## INTRODUCCIÓN A UNIX

MC. Dulce I. Valdivia

dulce.valdivia@cinvestav.mx

30.septiembre.2022



Red Mexicana de Bioinformática



Dr. Yalbi Balderas + Servidor + Servidor

## ¡GRACIAS!

O

Dr. Giovani Marco

Material de ejercicios

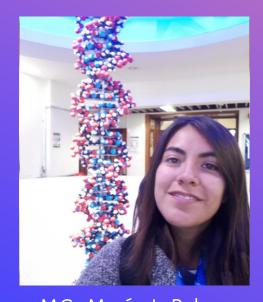


MC. Shirley Alquicira

Dr. Irma Martínez

Proyectos y Docencia

#### AYUDANTES 🐥



MC. María J. Palma

Doctorado en Ciencias Biomédicas

UNAM-CCG



Ing. Irecha Cano

Maestría en Biotecnología de Plantas

CINVESTAV-Irapuato

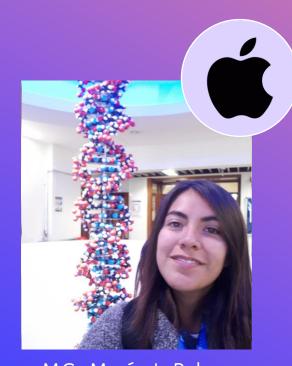


Ing. Adrián Díaz Maestría en Biología Integrativa, CINVESTAV-Irapuato



Biól. Emiliano Juárez Facultad de Ciencias de la UNAN

### **AYUDANTES**



MC. María J. Palma Doctorado en Ciencias Biomédicas UNAM-CCG



Ing. Irecha Cano



Ing. Adrián Díaz



Biól. Emiliano Juárez

# SISTEMAS OPERATIVOS UNIX



#### Sistemas Unix





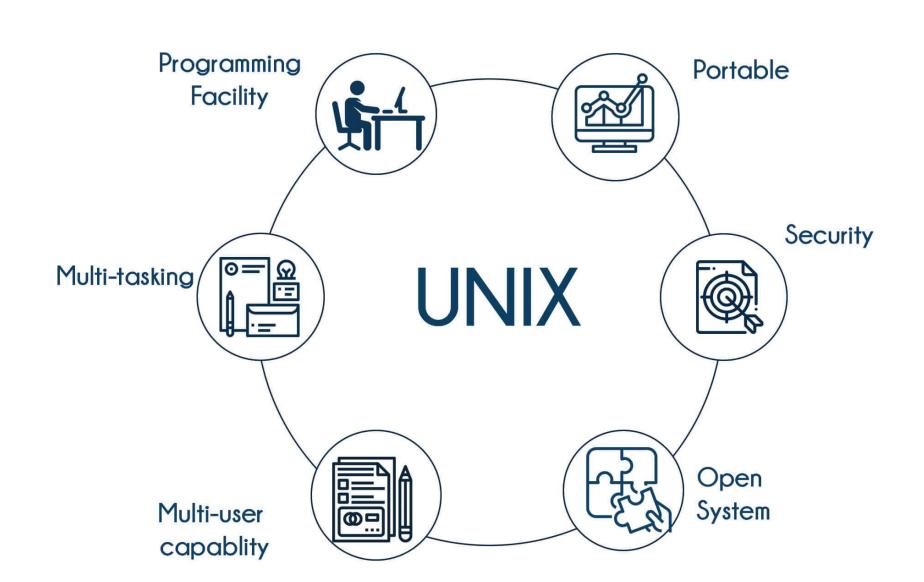




#### Sistemas Unix

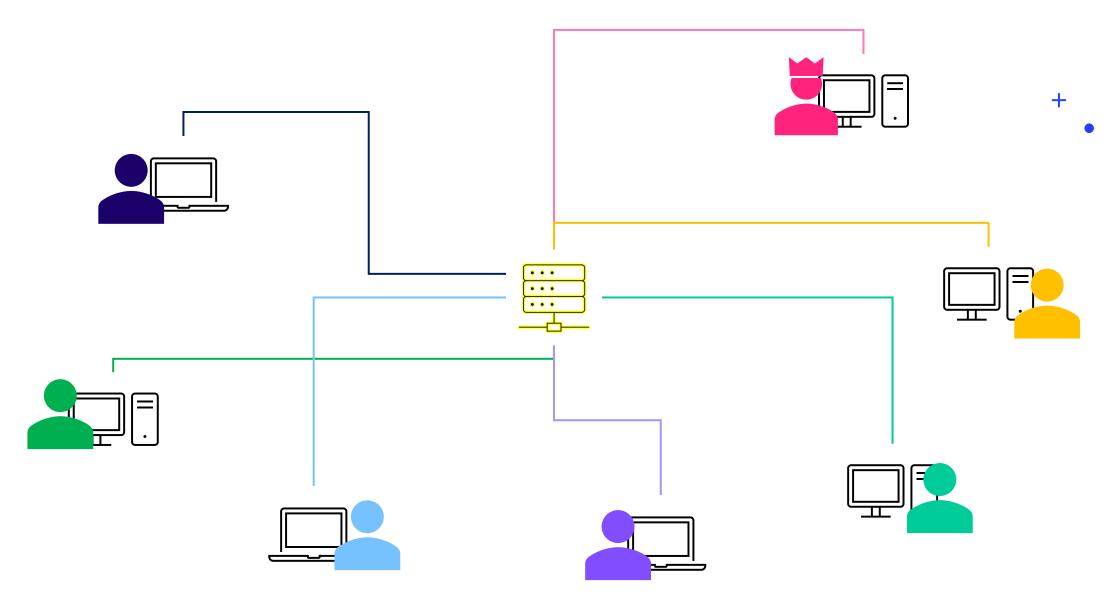


+

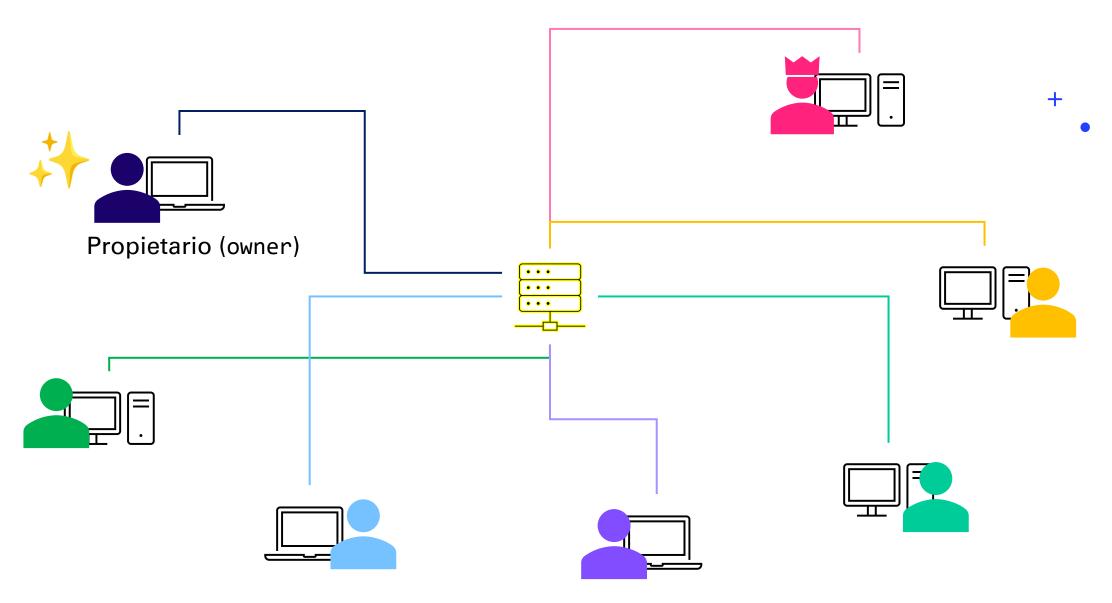




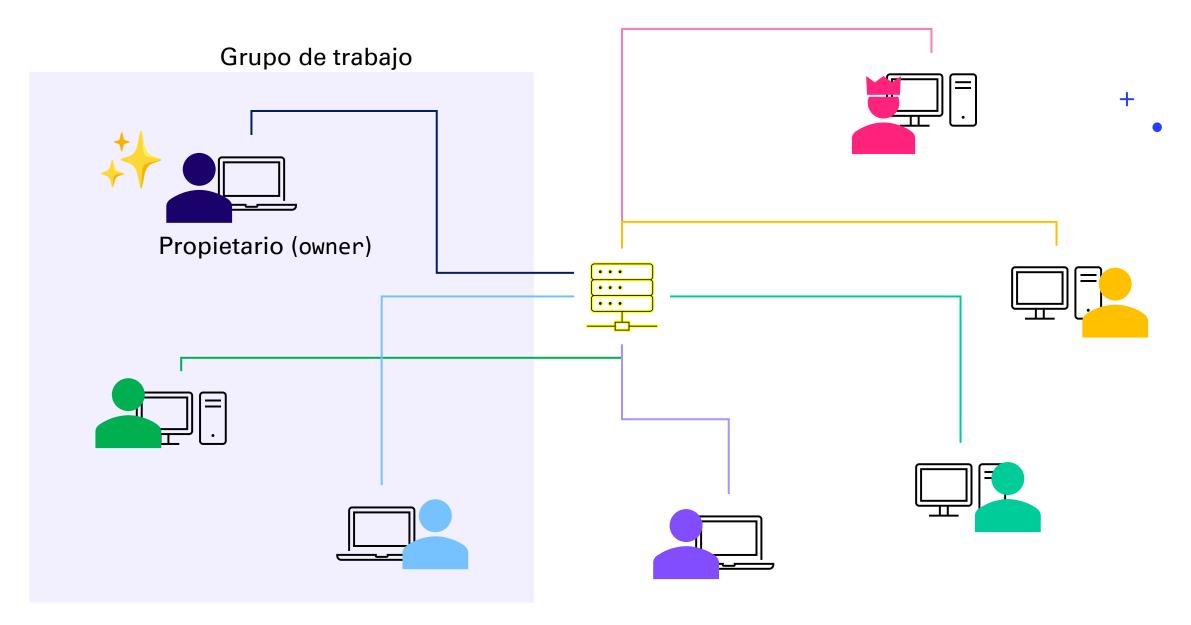




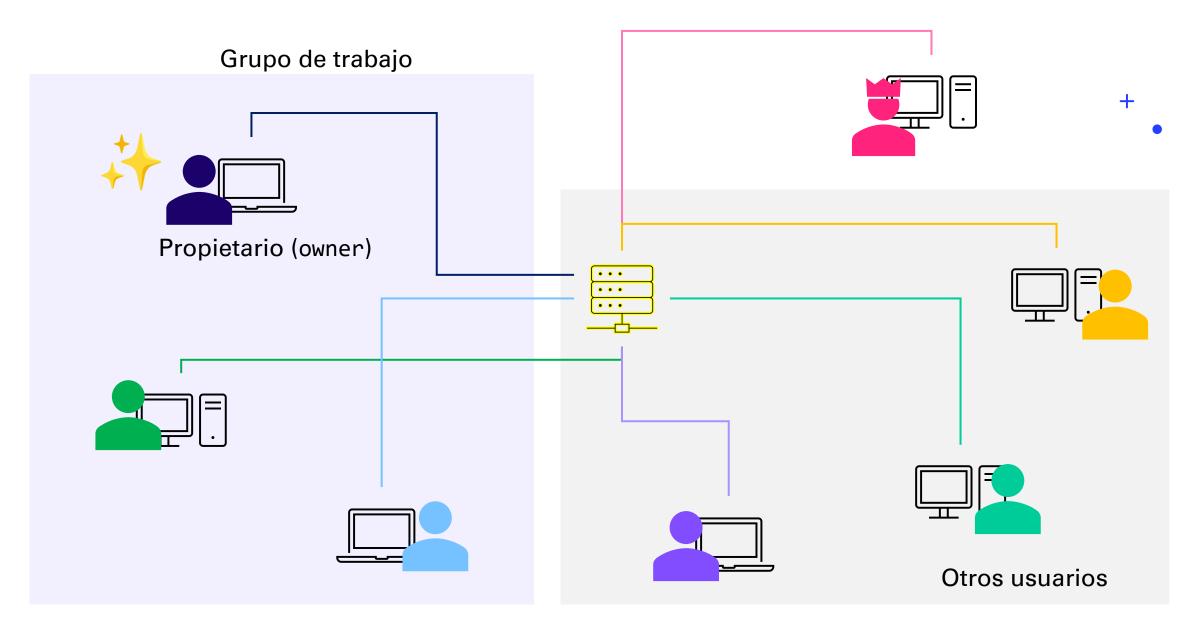




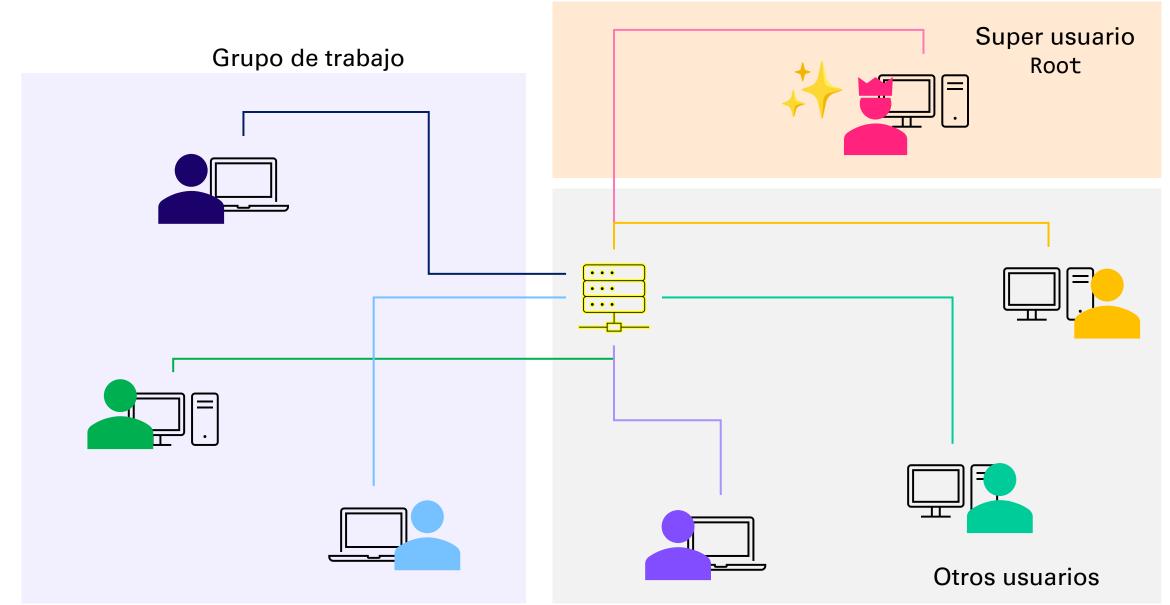




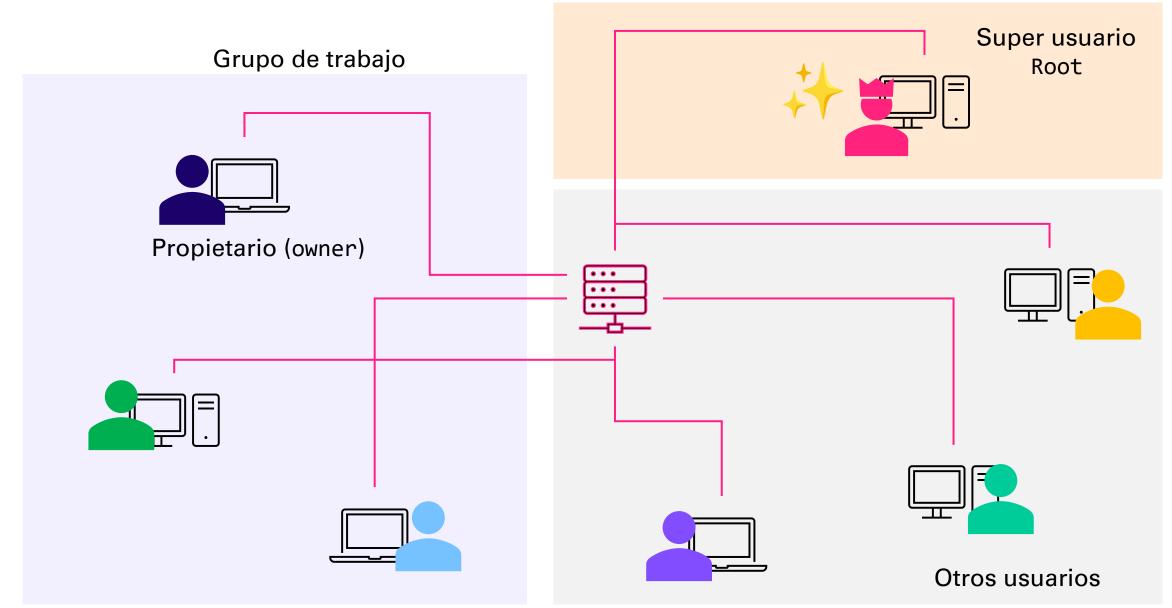




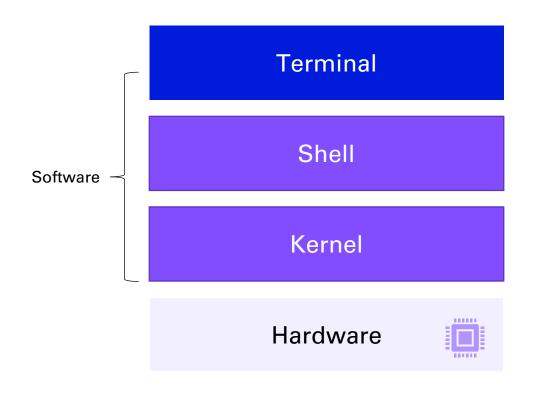




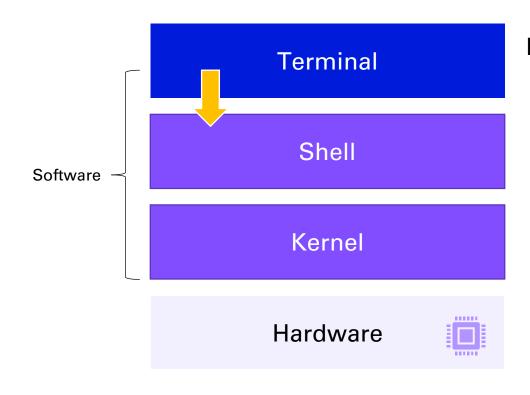






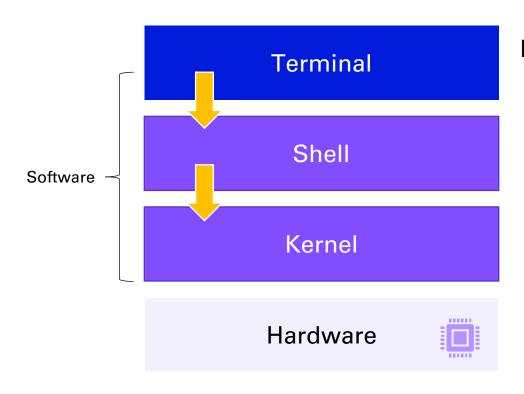






Entrada y salida de comandos de los usuarios

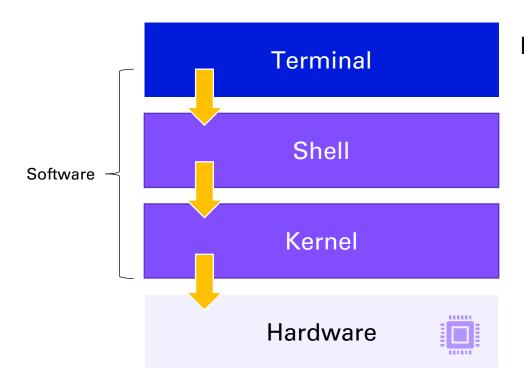




Entrada y salida de comandos de los usuarios

Intérprete de la línea de comandos y el kernel



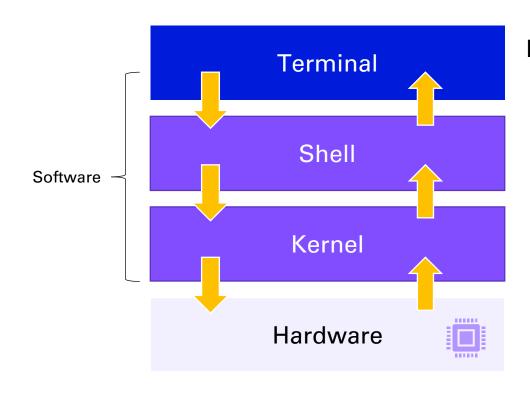


Entrada y salida de comandos de los usuarios

Intérprete de la línea de comandos y el kernel

