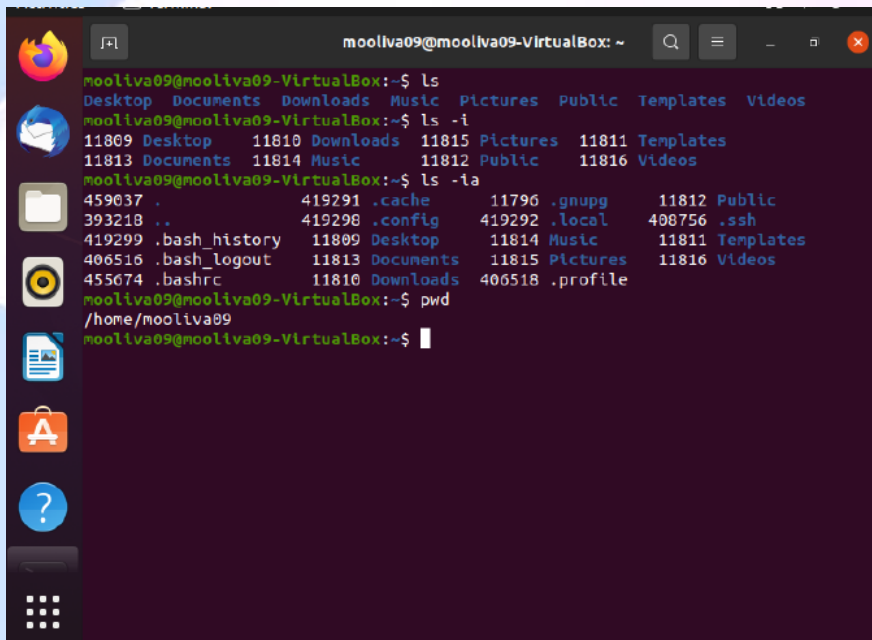
- **ls:** Muestra los archivos en el directorio
- **cd:** Función para moverse entre carpetas
- **cp:** Copiar archivos y directorios

- **chmod:** Cambiar modo de bits de un archivo
- **cat:** Concatena archivos y muestra una salida estándar
- **mv:** mover y renombrar archivos
- **vi:** Editor de archivos vim
- **grep:** Imprimir líneas que coincidan con los patrones
- **mkdir:** Crea directorios
- **pwd:** Imprime el nombre del directorio actual o de trabajo
- **rm:** Quitar (Borrar) archivos o directorios
- **rmdir:** Quitar (Borrar) directorios vacíos
- **clear:** Limpiar la pantalla de la terminal
- **ps:** Muestra una instantánea de los procesos actuales
- **whoami:** Imprimir ID del usuario efectivo

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

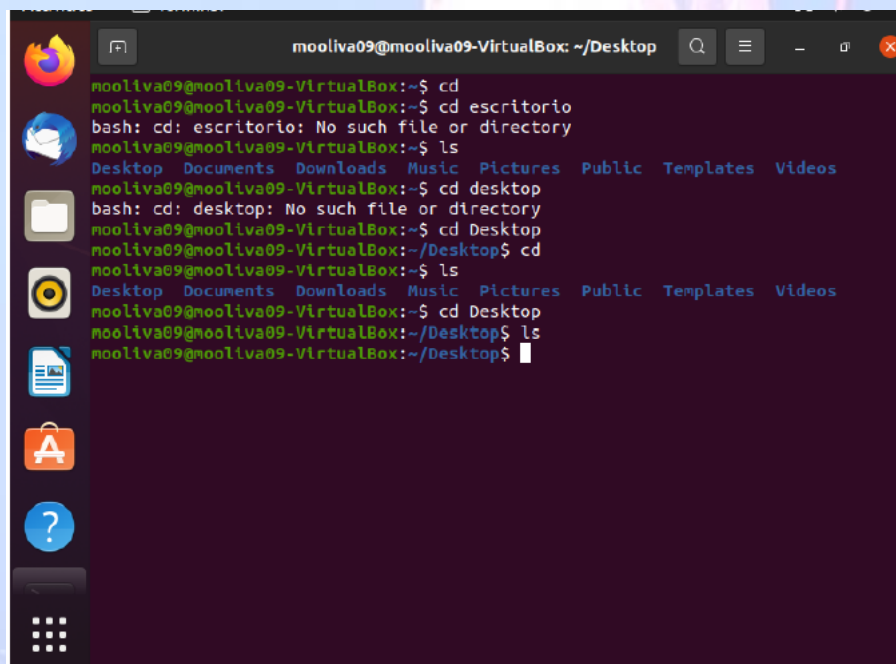
ls	cd nombre_directorio	cp [ruta1]archivo1 [ruta2][archivo 2]
ls -l	cat nombre_archivo	mv [ruta1]archivo1 [ruta2][archivo2]
ls -la	ls -la more	mkdir nombre_directorio
pwd	rm nombre_archivo	rmdir nombre_directorio
clear	Ps	whoami

Observe los resultados de la ejecución de cada comando.



```
mooliva09@mooliva09-VirtualBox: ~  
mooliva09@mooliva09-VirtualBox:~$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
mooliva09@mooliva09-VirtualBox:~$ ls -l  
11809 Desktop      11810 Downloads  11815 Pictures  11811 Templates  
11813 Documents  11814 Music     11812 Public   11816 Videos  
mooliva09@mooliva09-VirtualBox:~$ ls -la  
459037 .                419291 .cache      11796 .gnupg      11812 Public  
393218 ..             419298 .config     419292 .local      408756 .ssh  
419299 .bash_history  11809 Desktop  11814 Music   11811 Templates  
406516 .bash_logout  11813 Documents 11815 Pictures 11816 Videos  
455674 .bashrc       11810 Downloads 406518 .profile  
mooliva09@mooliva09-VirtualBox:~$ pwd  
/home/mooliva09  
mooliva09@mooliva09-VirtualBox:~$
```

Imagen 5 VirtualBox comandos



```
mooliva09@mooliva09-VirtualBox: ~/Desktop  
mooliva09@mooliva09-VirtualBox:~$ cd  
mooliva09@mooliva09-VirtualBox:~$ cd escritorio  
bash: cd: escritorio: No such file or directory  
mooliva09@mooliva09-VirtualBox:~$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
mooliva09@mooliva09-VirtualBox:~$ cd desktop  
bash: cd: desktop: No such file or directory  
mooliva09@mooliva09-VirtualBox:~$ cd Desktop  
mooliva09@mooliva09-VirtualBox:~/Desktop$ cd  
mooliva09@mooliva09-VirtualBox:~/Desktop$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
mooliva09@mooliva09-VirtualBox:~/Desktop$ cd Desktop  
mooliva09@mooliva09-VirtualBox:~/Desktop$ ls  
mooliva09@mooliva09-VirtualBox:~/Desktop$
```

Imagen 6 VirtualBox comandos 1

6. 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.

