

LABORATORIO DE SISTEMAS OPERATIVO

Lab 1: Linux básico y GCC

Objetivos de la práctica

- Introducir los conceptos mas básicos del manejo de consola necesarios para la realización de las tareas básicas de programación.
- Introducir en el manejo del gcc como herramienta para la compilación de código.

Criterios de evaluación para esta guía

- Todas las preguntas tienen el mismo valor
- $\text{Nota} = 5 * (\text{Correctas} / \text{Totales})$

Introducción al manejo del GCC

REQUISITOS

Pc con Linux



Compilador

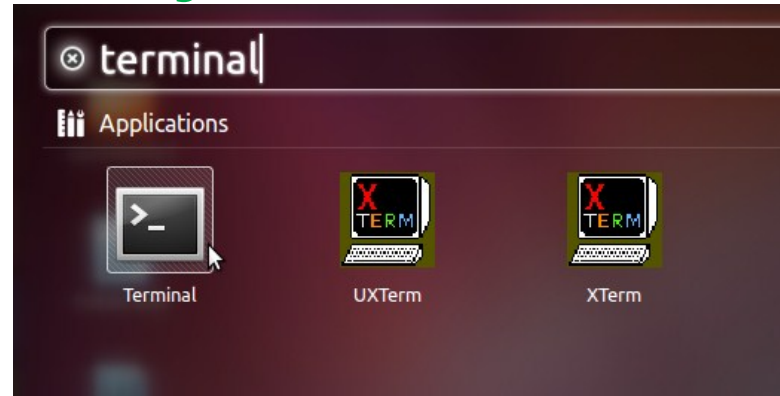


Editor de texto o IDE



Introducción al manejo del GCC

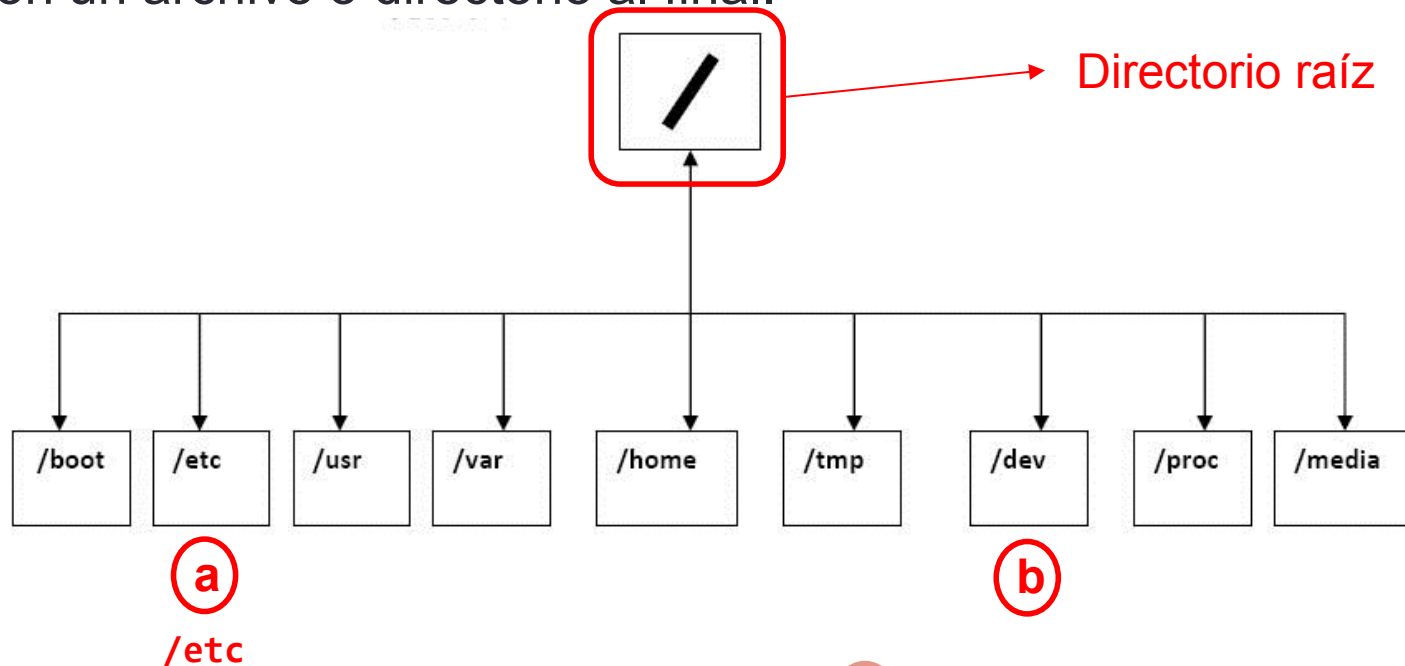
TERMINAL



MANEJO BÁSICO DE LA TERMINAL

RUTAS

Secuencia de directorios anidados separados con el carácter slash (/) con un archivo o directorio al final.