Ejecuta los procesos administrando los recursos (CPU, tiempo, entrada/salida).



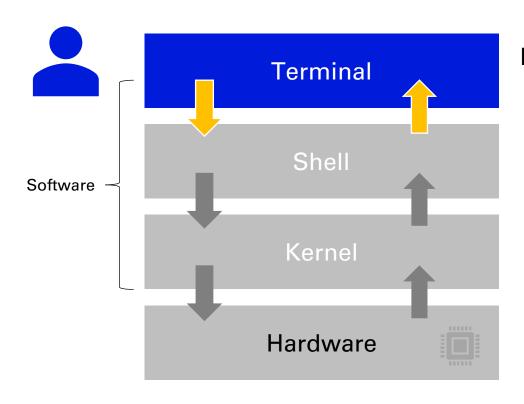


Entrada y salida de comandos de los usuarios

Intérprete de la línea de comandos y el kernel

Ejecuta los procesos administrando los recursos (CPU, tiempo, entrada/salida).





Entrada y salida de comandos de los usuarios

Intérprete de la línea de comandos y el kernel

Ejecuta los procesos administrando los recursos (CPU, tiempo, entrada/salida).





```
irecha@irecha-cano:~ Q = - □ 😣
```

quiénSoy@dóndeEstoyTrabajando \$





quiénSoy@dóndeEstoyTrabajando \$

usuario





quiénSoy@dóndeEstoyTrabajando \$

en (at)





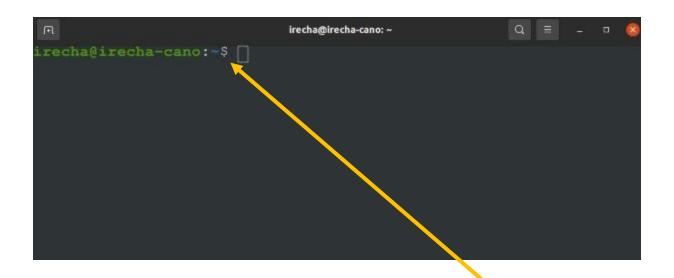
```
irecha@irecha-cano: ~
```

quiénSoy@dóndeEstoyTrabajando)\$

host / computadora





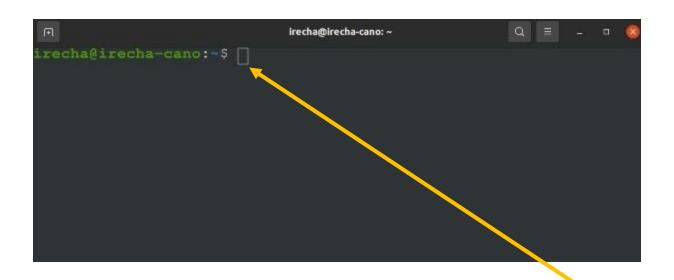


quiénSoy@dóndeEstoyTrabajando \$

**Prompt** 







quiénSoy@dóndeEstoyTrabajando \$

Línea de comandos



#### ¡Manos a la obra! 🐥

Vamos a conectarnos a un servidor remoto para trabajar todos juntos en el mismo sistema Unix:

ssh -v -p 2585 unix\_user01@hpc-matematicas-z.fciencias.unam.mx



#### ¡Manos a la obra! 🔆

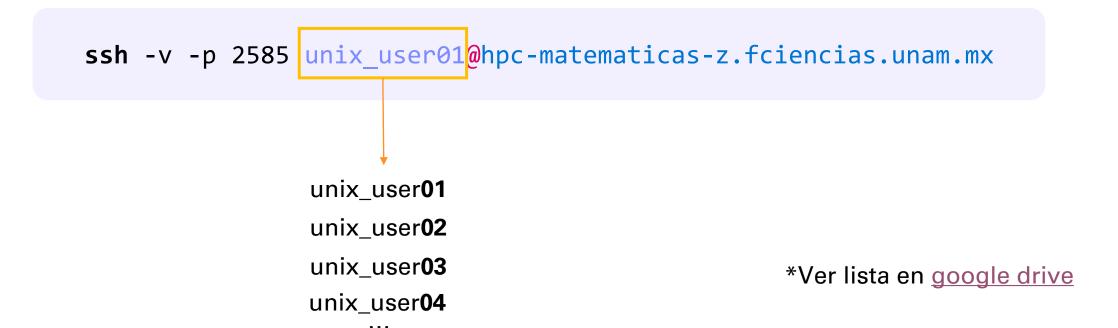
Vamos a conectarnos a un servidor remoto para trabajar todos juntos en el mismo sistema Unix:

ssh -v -p 2585 unix\_user01@hpc-matematicas-z.fciencias.unam.mx



#### ¡Manos a la obra! 🐥

Vamos a conectarnos a un servidor remoto para trabajar todos juntos en el mismo sistema Unix:





#### ¡Manos a la obra! 🔆

Vamos a conectarnos a un servidor remoto para trabajar todos juntos en el mismo sistema Unix:

```
user@whereami $ ssh

user@whereami $ whoami

user@whereami $ passwd

user@whereami $ exit
```