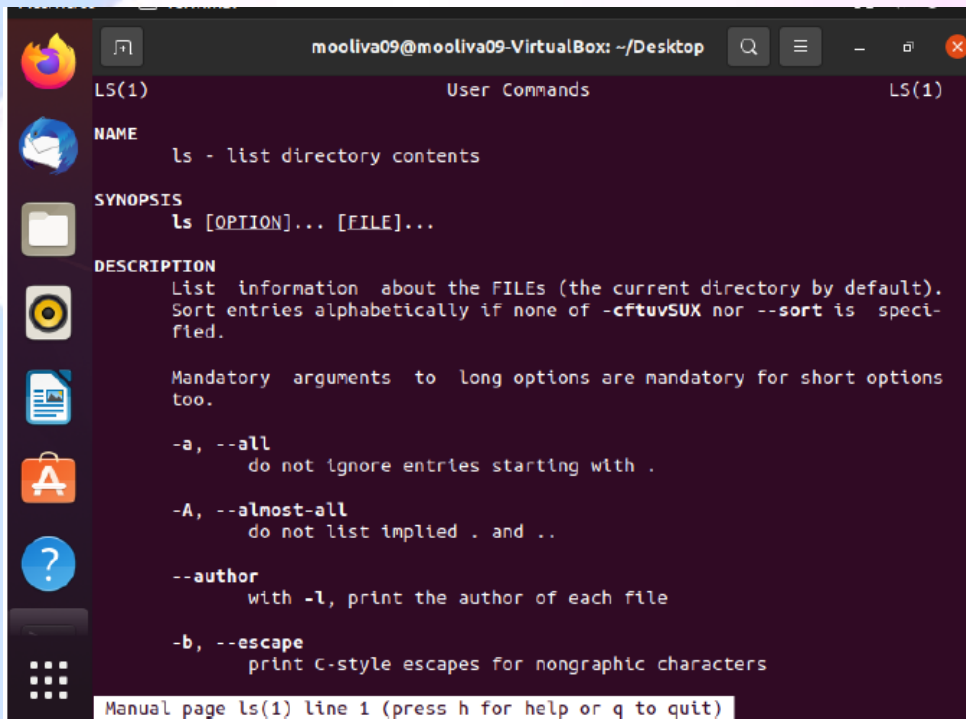


Imagen 7 VirtualBox comando man

7. 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.

```
mooliva09@mooliva09-VirtualBox:~/Desktop$ vi Holamundo.c  
mooliva09@mooliva09-VirtualBox:~/Desktop$ gcc Holamundo.c -o Holamundo  
  
mooliva09@mooliva09-VirtualBox:~/Desktop$ ./Holamundo
```

Imagen 8 VirtualBox "Hola mundo"

```
#include<stdio.h>

int main () {
printf("Hola Mundo");

}
```

8. 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.

```

Ingresa el numero de * que deaceas (debe ser impar): 7
  *           *           *
 **         ***         **
 ***       *****       ***
 ****     *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
 ****     *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
 ***       *****       ***
 **         ***         **
 *           *           *
```

Imagen 10 VirtualBzox programa asteriscos

```

#include<stdio.h>

void Print(const int *i);
void Print2(const int *aux);
void Print3(const int *i);
void Print4(const int *aux);

int main(){
    int a;
    do{
printf("Ingresa el numero de * que deseas (debe ser impar): ");
        scanf("%d", &a);
    }while((a % 2) == 0 );
    int i, j , n=0,cont=1;
    while(cont != a){
        cont+=2;
        n+=1;
    }
    n+=1;
    for(i=1;i<=n;i++){
        int aux,aux2;
        for(aux=3; aux>0; aux--){
            if(i==n){
                printf("*");
            }else{
                printf(" ");
            }
        }
        aux2 = n-i;
        Print(&i);
    }
}

```

```

Print2(&aux2);
Print2(&aux2);
for(j=1; j<=2*i-1; j++){
    printf("*");
}
Print2(&aux2);
Print2(&aux2);
Print(&i);
if(i==n){
for(aux=3; aux>0; aux--){
    printf("*");
}
}
printf("\n");
}
for(i=n; i>=1; i--){
    if(i!=n){
int aux, aux2;
for(aux=3; aux>0; aux--){
    printf(" ");
}
aux2 = n-i;
Print3(&i);
Print4(&aux2);
Print4(&aux2);
for(j=2*i-1; j>=1; j--){
    printf("*");
}
}

```



```
Print4(&aux2);
Print4(&aux2);
    Print3(&i);
    printf("\n");
    }
    }
    }
void Print (const int *i){
    int j;
    for(j=1; j <= *i; j++){
        printf("*");
    }
}
void Print2(const int *aux){
    int j;
    for(j=1; j <= *aux; j++){
        printf(" ");
    }
}
void Print3 (const int *i){
    int j;
    for(j= *i; j >= 1; j--){
        printf("*");
    }
}
void Print4(const int *aux){
    int j;
    for(j=*aux; j >= 1; j--){
```

```
printf(" ");  
}  
}
```