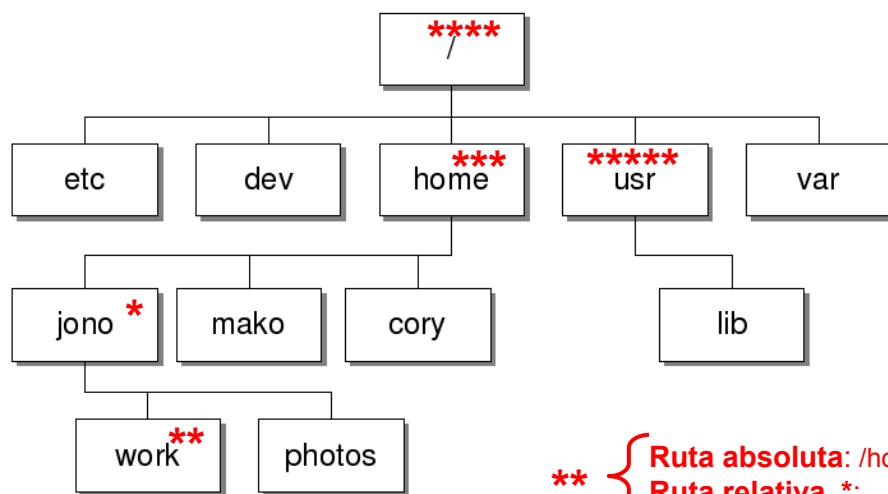


1 ¿Cuál es la ruta de b?

MANEJO BÁSICO DE LA TERMINAL

RUTAS

- **Absolutas:** Rutas vistas desde el directorio raíz.
- **Relativas:** Rutas vistas desde un directorio en particular.



* { Ruta absoluta: /home/jono

** { Ruta absoluta: /home/jono/work
Ruta relativa *: ../work ↔ work

*** { Ruta absoluta: /home
Ruta relativa a *: ..

**** { Ruta absoluta: /
Ruta relativa a *: ../../..

DIRECTORIOS ESPECIALES

- `./`: Directorio actual.
- `../`: Directorio padre del directorio en el cual me encuentro ubicado.

2

¿Cuál es la ruta de ***** relativa a *?

?

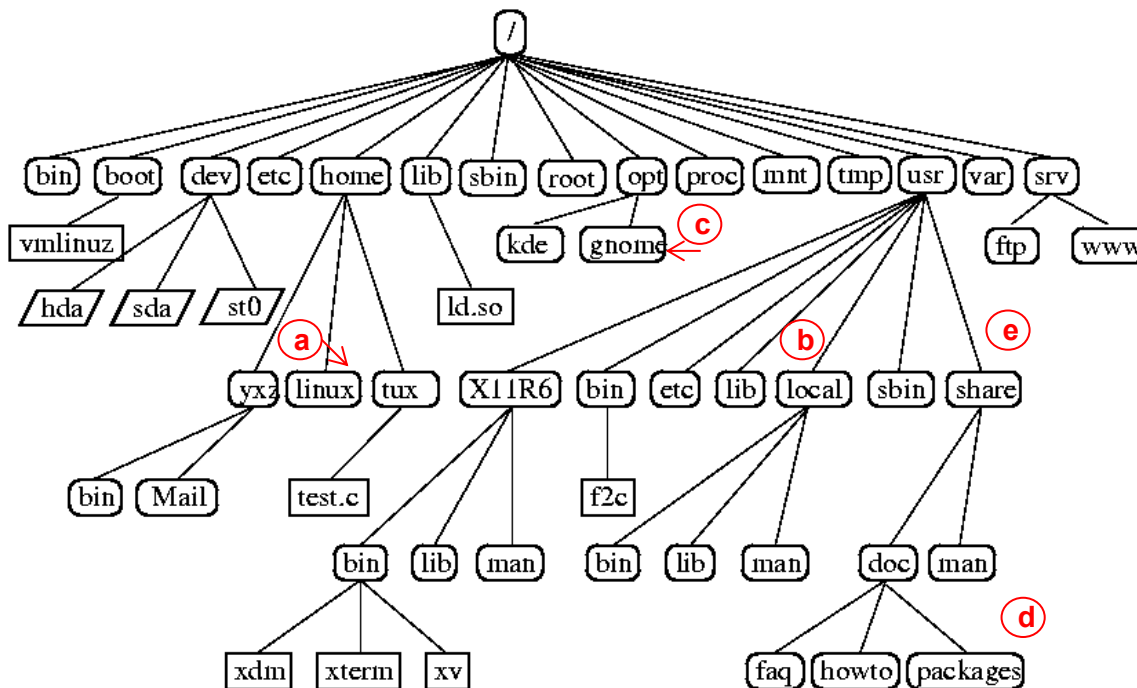
3

¿Cuál es la ruta de ***** relativa a **?