## COMANDOS EN UNIX





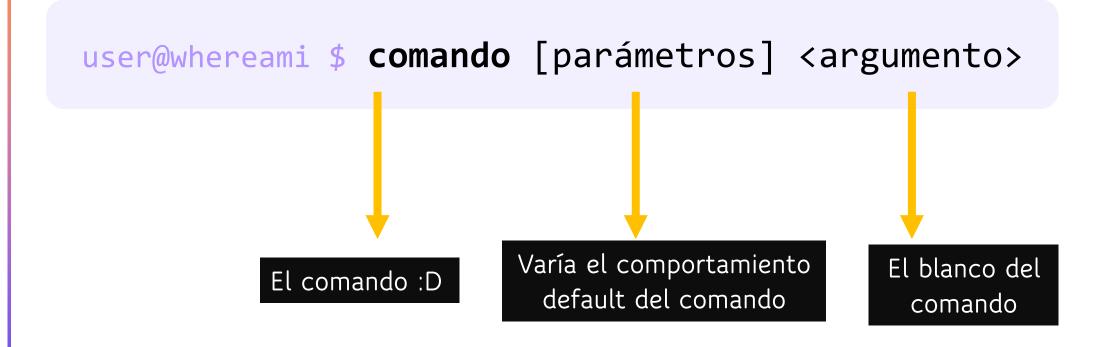


#### Sintaxis básica de los comandos 💝

```
user@whereami $ comando [parámetros] <argumento>
```



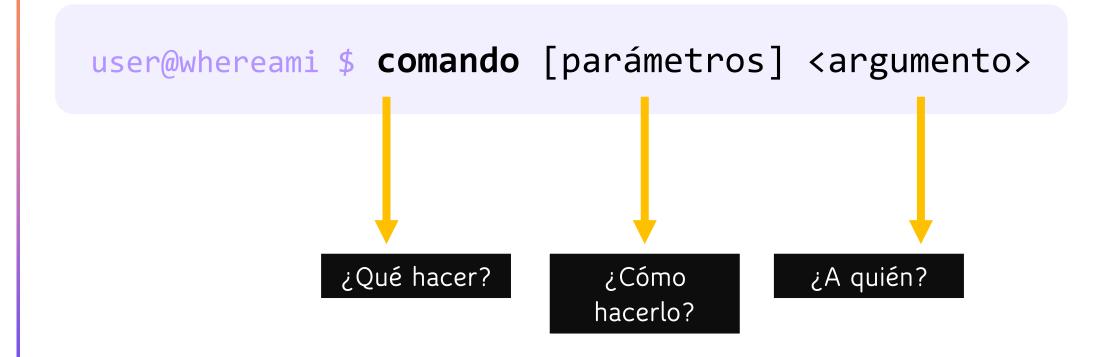






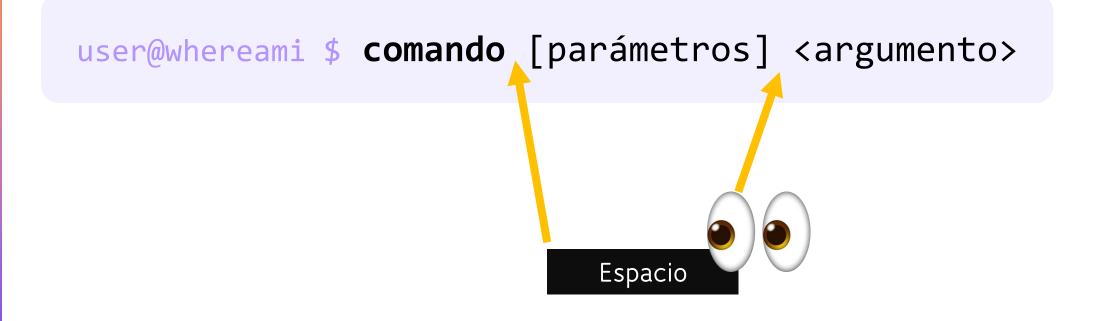
#### Sintaxis básica de los comandos 💝







#### Sintaxis básica de los comandos 💝





#### Programar es como hacer limonada

- 1. Llenar la jarra con agua
- 2. Cortar a la mitad los limones
- 3. Exprimir las mitades de limones en la jarra
- 1. Agregar 1 cucharadas de azúcar a la jarra
- 5. Revolver
- 6. Agregar hielos a la jarra





#### Programar es como hacer limonada

user@whereami \$ comando [parámetros] <argumentos>

- 1. Llenar «la jarra» [con agua]
- 2. Cortar a la mitad los limones
- 3. Exprimir las mitades de limones en la jarra
- 1. Agregar 1 cucharadas de azúcar a la jarra
- 5. Revolver
- 6. Agregar hielos a la jarra





user@whereami \$ comando [parámetros] <argumentos>

- 1. Llenar «la jarra» [con agua]
- 2. Cortar La la mitad 1 < los limones>
- 3. Exprimir las mitades de limones en la jarra
- 1. Agregar 1 cucharadas de azúcar a la jarra
- 5. Revolver
- 6. Agregar hielos a la jarra





user@whereami \$ comando [parámetros] <argumentos>

- 1. Llenar «la jarra» [con agua]
- 2. Cortar La la mitad 1 < los limones>
- 3. Exprimir < las mitades de limones > Len la jarral
- 1. Agregar 1 cucharadas de azúcar a la jarra
- 5. Revolver
- 6. Agregar hielos a la jarra





user@whereami \$ comando [parámetros] <argumentos>

- 1. Llenar «la jarra» [con agua]
- 2. Cortar La la mitad 1 < los limones>
- 3. Exprimir < las mitades de limones > Len la jarral
- 4. Agregar [1 cucharadas de azúcar] <a la jarra>
- 5. Revolver
- 6. Agregar hielos a la jarra





```
user@whereami $ comando [parámetros] <argumentos>
```

- 1. Llenar «la jarra» [con agua]
- 2. Cortar La la mitad 1 < los limones>
- 3. Exprimir < las mitades de limones > Len la jarral
- 4. Agregar [1 cucharadas de azúcar] <a la jarra>
- 5. Revolver [\_\_\_\_] <\_\_\_>
- 6. Agregar hielos a la jarra





```
user@whereami $ comando [parámetros] <argumentos>
```

- 1. Llenar «la jarra» [con agua]
- 2. Cortar La la mitad 1 < los limones>
- 3. Exprimir < las mitades de limones > Len la jarral
- 4. Agregar [1 cucharadas de azúcar] <a la jarra>
- 5. Revolver [\_\_\_\_] <\_\_\_>
- 6. Agregar [hielos] <a la jarra>





# ¡Manos a la obra! 🔆

\$ cowsay [-f] <message>

# . SISTEMA DE ARCHIVOS





En Biología...

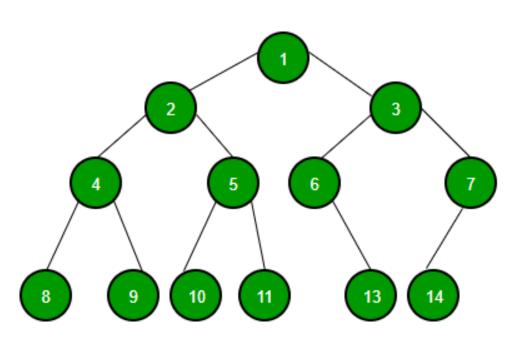


+



En Biología...

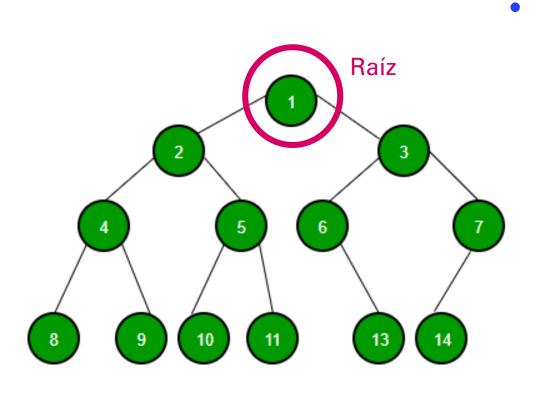






En Biología...

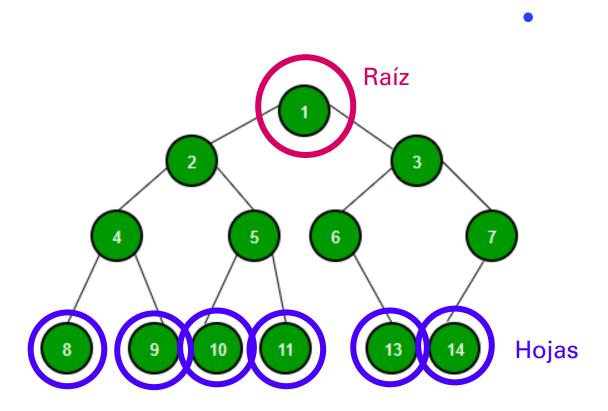






En Biología...

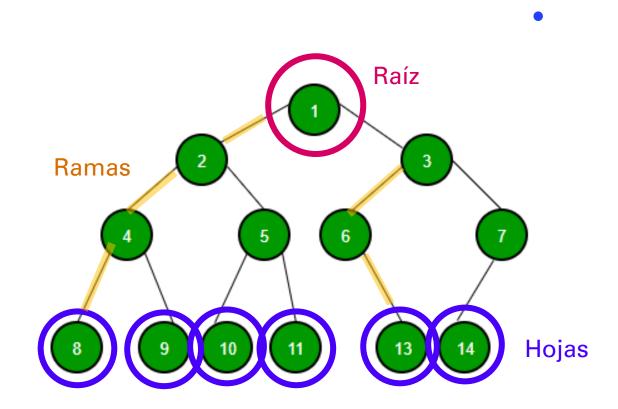






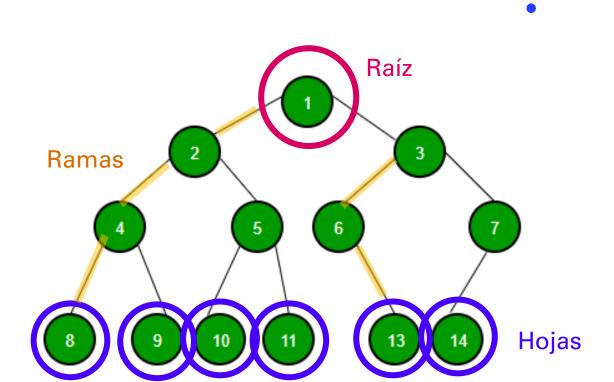
En Biología...