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

```
#include <stdio.h>  
  
int main()  
{  
    int i, n, t1 = 0, t2 = 1, nextTerm;  
    printf("Ingresa un numero: ");  
    scanf("%d", &n);  
    printf("Serie Fibonacci: ");  
    for (i = 1; i <= n; ++i)  
    {  
        printf("%d, ", t1);  
        nextTerm = t1 + t2;  
        t1 = t2;  
        t2 = nextTerm;  
    }  
    return 0;  
}
```

```
Ingresa un numero: 8  
Serie Fibonacci: 0, 1, 1, 2, 3, 5, 8, 13,
```

Imagen 11 VirtualBox código fibonacci

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

Primero tenemos el funcionamiento de la pila que es el siguiente:

```
typedef struct Nodo{
    Elem dato;
    struct Nodo * sig;
} * Pila;

Pila empty(){
    return NULL;
}

Pila push (Elem e, Pila l){
    Pila t=(Pila)malloc(sizeof(struct Nodo));
    t->dato=e;
    t->sig=l;
    return t;
}

int isempty(Pila l){
    return l==NULL;
}

Elem top(Pila l){
    return l->dato;
}

Pila pop(Pila l){
    return l->sig; }
```

Y ya que tenemos la pila el código del programa en sí es el siguiente:

```
#include <stdio.h>
#include <stdlib.h>
```

```
#include "Elem.h"
#include "Pila.h"

int parenthesisAbre(char c){
    int i=0;
    if(c=='(' || c=='[' || c=='{')
        i=1;
    return i;
}

int parenthesisCierra(char c){
    int i=0;
    if(c==')' || c==']' || c=='}')
        i=1;
    return i;
}

int concuerdan(char a, char c){
    int i=0;
    if((a=='(' && c==')')
        || (a=='[' && c==']')
        || (a=='{' && c=='}'))
        i=1;
    return i;
}

int main(int argc, char *argv[]){
    int i=0;
    Pila balanceo = empty();
    char *pal;
    pal=*(argv+1);
    while(*pal){
```

```

        if(parentesisAbre(*pal))
balanceo = push(*pal,balanceo);
        else if(parentesisCierra(*pal)){
            if(isempty(balanceo)){
                int i=0;
                break;
            }
            else{
                i = concuerdan(top(balanceo),*pal);
                balanceo=pop(balanceo);
            }
        }
        *pal++;
    }
    if(i==1 && isempty(balanceo))
printf("La cadena esta balanceada de parentesis");
        else
printf("La cadena no esta balanceada de parentesis");
        return 0;
    }

```

La cadena esta balanceada de parentesis

La cadena no esta balanceada de parentesis

11. 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

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>

struct nodo {
    char simbolo;
    struct nodo *sig;
};

struct nodo *raiz=NULL;

void insertar(char x)
{
    struct nodo *nuevo;
    nuevo = malloc(sizeof(struct nodo));
    nuevo->simbolo = x;
    if (raiz == NULL)
    {
        raiz = nuevo;
        nuevo->sig = NULL;
    }
    else
    {
        nuevo->sig = raiz;
        raiz = nuevo;
    }
}
```

```

    }
    char extraer()
    {
        if (raiz != NULL)
        {
            char informacion= raiz->simbolo;
            struct nodo *bor = raiz;
            raiz = raiz->sig;
            free(bor);
            return informacion;
        }
        else
        {
            return -1;
        }
    }

    void liberar()
    {
        struct nodo *reco = raiz;
        struct nodo *bor;
        while (reco != NULL)
        {
            bor = reco;
            reco = reco->sig;
            free(bor);
        }
    }

    int vacia()

```

```
        {
            if (raiz == NULL)
                return 1;
            else
                return 0;
        }
void cargarFormula(char *formula)
{
    printf("Ingrese la formula:");
    gets(formula);
}
int verificarBalanceada(char *formula)
{
    int f;
    for (f=0;f<strlen(formula);f++)
    {
        if (formula[f]=='(' || formula[f]=='[' || formula[f]=='{')
        {
            insertar(formula[f]);
        }
        else
        {
            if (formula[f]==')')
            {
                if (extraer()!='(')
                {
                    return 0;
                }
            }
        }
    }
}
```

```

    }
    else
    {
if (formula[f]=='')
        {
            if (extraer()!='(')
                {
                    return 0;
                }
            }
        else
        {
if (formula[f]=='{')
            {
                if (extraer()!='{')
                    {
                        return 0;
                    }
                }
            }
        }
        }
        }
        }
        }
    if (vacía())
    {
        return 1;
    }
    else

```

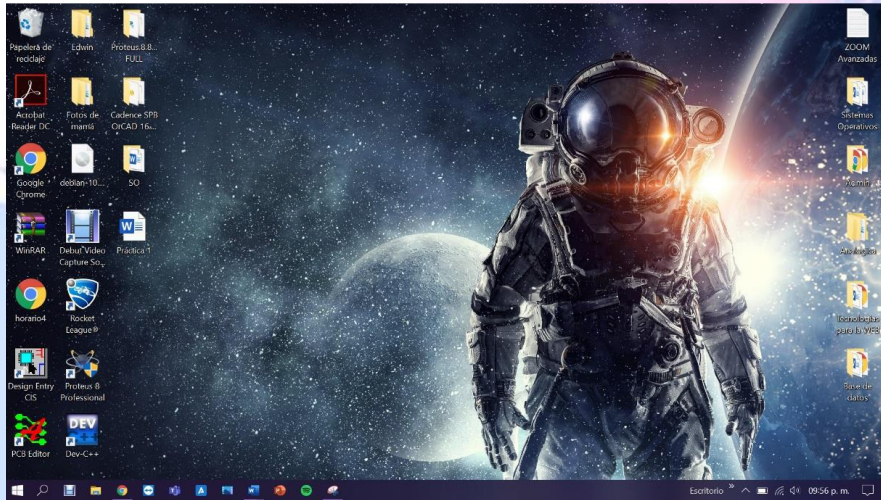
```
    {  
        return 0;  
    }  
}  
  
int main()  
{  
    char formula[100];  
    cargarFormula(formula);  
    if (verificarBalanceada(formula))  
    {  
        printf("La formula esta correctamente balanceada");  
    }  
    else  
    {  
        printf("La formula no esta correctamente balanceada");  
    }  
    liberar();  
    getch();  
    return 0;  
}
```

```
Ingrese la formula:(((5-2)+4)*10)  
La formula esta correctamente balanceada
```


12. Guarde sus programas (fuente y ejecutable) en memoria usb.

Sección Windows

1. Inicie sesión en Windows.



2. Abra una consola.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 10.0.19042.804]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Edwin>
```

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).