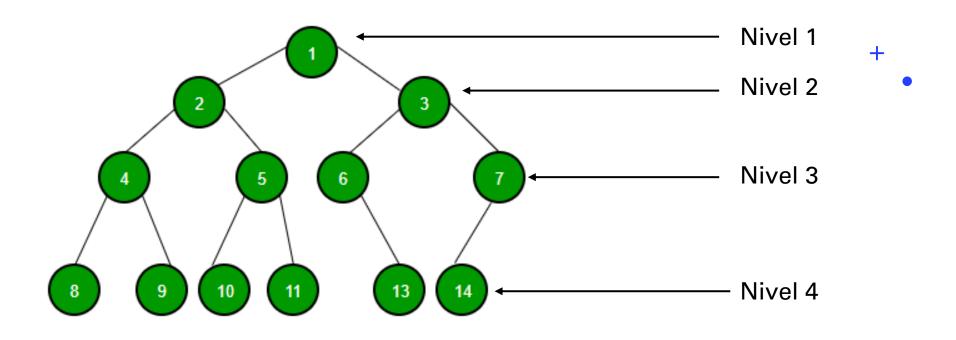




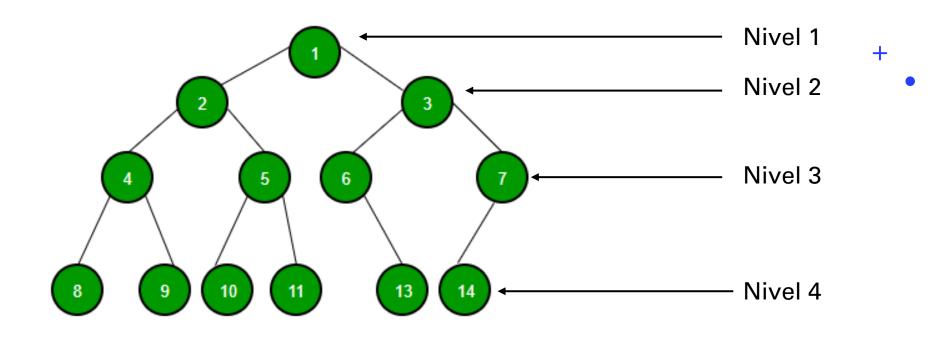


# Los árboles definen una jerarquía

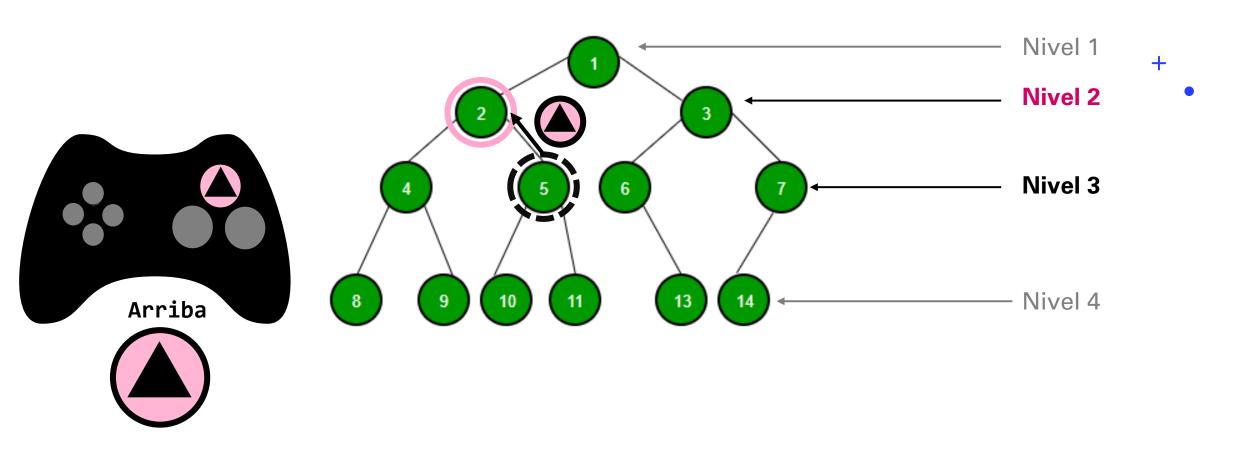




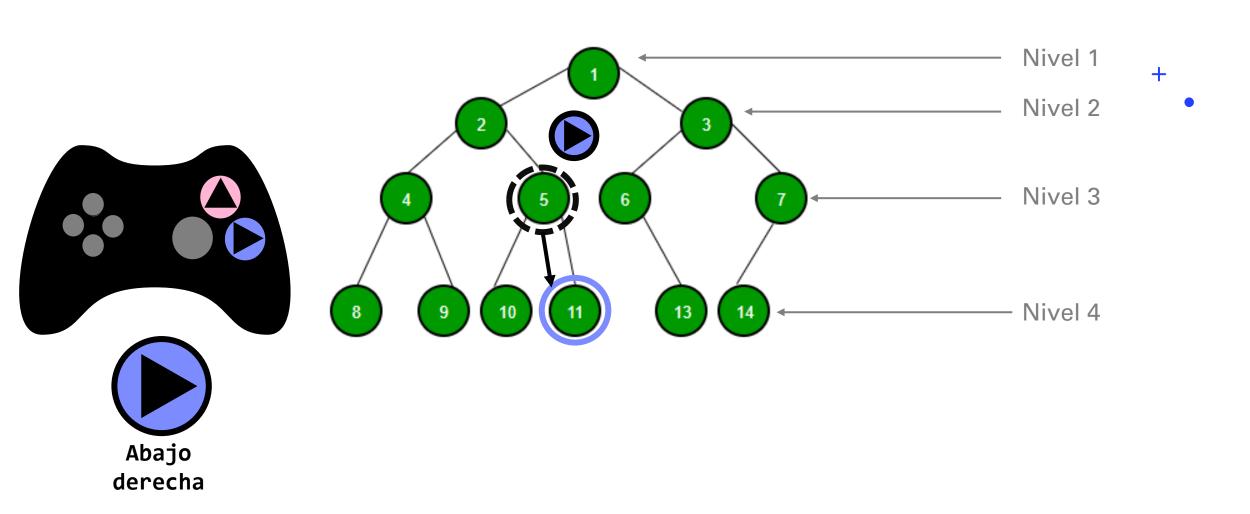




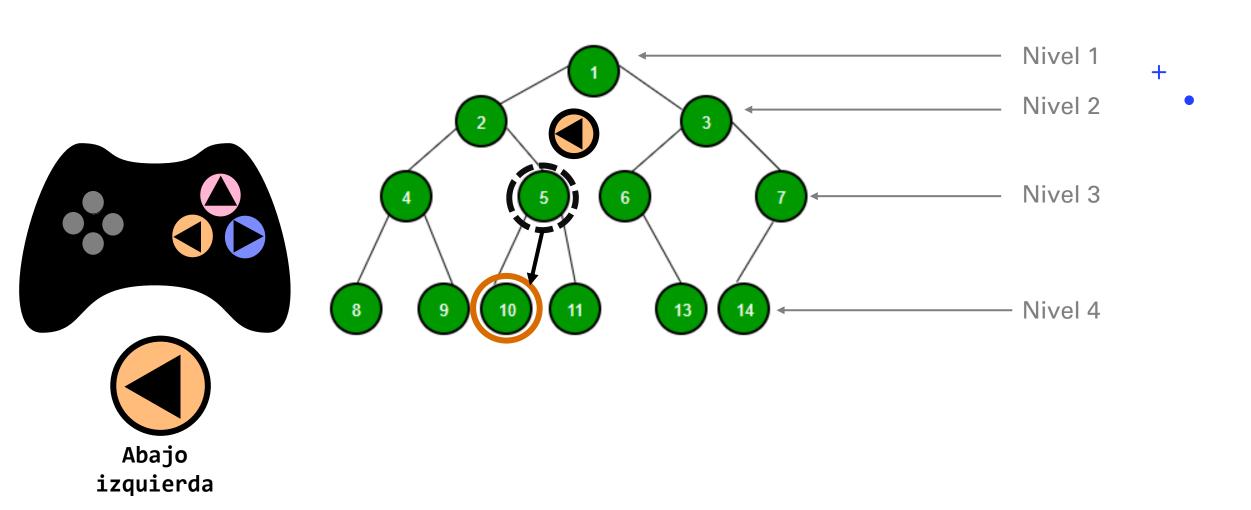






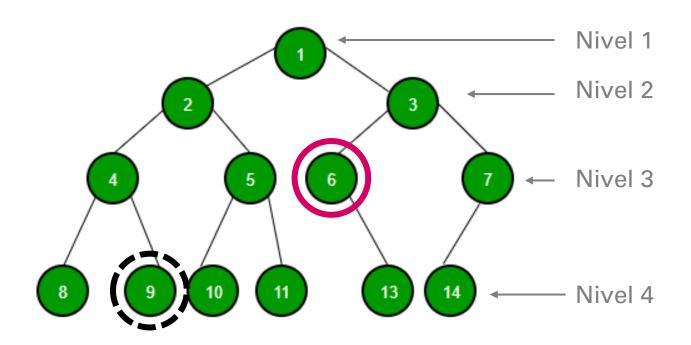


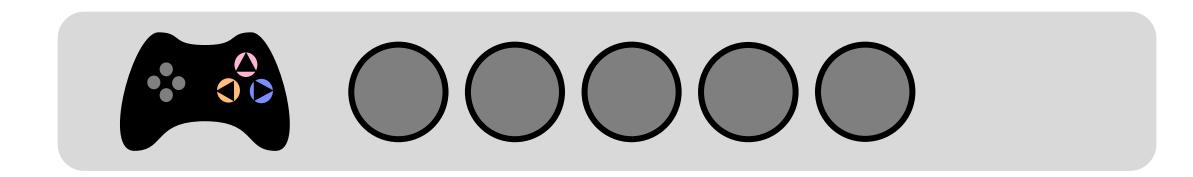




## ¿Cómo llegamos del nodo 9 al 6?

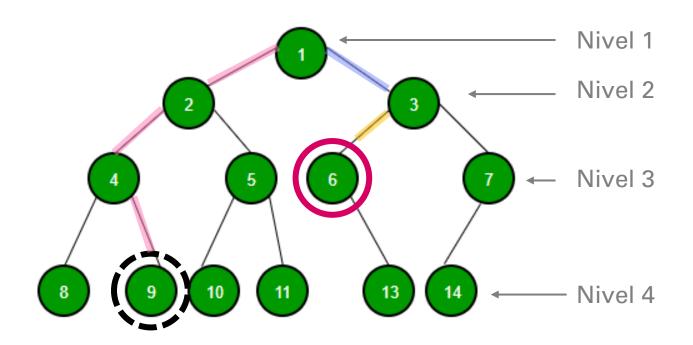


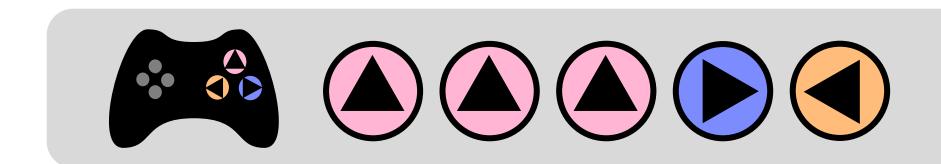




## ¿Cómo llegamos del nodo 9 al 6?

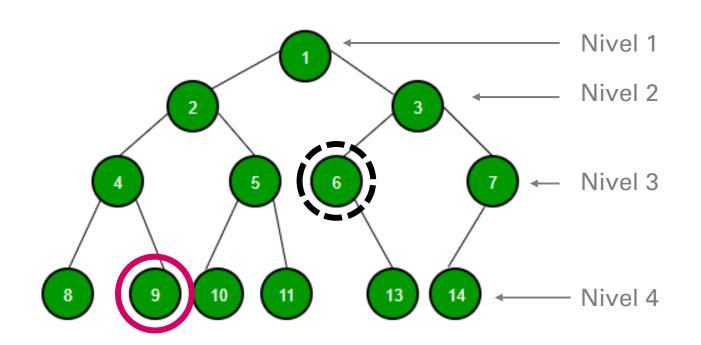


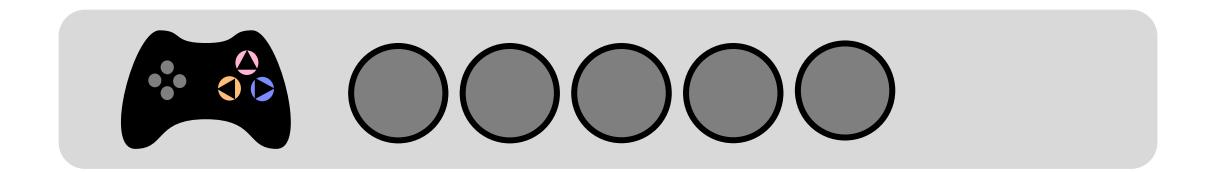




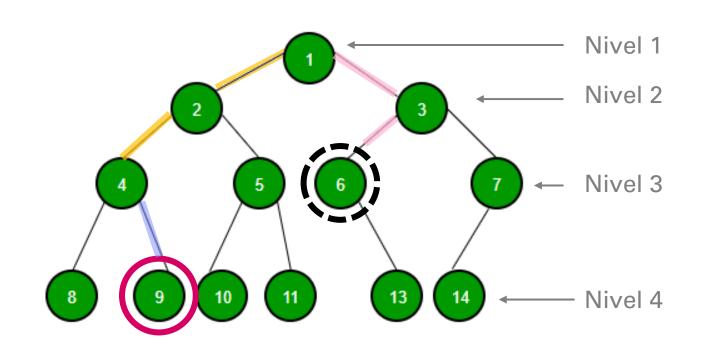
## ¿Cómo **regresamos** del nodo **9** al **6**?

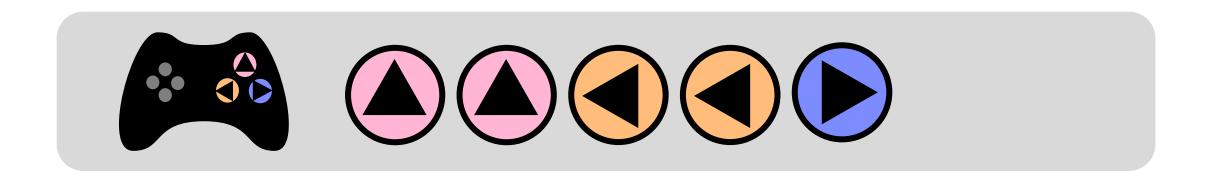






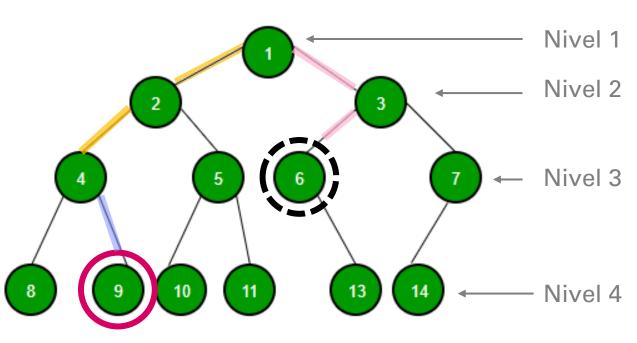
## ¿Cómo **regresamos** del nodo **9** al **6**?

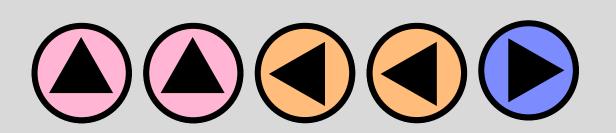




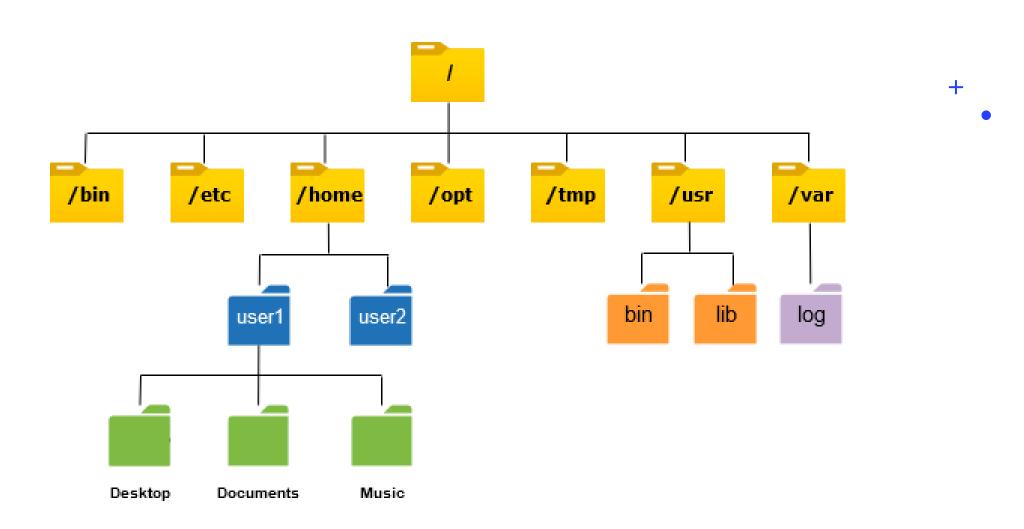


La secuencia de movimientos para llegar de un nodo a otro es llamado un camino (path)