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_archivo

- Dir.

```
C:\Users\Edwinn\Desktop\P1>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: D440-2F9D

Directorio de C:\Users\Edwinn\Desktop\P1

28/02/2021  10:25 p. m.    <DIR>        .
28/02/2021  10:25 p. m.    <DIR>        ..
28/02/2021  10:25 p. m.                0 gg.txt
28/02/2021  10:25 p. m.                0 prueba borrar.txt
                2 archivos                0 bytes
                2 dirs  86,693,040,128 bytes libres
```

- Cd

```
C:\Users\Edwinn\Desktop\P1>cd ..

C:\Users\Edwinn\Desktop>cd P1

C:\Users\Edwinn\Desktop\P1>
```

- Copy

```
C:\Users\Edwinn\Desktop\P1>Copy gg.txt gg1.txt
1 archivo(s) copiado(s).

C:\Users\Edwinn\Desktop\P1>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: D440-2F9D

Directorio de C:\Users\Edwinn\Desktop\P1

28/02/2021  10:28 p. m.    <DIR>        .
28/02/2021  10:28 p. m.    <DIR>        ..
28/02/2021  10:25 p. m.                0 gg.txt
28/02/2021  10:25 p. m.                0 gg1.txt
28/02/2021  10:25 p. m.                0 prueba borrar.txt
                3 archivos                0 bytes
                2 dirs  86,690,308,096 bytes libres
```

- Ipconfig

```

C:\Users\Edwinn\Desktop\P1>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Estado de los medios. . . . . : medios desconectados
    Sufijo DNS específico para la conexión. . . :

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 . . . . . : 2806:104e:18:ba22::1
    Dirección IPv6 . . . . . : 2806:104e:18:ba22:490e:c5a6:c577:1982
    Dirección IPv6 . . . . . : fd34:6ac2:c93d:4800:490e:c5a6:c577:1982
    Dirección IPv6 temporal. . . . . : 2806:104e:18:ba22:6d94:d97f:6931:3f6d
    Dirección IPv6 temporal. . . . . : fd34:6ac2:c93d:4800:2084:30ae:94df:67ed
    Vínculo: dirección IPv6 local. . . : fe80::490e:c5a6:c577:1982%7
    Dirección IPv4. . . . . : 192.168.1.107
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . : fe80::1%7
                                              192.168.1.254

```

- Type

```

C:\Users\Edwinn\Desktop\P1>type gg.txt
Esta prueba es para el comando TYPE.

```

- Ren

```

C:\Users\Edwinn\Desktop\P1>ren gg1.txt ren.txt

C:\Users\Edwinn\Desktop\P1>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: D440-2F9D

Directorio de C:\Users\Edwinn\Desktop\P1

28/02/2021  11:00 p. m.    <DIR>          .
28/02/2021  11:00 p. m.    <DIR>          ..
28/02/2021  10:34 p. m.                37 gg.txt
28/02/2021  10:25 p. m.                0 prueba borrar.txt
28/02/2021  10:25 p. m.                0 ren.txt
                3 archivos                37 bytes
                2 dirs 86,689,378,304 bytes libres

```

- Cls

```

28/02/2021  11:00 p. m.    <DIR>        .
28/02/2021  11:00 p. m.    <DIR>        ..
28/02/2021  10:34 p. m.                37 gg.txt
28/02/2021  10:25 p. m.                0 prueba borrar.txt
28/02/2021  10:25 p. m.                0 ren.txt
                3 archivos                37 bytes
                2 dirs  86,689,378,304 bytes libres

C:\Users\Edwinn\Desktop\P1>cls

```

```

C:\Users\Edwinn\Desktop\P1>

```

- Mkdir

```

C:\Users\Edwinn\Desktop\P1>mkdir Nueva

C:\Users\Edwinn\Desktop\P1>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: D440-2F9D

Directorio de C:\Users\Edwinn\Desktop\P1

28/02/2021  11:51 p. m.    <DIR>        .
28/02/2021  11:51 p. m.    <DIR>        ..
28/02/2021  11:43 p. m.    <DIR>        Desktop
28/02/2021  10:34 p. m.                37 gg.txt
28/02/2021  11:51 p. m.    <DIR>        Nueva
28/02/2021  10:25 p. m.                0 prueba borrar.txt
28/02/2021  10:25 p. m.                0 ren.txt
                3 archivos                37 bytes
                4 dirs  86,657,441,792 bytes libres

```

- Chdir

```

C:\Users\Edwinn\Desktop\P1>chdir
C:\Users\Edwinn\Desktop\P1

```

- Ver

```

C:\Users\Edwinn\Desktop\P1>ver

Microsoft Windows [Versión 10.0.19042.804]

```

- Rmdir

```
C:\Users\Edwinn\Desktop\P1>ver
```

```
Microsoft Windows [Versión 10.0.19042.804]
```

- Echo

```
C:\Users\Edwinn\Desktop\P1>echo hola mundo  
hola mundo
```

- Tree

```
C:\Users\Edwinn\Desktop\P1>Tree  
Listado de rutas de carpetas  
El número de serie del volumen es D440-2F9D  
C:.  
└──Desktop
```

- Del

```
C:\Users\Edwinn\Desktop\P1>del "prueba borrar.txt"
```

```
C:\Users\Edwinn\Desktop\P1>dir  
El volumen de la unidad C no tiene etiqueta.  
El número de serie del volumen es: D440-2F9D
```

```
Directorio de C:\Users\Edwinn\Desktop\P1
```

01/03/2021	12:22 a. m.	<DIR>	.
01/03/2021	12:22 a. m.	<DIR>	..
28/02/2021	11:43 p. m.	<DIR>	Desktop
28/02/2021	10:34 p. m.		37 gg.txt
28/02/2021	10:25 p. m.		0 ren.txt
		2 archivos	37 bytes
		3 dirs	86,627,151,872 bytes libres