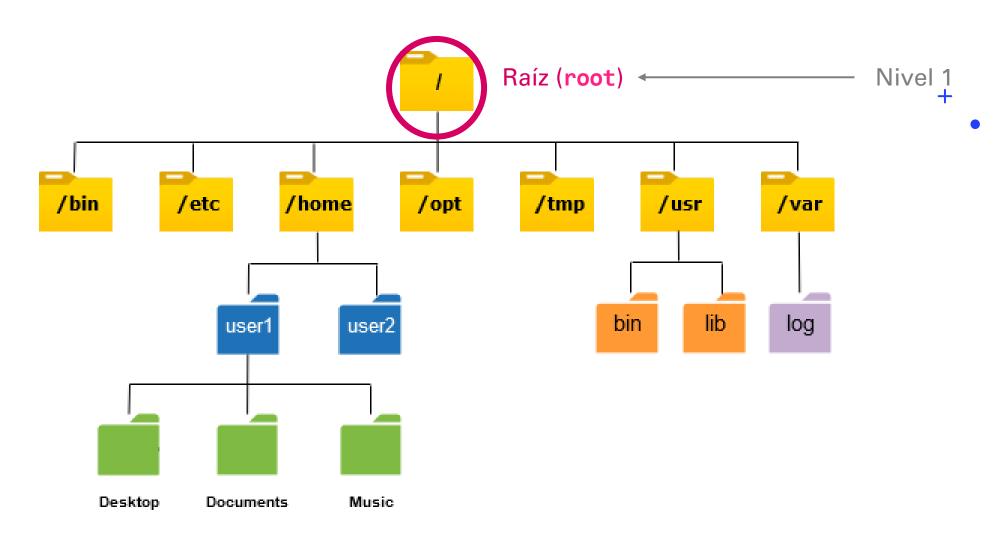




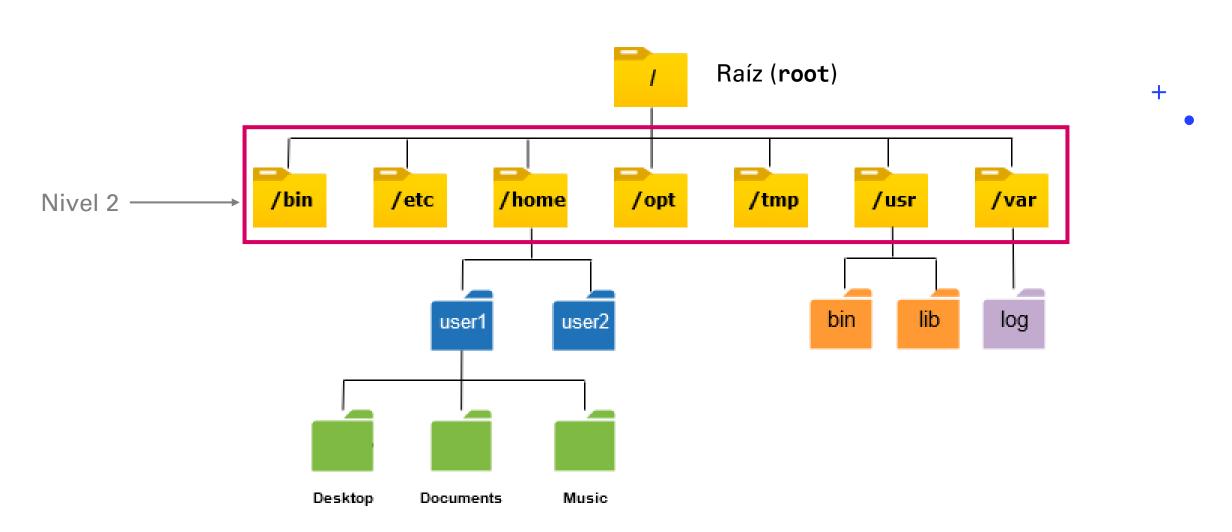




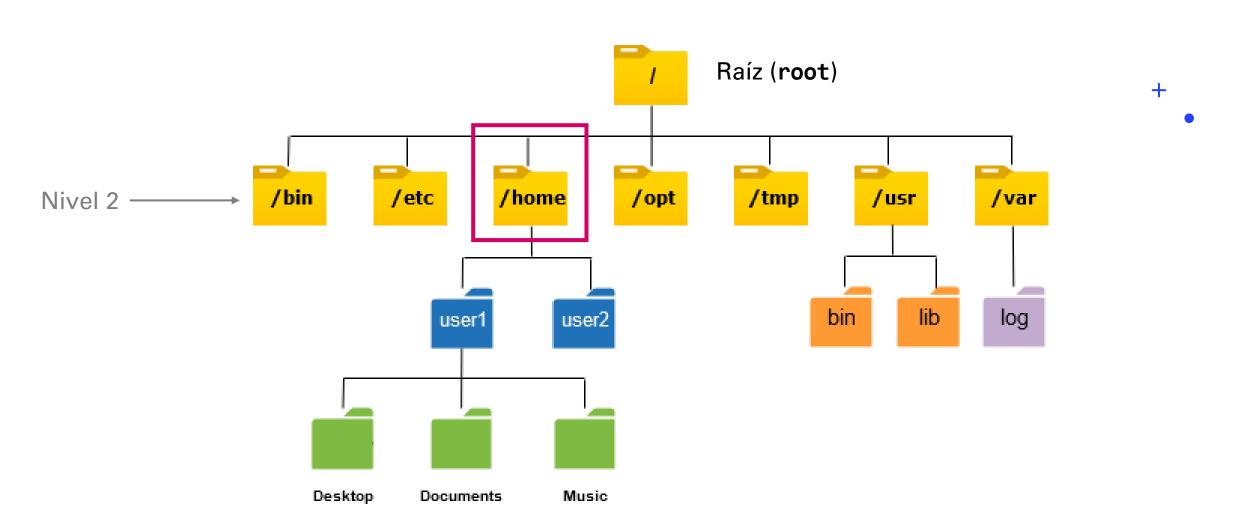




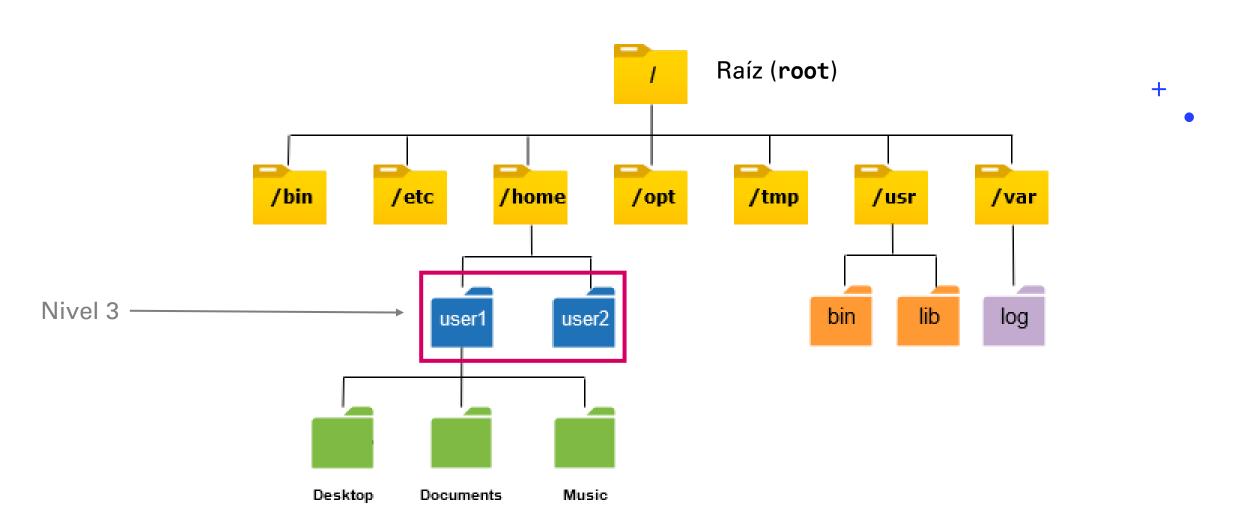




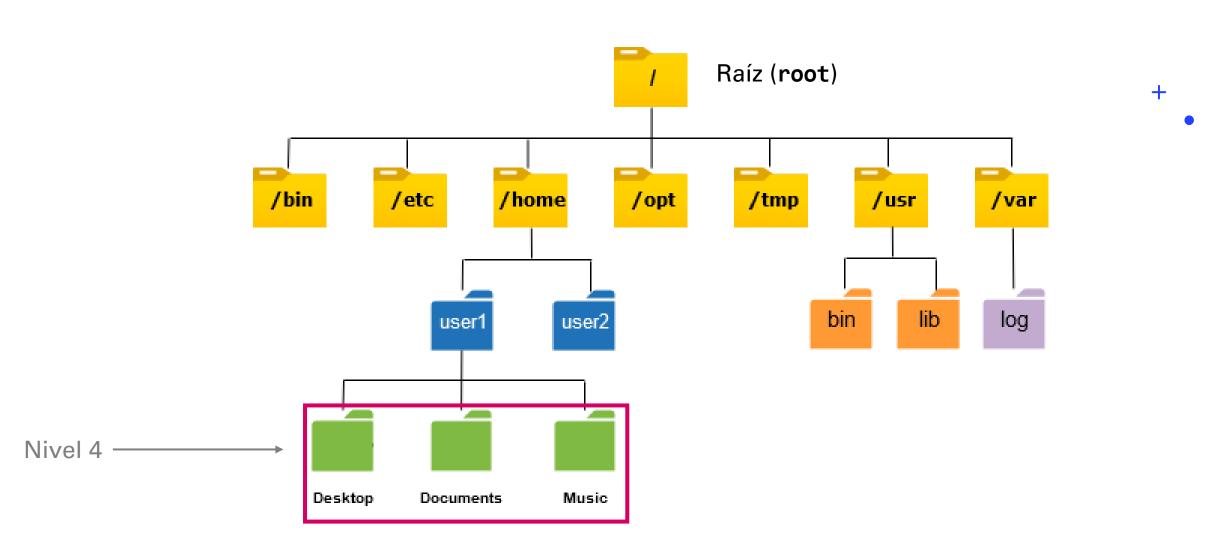




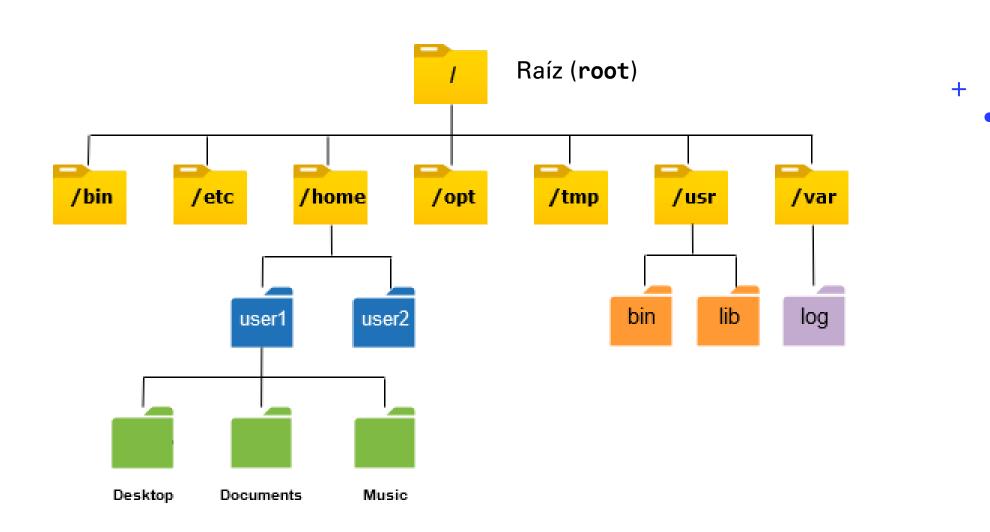












Nivel ...

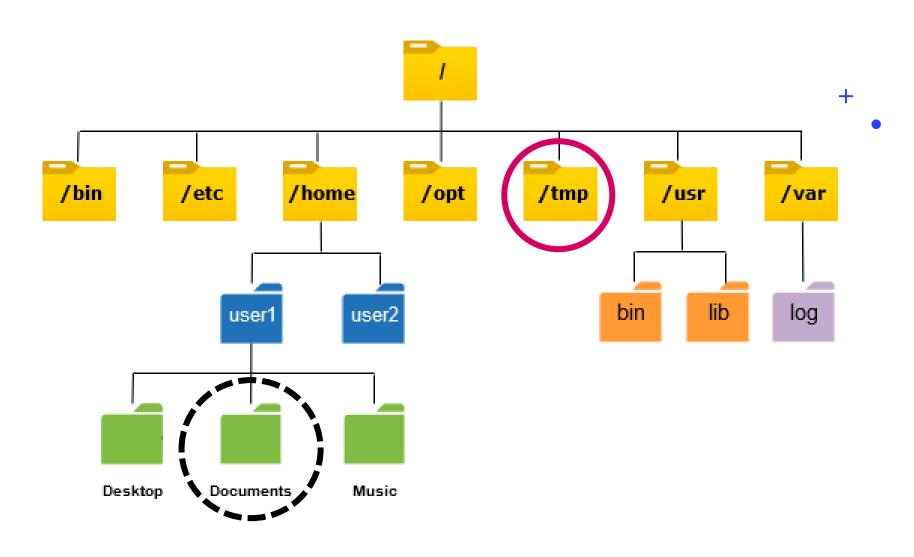
#### Podemos recorrer el árbol de archivos con el comando cd



```
user@whereami $ cd <path>
```