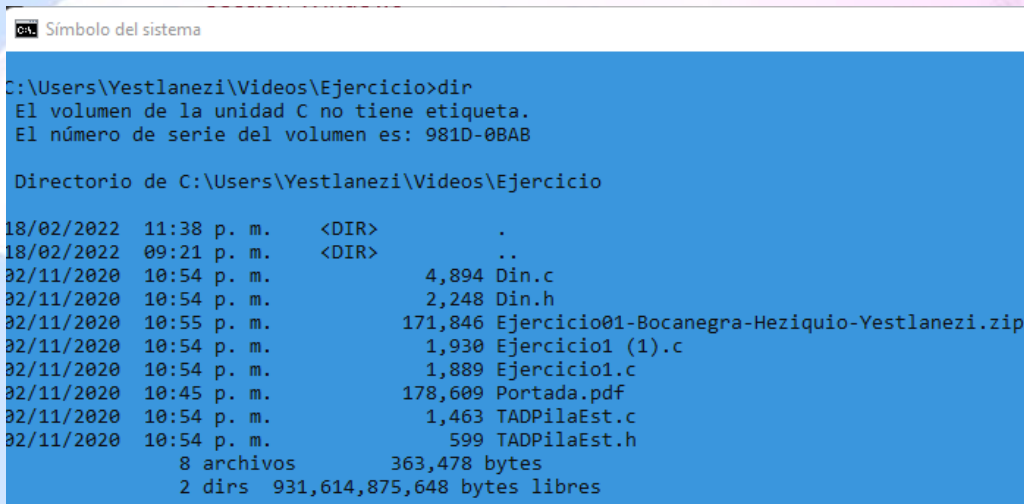
- Find

```
C:\Users\Edwinn\Desktop\P1>FIND "Esta prueba" gg.txt
```

```
----- GG.TXT  
Esta prueba es para el comando TYPE.
```

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

Dir: El comando se utiliza para saber la ubicación en la que te encuentras



```

C:\Users\Vestlanezi\Videos\Ejercicio>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 981D-0BAB

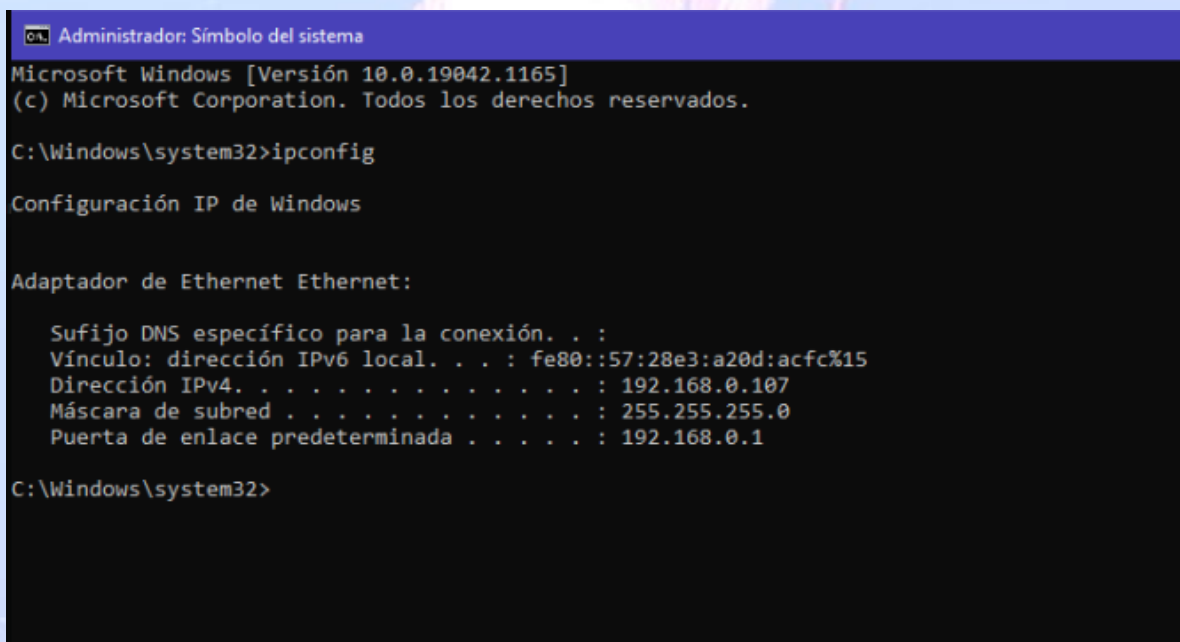
Directorio de C:\Users\Vestlanezi\Videos\Ejercicio

18/02/2022  11:38 p. m.   <DIR>          .
18/02/2022  09:21 p. m.   <DIR>          ..
02/11/2020  10:54 p. m.       4,894 Din.c
02/11/2020  10:54 p. m.       2,248 Din.h
02/11/2020  10:55 p. m.    171,846 Ejercicio01-Bocanegra-Heziquio-Vestlanezi.zip
02/11/2020  10:54 p. m.       1,930 Ejercicio01 (1).c
02/11/2020  10:54 p. m.       1,889 Ejercicio01.c
02/11/2020  10:45 p. m.    178,609 Portada.pdf
02/11/2020  10:54 p. m.       1,463 TADPilaEst.c
02/11/2020  10:54 p. m.         599 TADPilaEst.h
                8 archivos      363,478 bytes
                2 dirs  931,614,875,648 bytes libres

```

Imagen 12 CMD comando Dir

Ipconfig: el comando se utiliza para obtener la configuración de la ip



```

C:\Windows\system32>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Sufijo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . : fe80::57:28e3:a20d:acfc%15
    Dirección IPv4. . . . . : 192.168.0.107
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . : 192.168.0.1

C:\Windows\system32>

```

Imagen 13 CMD comando ipconfig

Cls: El comando se utiliza para borrar la pantalla

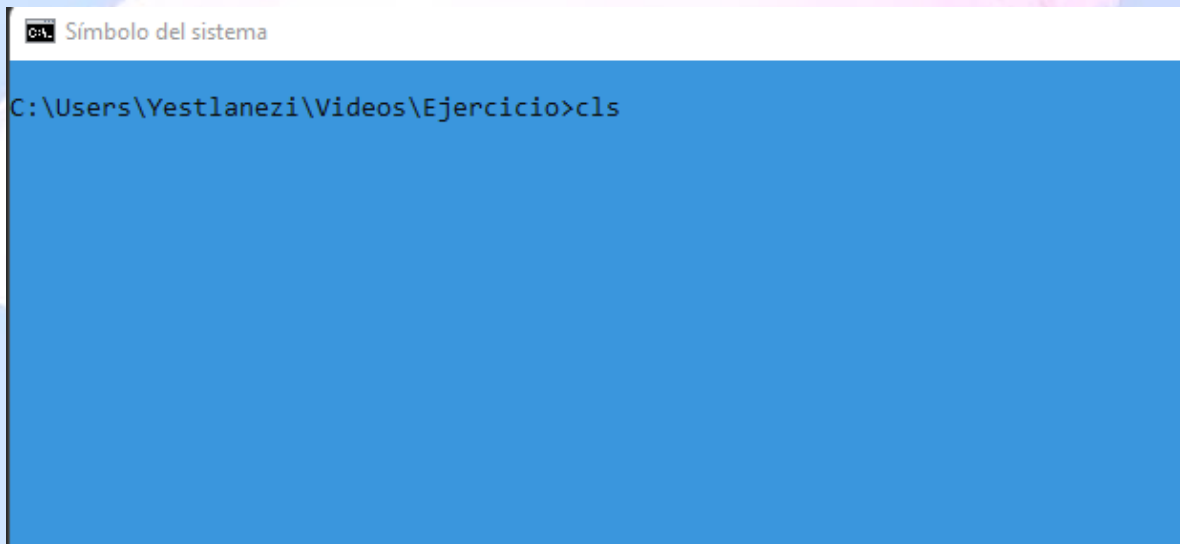
A screenshot of a Windows Command Prompt window. The title bar at the top is white and contains the text 'Símbolo del sistema' next to a small icon. The main area of the window has a blue background. The text 'C:\Users\Yestlanezi\Videos\Ejercicio>cls' is displayed in white, indicating the current directory and the command entered.

Imagen 14 CMD comando Cls

Ver: el comando es utilizado para saber la versión de Windows

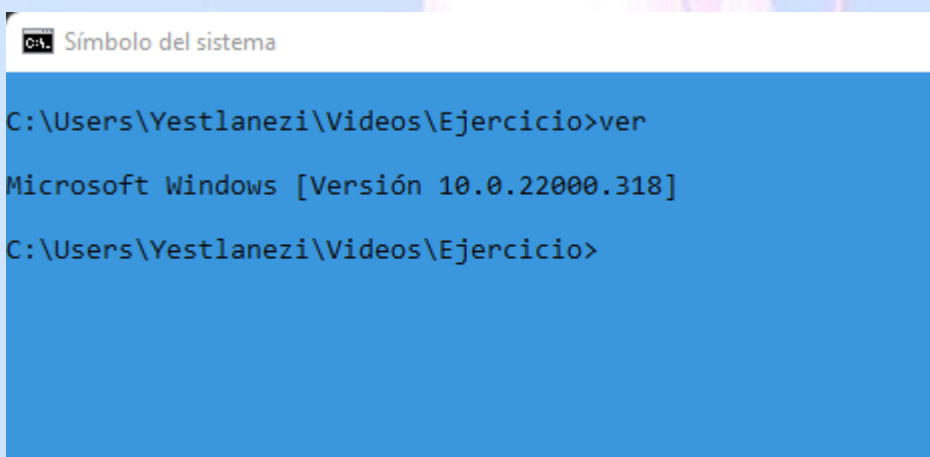
A screenshot of a Windows Command Prompt window. The title bar at the top is white and contains the text 'Símbolo del sistema' next to a small icon. The main area of the window has a blue background. The text 'C:\Users\Yestlanezi\Videos\Ejercicio>ver' is displayed in white. Below this, the output 'Microsoft Windows [Versión 10.0.22000.318]' is shown in white. The prompt 'C:\Users\Yestlanezi\Videos\Ejercicio>' is shown again at the bottom.

Imagen 15 CMD comando ver

Tree: el comando sirve para obtener datos sobre la raíz del disco duro y las subcarpetas

```

C:\Users\Yestlanezi\Videos\Ejercicio>ver

Microsoft Windows [Versión 10.0.22000.318]

C:\Users\Yestlanezi\Videos\Ejercicio>tree
Listado de rutas de carpetas
El número de serie del volumen es 981D-0BAB
C:.
No existe ninguna subcarpeta

C:\Users\Yestlanezi\Videos\Ejercicio>
```

Imagen 16 CMD comando tree

Cd : < RutaDirectorio >* para ir al directorio o carpeta concreta que le diga

```

C:\Users\Yestlanezi\Videos\S0>C:

C:\Users\Yestlanezi\Videos\S0>D:

D:\>C:

C:\Users\Yestlanezi\Videos\S0>..
".." no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\Yestlanezi\Videos\S0>..
".." no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\Yestlanezi\Videos\S0>cd ..

C:\Users\Yestlanezi\Videos>cd ..

C:\Users\Yestlanezi>cd videos

C:\Users\Yestlanezi\Videos>
```

Imagen 17 CMD comando Cd

Type: sirve para ver el contenido de los archivos de texto planos.

```
C:\Users\Yestlanezi\Videos>Type rombo.c
#include <stdio.h>

int main()
{
    int i,j,n, c, k, lineas = 1;

    printf("Ingrese el número de filas:\n");
    scanf("%d", &n);

    lineas = n - 1;
    for (k = 1; k <= n; k++){
        for (c = 1; c <= lineas; c++){
            printf(" ");
        }

        lineas--;

        for (c = 1; c <= ((2*k)-1)/2; c++){
            printf("*");
        }

        printf("\n");
    }

    lineas = 1;

    for (k = 1; k <= n - 1; k++){
        for (c = 1; c <= lineas; c++){
```