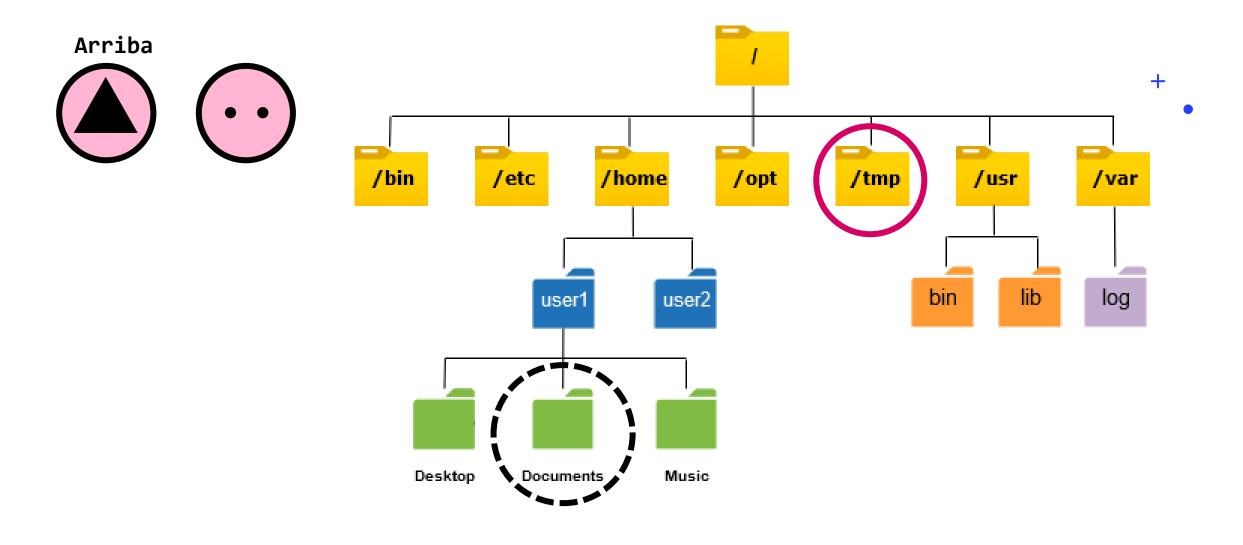
## En unix tenemos dos "movimientos" para construir los paths





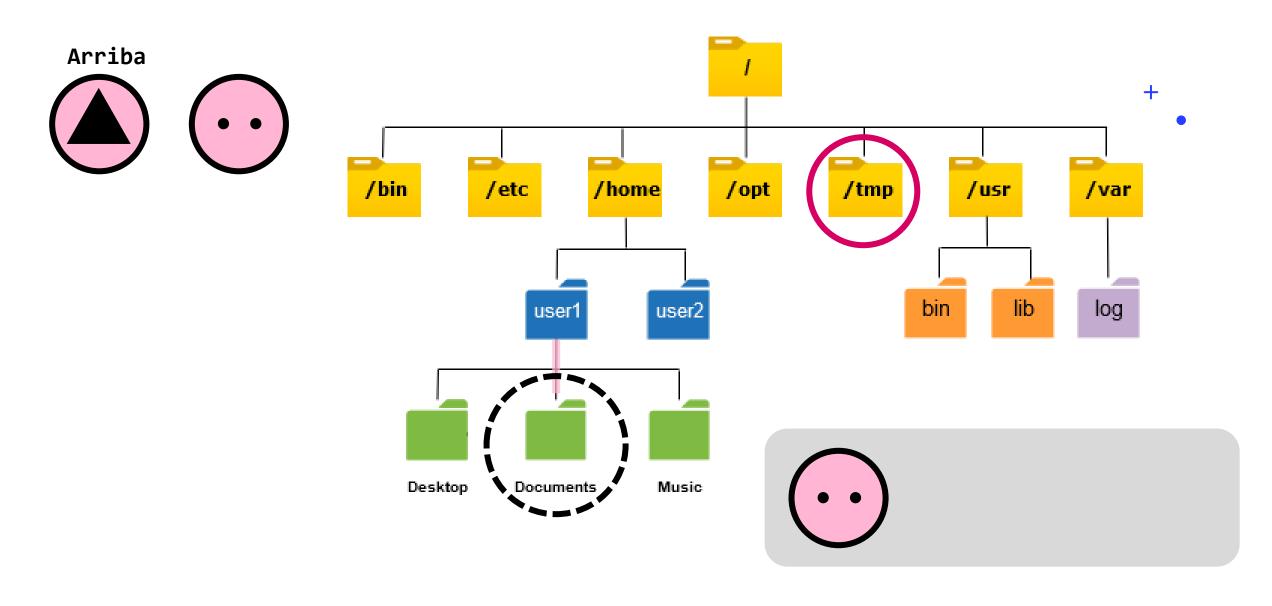
#### En unix tenemos dos "movimientos" para construir los paths





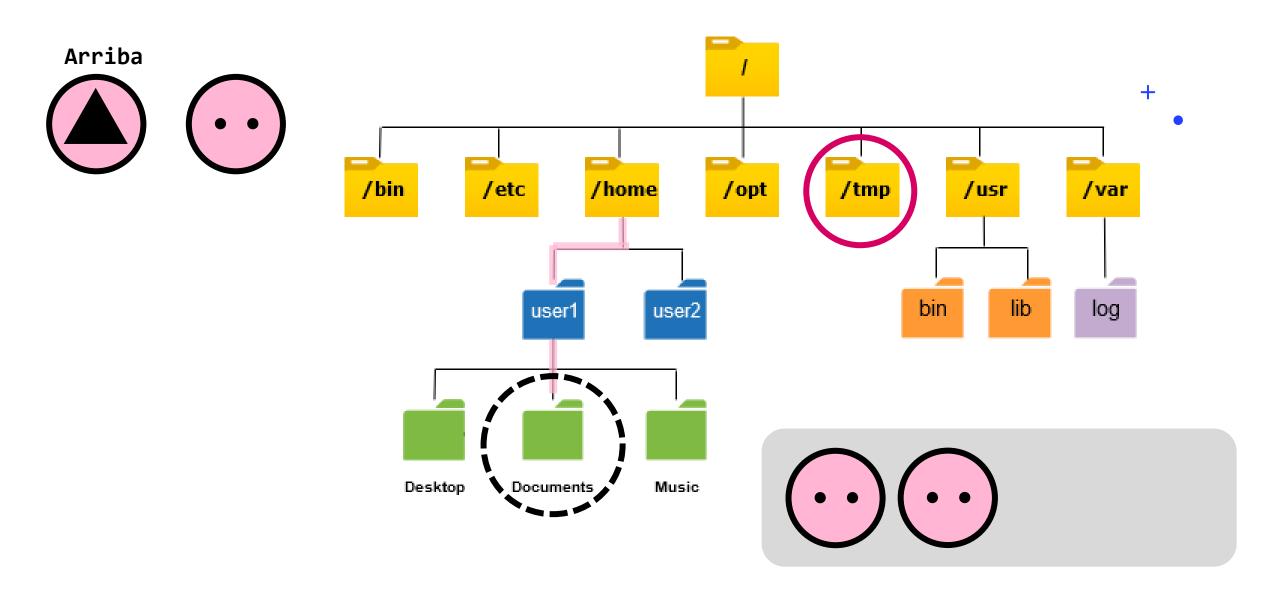
## En unix tenemos dos "movimientos" para recorrer el árbol de archivos





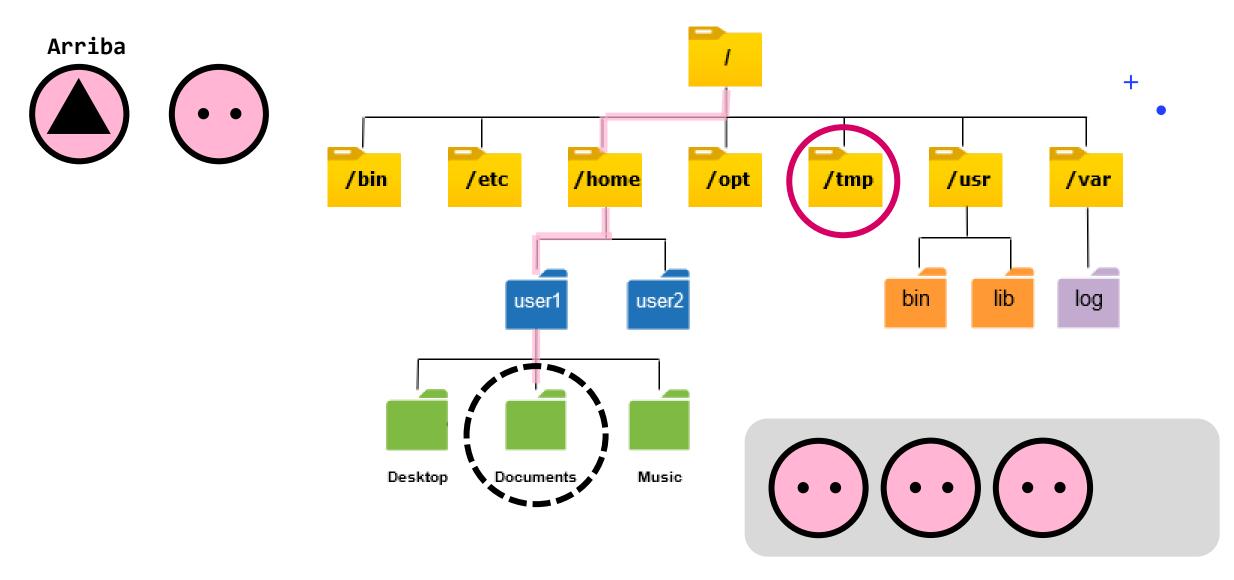
## En unix tenemos dos "movimientos" para recorrer el árbol de archivos





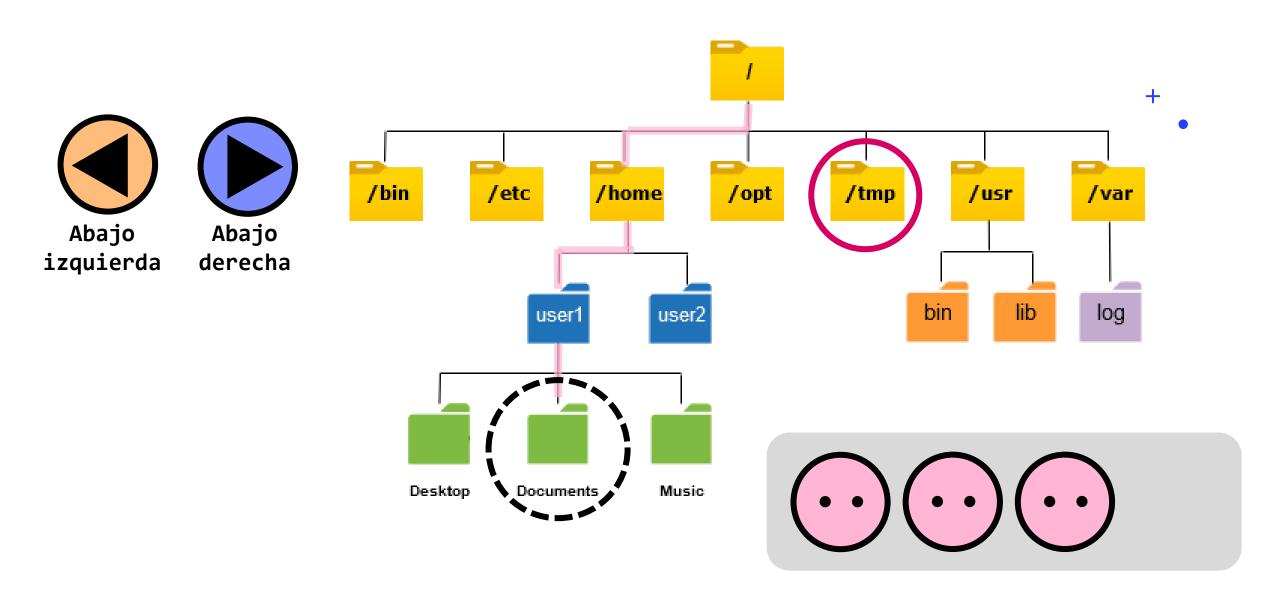
#### En unix tenemos dos "movimientos" para recorrer el árbol de archivos





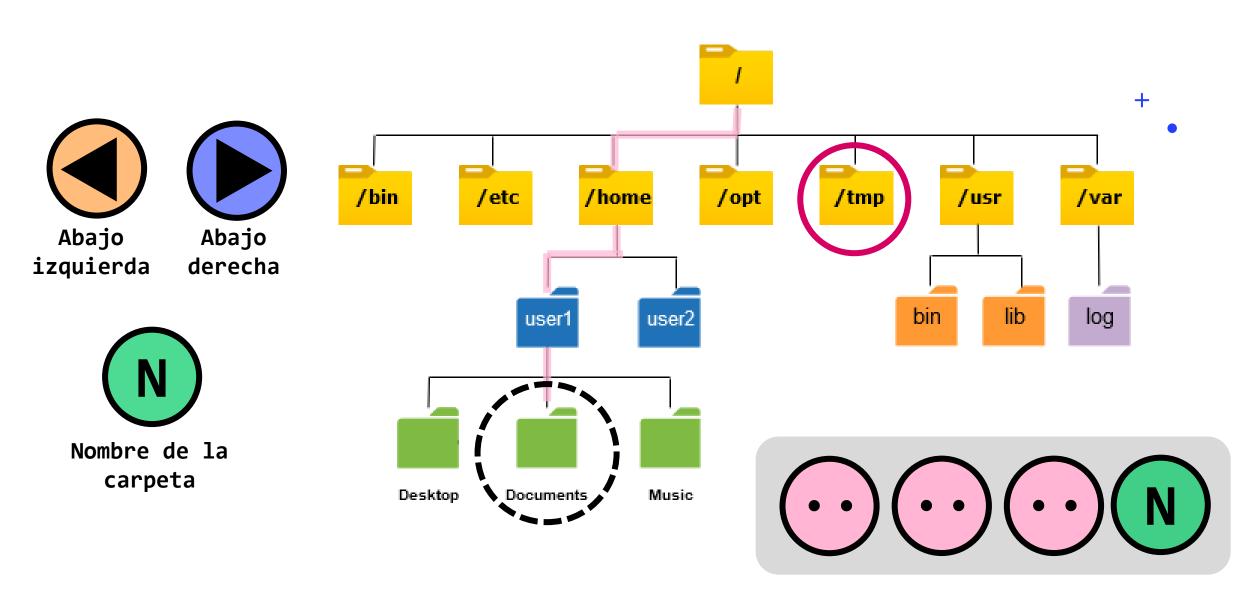
#### En unix tenemos dos "movimientos" para recorrer el árbol de archivos