Imagen 18 CMD comando Type

Mkdir: Esta herramienta es usada para crear un nuevo subdirectorio o carpeta del sistema de archivos.

```
C:\Users\Yestlanezi\Videos>Mkdir videos

C:\Users\Yestlanezi\Videos>Mkdir Videos
Ya existe el subdirectorio o el archivo Videos.

C:\Users\Yestlanezi\Videos>
```

Imagen 19 CMD comando Mkdir

Rmdir: Elimina un directorio. Quita un directorio. desea quitar un árbol.

```
C:\Users\Yestlanezi\Videos>Mkdir videos
C:\Users\Yestlanezi\Videos>Mkdir Videos
Ya existe el subdirectorio o el archivo Videos.

C:\Users\Yestlanezi\Videos>rmdir
La sintaxis del comando no es correcta.

C:\Users\Yestlanezi\Videos>Rmdir
La sintaxis del comando no es correcta.

C:\Users\Yestlanezi\Videos>rmdir videos
C:\Users\Yestlanezi\Videos>
```

Imagen 20 CMD comando rmdir

Copy: sirve para copiar uno o varios archivos en otra ubicación (directorio). COPY es un comando interno de cmd.exe.

```
C:\Users\Yestlanezi\Videos>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 981D-0BAB

Directorio de C:\Users\Yestlanezi\Videos

22/02/2022 09:08 p. m. <DIR> .
18/02/2022 09:17 p. m. <DIR> ..
21/02/2022 12:52 a. m. 41,450 a.exe
21/02/2022 12:53 a. m. 520 asteriscos.c
04/02/2022 10:28 a. m. <DIR> Captures
18/02/2022 11:38 p. m. <DIR> Ejercicio
22/02/2022 07:24 p. m. 41,476 f.exe
05/01/2021 11:13 a. m. 1,139 fibonacci.c
22/02/2022 08:45 p. m. 1,439 p.c
22/02/2022 08:09 p. m. 1,819 parentesis.c
20/02/2022 03:20 p. m. 1,094 rombo.c
20/02/2022 03:20 p. m. 41,448 rombo.exe
22/02/2022 08:53 p. m. <DIR> SO
22/02/2022 08:33 p. m. 4,578 TADPilaEst.c
22/02/2022 08:34 p. m. 1,842 TADPilaEst.h
22/02/2022 08:47 p. m. 51,713 xd.exe
11 archivos 188,518 bytes
5 dirs 929,059,028,992 bytes libres

C:\Users\Yestlanezi\Videos>copy rombo.c
No se puede copiar el archivo sobre sí mismo.
0 archivo(s) copiado(s).
```

Imagen 21 CMD comando Copy

Ren: es un comando del sistema operativo DOS, de la Interfaz de línea de comandos

```
C:\Users\Yestlanezi\Videos>nuevo rombo.c
"nuevo" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

C:\Users\Yestlanezi\Videos>ren rombo.c rom.c
```

Imagen 22 CMD comando ren

Chdir: (CHange DIRectory) tiene la misma funcionalidad que el comando CD, es decir, permite cambiar de directorio y, también, se puede utilizar para mostrar el nombre del directorio actual. Ambos son comandos internos de cmd.exe.

```
C:\Users\Yestlanezi\Videos>Chdir videos
El sistema no puede encontrar la ruta especificada.

C:\Users\Yestlanezi\Videos>chdir videos
El sistema no puede encontrar la ruta especificada.
```

Imagen 23 CMD comando Chdir

Find: Básicamente el comando find, busca una cadena de texto en un archivo o varios archivos y despliega las líneas de texto que contiene la cadena seleccionada.

```
C:\Users\Yestlanezi\Videos>Find rombo.c
FIND: formato de parámetros incorrecto
```

Imagen 24 CMD comando find

5. Abra la carpeta de su memoria usb donde tiene los programas que desarrolló en la sección de Linux y ejecútelos. Dé sus observaciones sobre la ejecución de los mismos.

HolaMundo.cpp

```
1 #include<stdio.h>
2 int main (){
3     printf("Hola Mundo");
4 }
5
```

C:\Windows\system32\cmd.exe

C:\Users\Edwinn\Desktop\P1>FIND "Esta prueba" gg.txt

----- GG.TXT
Esta prueba es para el comando TYPE.

C:\Users\Edwinn\Desktop\P1>

HolaMundo.cpp Fig. Rombo.cpp

```
1 #include<stdio.h>
2
3 void Print(const int *i);
4 void Print2(const int *aux);
5 void Print3(const int *i);
6 void Print4(const int *aux);
7
8 int main(){
9     int a;
10    do{
11        printf("Ingresa el numero de * que deseas (debe ser impar): ");
12        scanf("%d", &a);
13    }while((a % 2) == 0);
14
15    int i, j , n=0,cont=1;
16
17    while(cont != a){
18        cont+=2;
19        n+=1;
20    }
21
22    n+=1;
23    for(i=1;i<=n;i++){
24        Print(i);
25        Print2(aux);
26    }
27}
```

C:\Users\Edwinn\Desktop\P1\Fig. Rombo.exe

Ingresa el numero de * que deseas (debe ser impar): 9

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

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

Process exited after 5.508 seconds with return value 0
Presione una tecla para continuar . . .