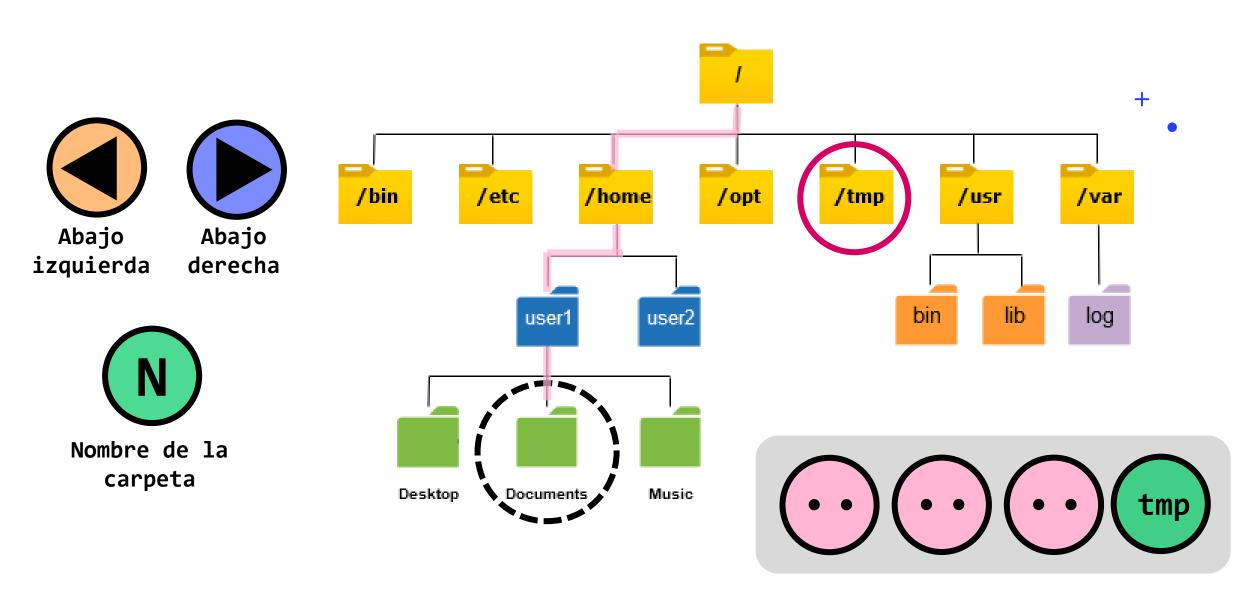
#### En unix tenemos dos "movimientos" para recorrer el árbol de archivos