HolaMundo.cpp Fig. Rombo.cpp Serie Fibonacci.cpp

```
1 #include <stdio.h>
2 int main()
3 {
4     int i, n, t1 = 0, t2 = 1, nextTerm;
5
6     printf("Ingresa un numero: ");
7     scanf("%d", &n);
8
9     printf("Serie Fibonacci: ");
10
11    for (i = 1; i <= n; ++i)
12    {
13        printf("%d, ", t1);
14        nextTerm = t1 + t2;
15        t1 = t2;
16        t2 = nextTerm;
17    }
18    return 0;
19 }
20
```

C:\Users\Edwinn\Desktop\P1\Serie Fibonacci.exe

Ingresa un numero: 7
Serie Fibonacci: 0, 1, 1, 2, 3, 5, 8,

Process exited after 2.794 seconds with return value 0
Presione una tecla para continuar . . .


```
Probl Aritmetica.c
1  #include<stdio.h>
2  #include<conio.h>
3  #include<stdlib.h>
4  #include<string.h>
5
6
7  struct nodo {
8      char simbolo;
9      struct nodo *sig;
10 };
11
12 struct nodo *raiz=NULL;
13
14 void insertar(char x)
15 {
16     struct nodo *nuevo;
17     nuevo = malloc(sizeof(struct nodo));
18     nuevo->simbolo = x;
19     if (raiz == NULL)
20     {
21         raiz = nuevo;
22         nuevo->sig = NULL;
23     }
24     else
25     {
26         nuevo->sig = raiz;
27         raiz = nuevo;
28     }
29 }
30
31
32 char extraer()
33 {
34     if (raiz != NULL)
```

C:\Users\Edwinn\Desktop\P1\Prob. Aritmetica\Probl Aritmetica.exe

Ingrese la formula: (((5-2)+4)*10).

La formula esta correctamente balanceada

6. Busque el directorio donde está instalado Dev C.

```
C:\Program Files (x86)>cd Dev-Cpp
```

```
C:\Program Files (x86)\Dev-Cpp>
```

7. Desde la consola ubíquese en el directorio de Dev C, y cambie al directorio "bin"

```
C:\Program Files (x86)\Dev-Cpp>cd C:\Program Files (x86)\Dev-Cpp\MinGW64\bin
```

```
C:\Program Files (x86)\Dev-Cpp\MinGW64\bin>dir
```

8. Desde ese directorio compile en la consola como se indica en el punto 10 de la sección Linux cada uno de los programas creados en esa sección y que guardó en la memoria usb, ejecútelos y observe su funcionamiento.


```
C:\Users\Edwinn\Desktop\P1\Fig. Rombo>gcc Rombo.c -o rombo
```

```
C:\Users\Edwinn\Desktop\P1\Fig. Rombo>rombo
```

```
Ingresa el numero de * que deseas (debe ser impar): 10
```

```
Ingresa el numero de * que deseas (debe ser impar): 9
```

```
  *      *      *
 **     ***     **
 ***    *****    ***
 ****   ********   ****
*****
*****
****   *****   ****
***    *****   ***
**     ***       **
*      *         *
```

```
C:\Users\Edwinn\Desktop\P1\Prob. Aritmetica>gcc "Probl Aritmetica.c" -o Probl
```

```
C:\Users\Edwinn\Desktop\P1\Prob. Aritmetica>Probl
```

```
Ingresa la formula:(((5-2)+4)*10)
```

```
La formula esta correctamente balanceada
```

```
C:\Users\Edwinn\Desktop\P1>gcc BalanceoF.c -o Balanceo
```

```
C:\Users\Edwinn\Desktop\P1>Balanceo
```

```
()()(){}[]
```

```
La formula esta correctamente balanceada
```

```
C:\Users\Edwinn\Desktop\P1>Balanceo
```

```
()()({}[]
```

```
La formula no esta correctamente balanceada
```

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

```
C:\Users\Yestlanezi\Videos>gcc a.c -o a
```

```
C:\Users\Yestlanezi\Videos>a.exe
```

```
Ingresa el numero de * que deseas (debe ser impar): 7
```

```
  *      *      *
 **     ***     **
 ***    *****    ***
*****
***   *****   ***
**    ***       **
*     *         *
```

10. Imagen 25 CMD Código asteriscos en Windows

```
C:\Users\Yestlanezi\Videos>fibonacci.exe  
|-----|  
|Serie de Fibonacci|  
|-----|  
Introduzca un numero:10  
La serie Fibonacci en la posicion [0] es: 0  
La serie Fibonacci en la posicion [1] es: 1  
La serie Fibonacci en la posicion [2] es: 1  
La serie Fibonacci en la posicion [3] es: 2  
La serie Fibonacci en la posicion [4] es: 3  
La serie Fibonacci en la posicion [5] es: 5  
La serie Fibonacci en la posicion [6] es: 8  
La serie Fibonacci en la posicion [7] es: 13  
La serie Fibonacci en la posicion [8] es: 21  
La serie Fibonacci en la posicion [9] es: 34  
La serie Fibonacci en la posicion [10] es: 55
```

Imagen 26 CMD serie de Fibonacci desde windows

Los programas se compilan relativamente igual, en ambos se accede a la carpeta donde se encuentra el código, se ejecutan con un gcc NombreDelPrograma.c -o NombreADarle, siendo más notable la diferencia en la ejecución de los programas ya que en windows se escribe: NombrePrograma.exe y en linux solo se usa: ./NombrePrograma, también podemos señalar que si bien algunos comandos en linux y en windows se escriben diferente y algunos se escriben igual podemos encontrar teniendo funcionalidades parecidas en ambos sistemas

Conclusión

Con base en lo anterior, podemos observar que el objetivo de la practica era familiarizarnos con los diferentes sistemas operativos, en este caso Windows y Linux, es muy común que Linux sea utilizado desde una máquina virtual, aunque Linux es un sistema muy amigable con el usuario, estas mas acostumbrados a utilizar Windows.

Windows es un sistema operativo que con el paso del tiempo se actualiza, actualmente podemos interactuar con el entorno de Windows 11 donde se puede observar que el entorno es amigable con el usuario que lleva tiempo conociendo este sistema operativo. Los comandos y terminal de Windows CMD es más fácil de utilizar, aunque para ejecutar y compilar programas es necesario instalarlo directamente, mientras que la terminal de Linux te permite hacer la instalación de estos desde su terminal.

En los dos sistemas operativos tenemos comandos con nombres distintos, pero las funciones son las mismas, al menos de forma estética, al igual que la seguridad, el multiusuario, distribuciones de Linux, entre otras cosas, pero en sí, en este apartado podemos usar ambos sistemas operativos sin problemas.