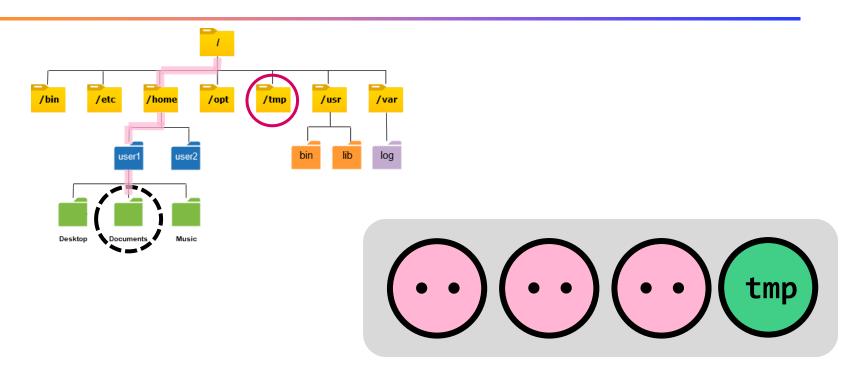


#### En unix tenemos dos "movimientos" para recorrer el árbol de archivos

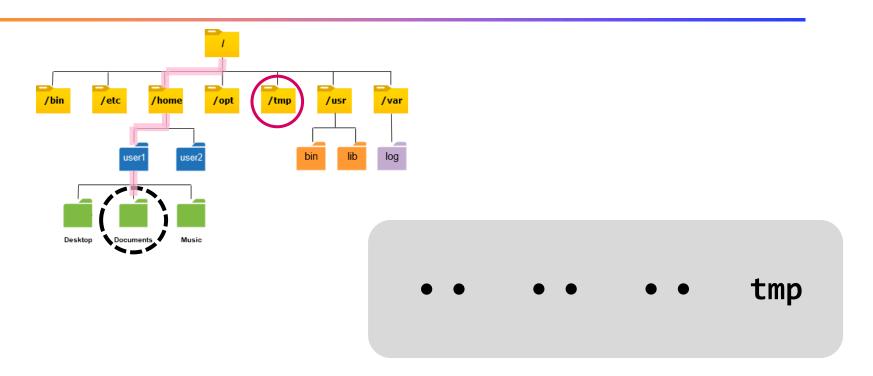




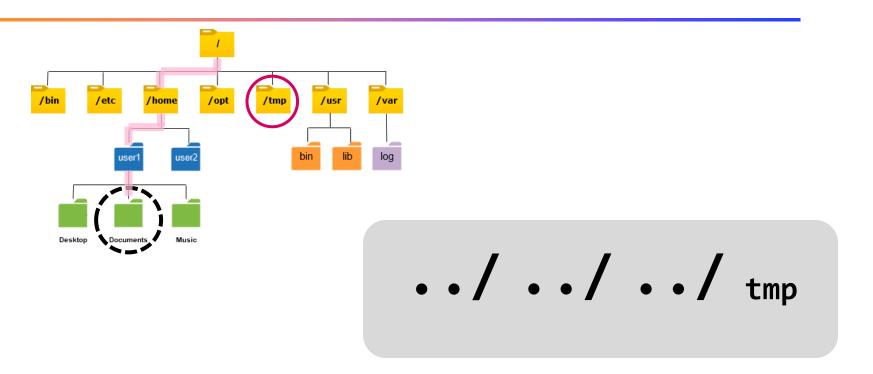




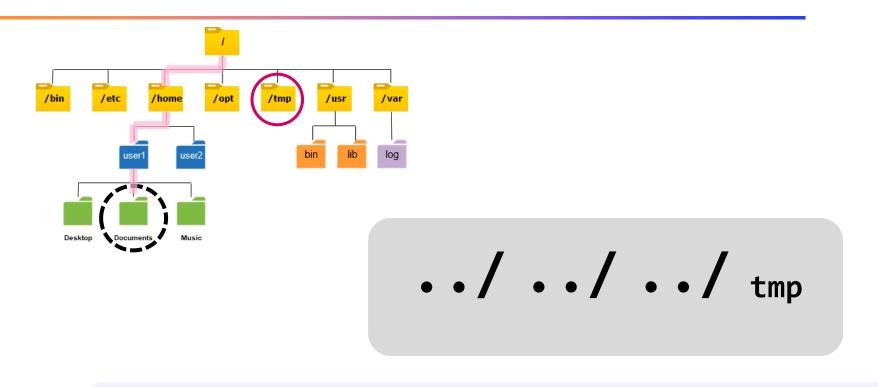






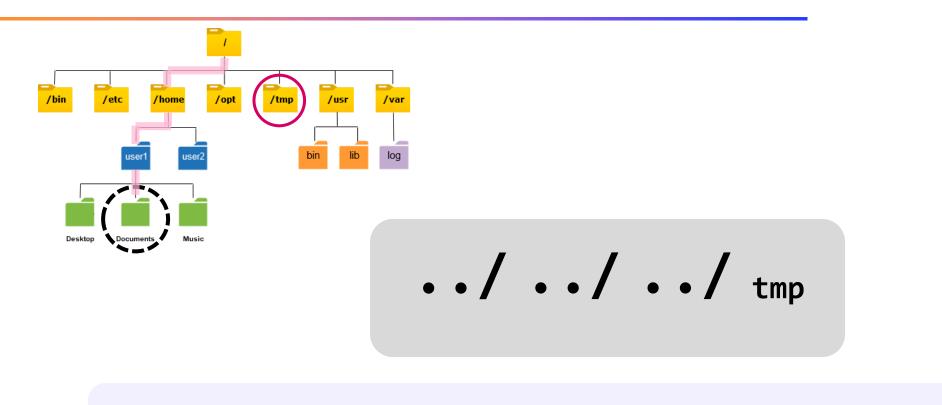




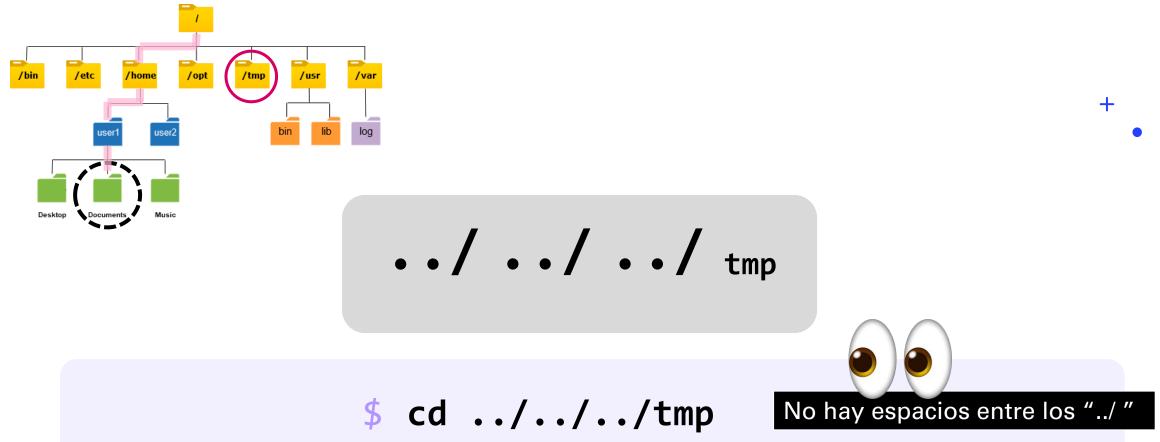


\$ cd <path>

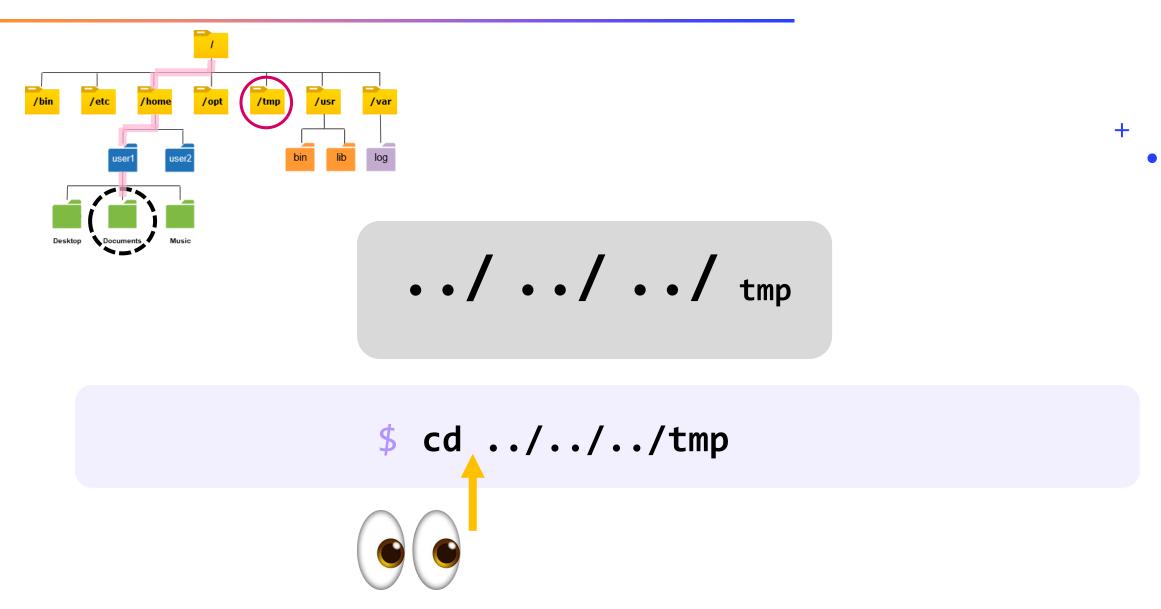




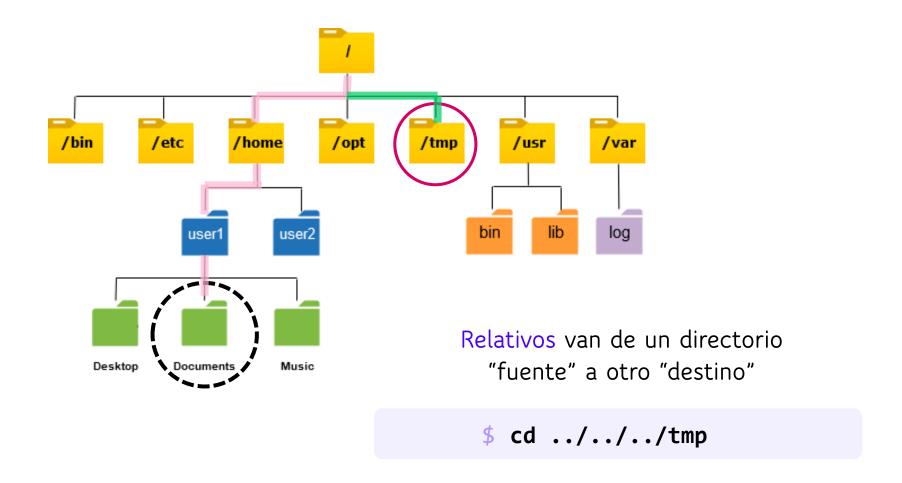




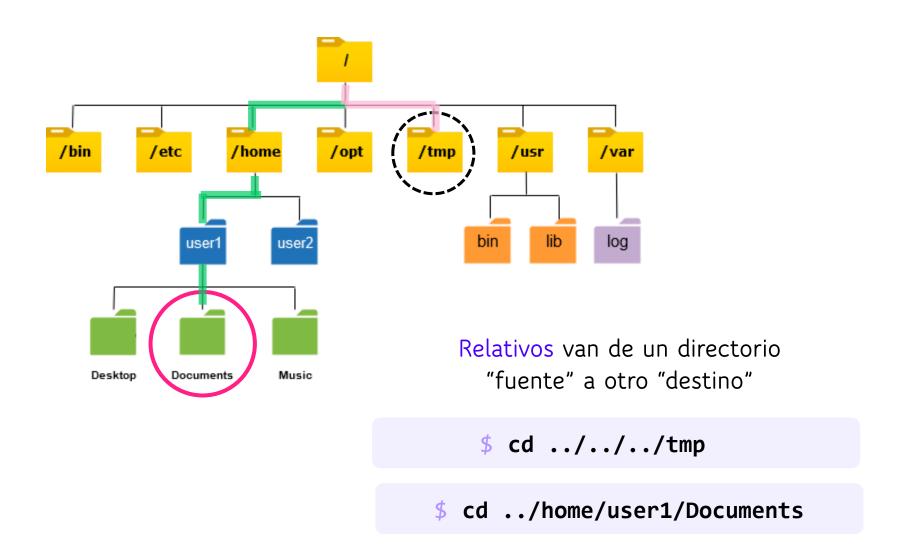




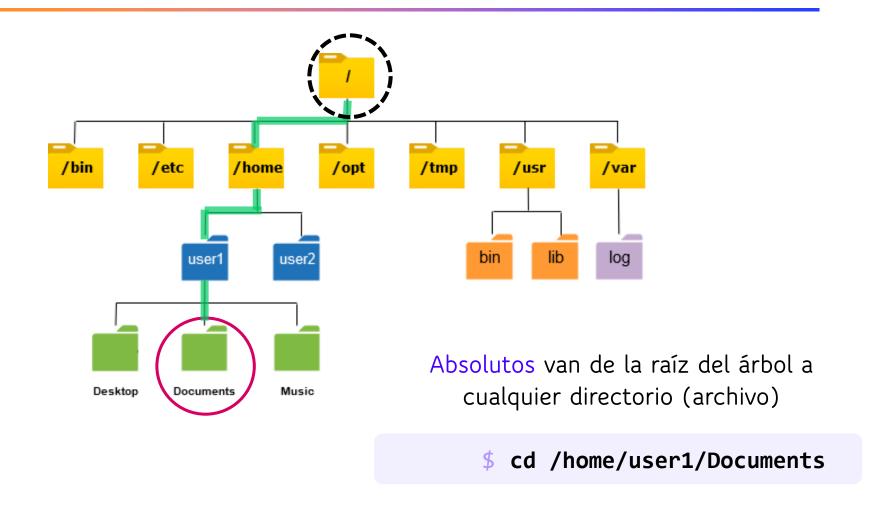




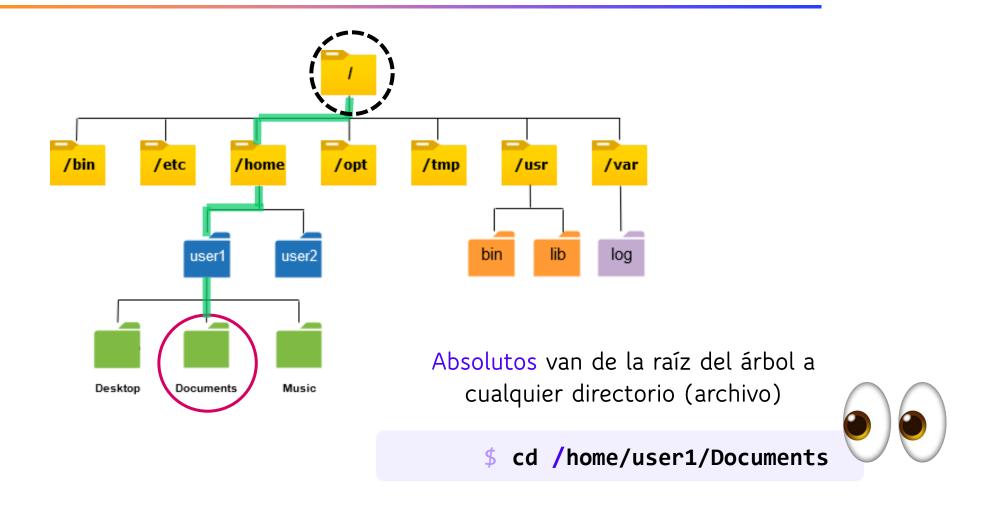




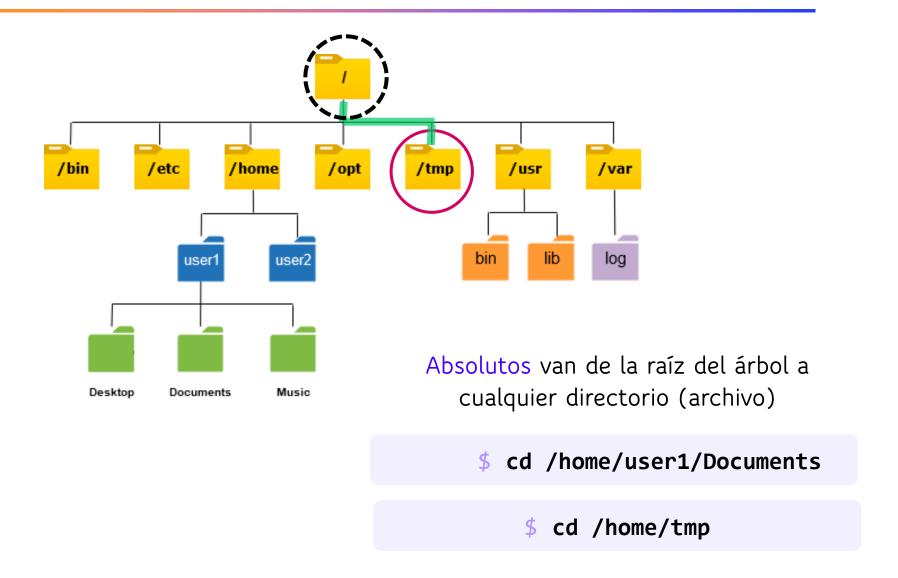












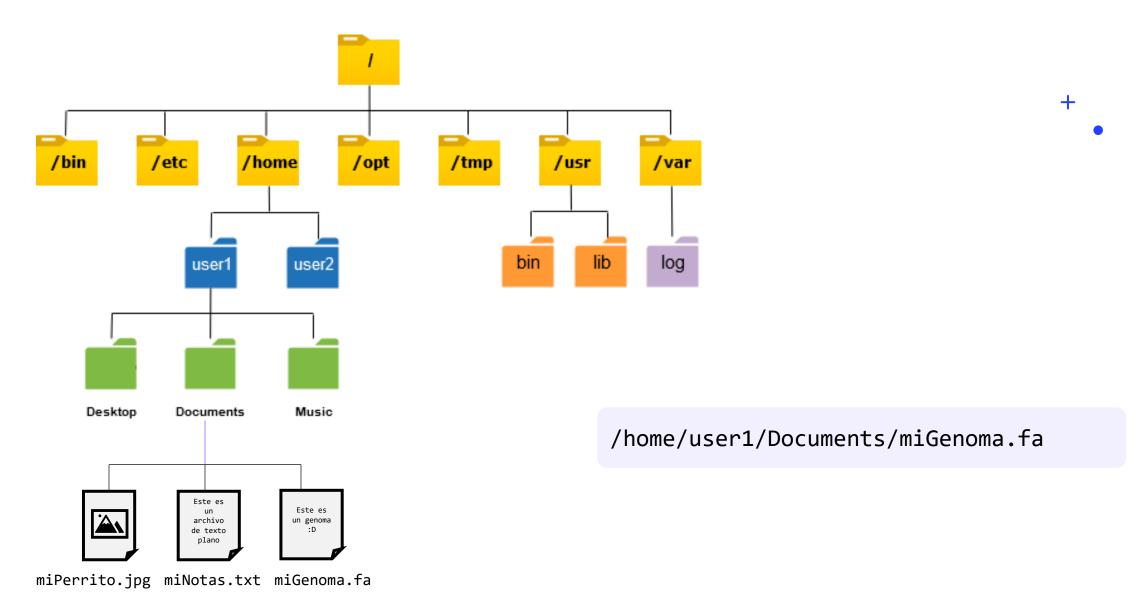
### En Unix hay varios tipos de archivos:



- 1. Archivos regulares
- 2. Directorios
- 3. Links simbólicos

#### Cada directorio contiene: directorios y archivos







## ¡Manos a la obra! 🔆

```
$ pwd
print working directory
                 $ ls [-a,-1,-h] <path>
     list
change directory $ cd <path>
                 $ mv <pathDeDónde> <pathADonde>
     move
                 $ cp [-r] <pathDeQué> <pathADonde>
     copy
                 $ rm [-rf] <path>
    remove
   Caracteres especiales: . . (subir) . (aquí) ~ (absPathMiHome)
```



# TRABAJANDO CON ARCHIVOS





## ¡Manos a la obra! 🐥



```
$ head [-n] <path>
head
             $ tail [-n] <path>
tail
             $ more <path>
more
less
             $ less <path>
             $ nano <path>
nano
             $ sort [-r,n] <path>
sort
             $ uniq [-c] <path>
uniq
             $ cut [-d,f] <path>
cut
             $ grep [-i,v,f] <path1> <path2>
grep
   Caracteres especiales: > (salida) >> (append)
```



# COMPOSICIÓN DE COMANDOS





A veces necesitamos aplicar varios comandos a un archivo inicial para obtener el resultado que necesitamos

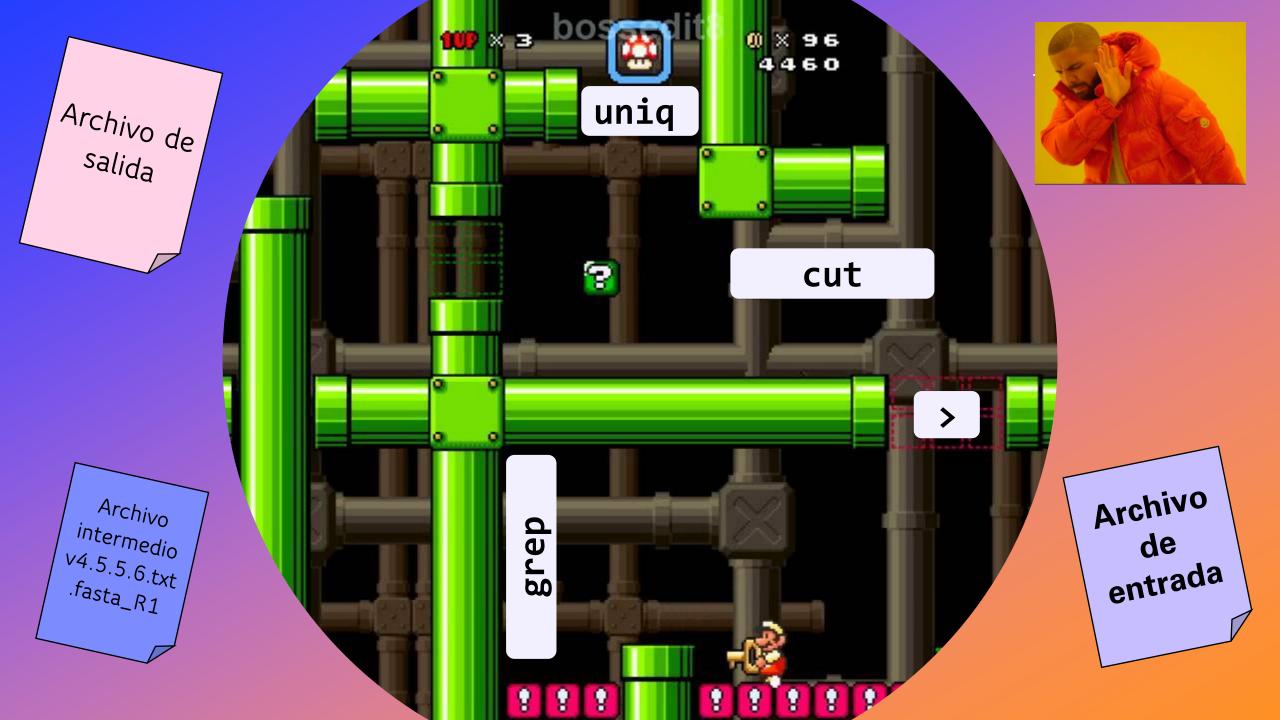


A veces necesitamos aplicar varios comandos a un archivo inicial para obtener el resultado que necesitamos



# Tuberías o pipes 🐥





Archivo de entrada





Archivo de salida

Archivo de entrada







## Programar es como hacer limonada (pt #2)

- 1. Llenar la jarra con agua
- 2. Cortar a la mitad los limones
- 3. Exprimir los limones en la jarra
- 4. Agregar 1 cucharadas de azúcar a la jarra
- 5. Revolver
- 6. Agregar hielos a la jarra





## Programar es como hacer limonada (pt #2)

- 1. Exprimir los limones en la jarra
- 2. Agregar 1 cucharadas de azúcar a la jarra
- 3. Revolver
- 4. Agregar hielos a la jarra
- 5. Cortar a la mitad los limones
- 6. Llenar la jarra con agua





# ¡Manos a la obra! 🐥

Caracteres especiales: (pipe)

# +.¡GRACIAS!

\$ exit