?

MANEJO BÁSICO DE LA TERMINAL

RUTAS (EJERCICIOS DE REFUERZO)



4

¿Cuál es la ruta de **c** relativa a **a**?



5

¿Cuál es la ruta de **d** relativa a **b**?



6

¿Cuál es la ruta de **b** relativa a **d**?



7

¿Cuál es la ruta de **d** relativa a **e**?



MANEJO BASICO DE LA TERMINAL

ASPECTOS BÁSICOS A DOMINAR EN CONSOLA

Creación de directorios (mkdir)

```
tigarto@my-fuck-pc: ~
tigarto@my-fuck-pc:~/Documents$ cd ..
tigarto@my-fuck-pc:~$ ls
Desktop  Downloads  driversmodules  Music  Public  Videos
Documents  drivers  examples.desktop  Pictures  Templates
tigarto@my-fuck-pc:~$ mkdir test
tigarto@my-fuck-pc:~$ ls
Desktop  Downloads  driversmodules  Music  Public  test
Documents  drivers  examples.desktop  Pictures  Templates  Videos
tigarto@my-fuck-pc:~$
```

Navegando a través del sistema de archivos (cd)

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~$ cd Documents/
tigarto@my-fuck-pc:~/Documents$ pwd
/home/tigarto/Documents
tigarto@my-fuck-pc:~/Documents$
```

Listado archivos y directorios (ls)

```
tigarto@my-fuck-pc: ~
tigarto@my-fuck-pc:~$ ls
Desktop  drivers  Music  Templates
Documents  driversmodules  Pictures  Videos
Downloads  examples.desktop  Public
tigarto@my-fuck-pc:~$
```

Verificación de la ruta en la cual me encuentro ubicado

```
tigarto@my-fuck-pc: ~
tigarto@my-fuck-pc:~$ pwd
/home/tigarto
tigarto@my-fuck-pc:~$
```

Usuario actual (users)

```
tigarto@my-fuck-pc: ~
tigarto@my-fuck-pc:~$ users
tigarto
tigarto@my-fuck-pc:~$ 3~
```

MANEJO BÁSICO DE LA TERMINAL

COMANDOS CD Y PWD

cd <dir>

Cambia el directorio actual a dir

cd ..

Retorna al directorio previo actual

pwd

Despliega el directorio actual
(directorio de trabajo)

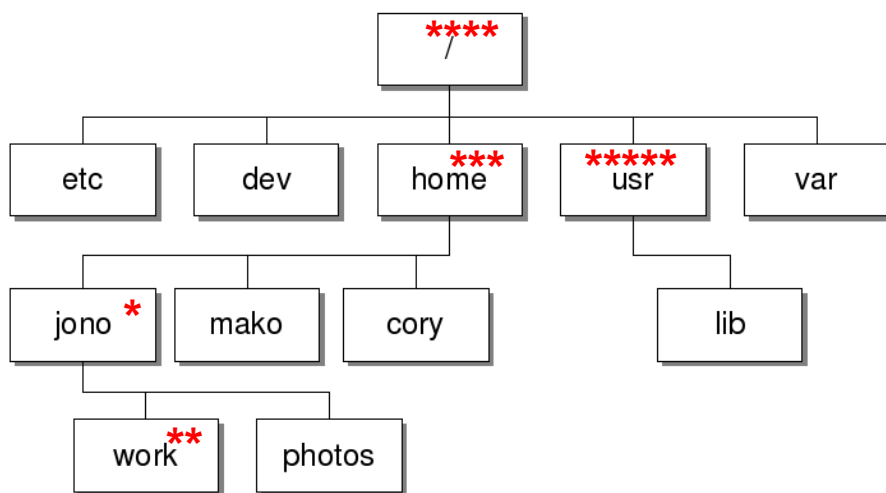


Es común usar `pwd` después del comando `cd` para verificar que si nos encontremos en la ruta actual deseada.

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~$ cd Documents/
tigarto@my-fuck-pc:~/Documents$ pwd
/home/tigarto/Documents
tigarto@my-fuck-pc:~/Documents$
```

MANEJO BÁSICO DE LA TERMINAL

COMANDOS CD Y PWD – EJEMPLOS Y PREGUNTAS



8

?

¿Cuál es el resultado de ejecutar el comando `pwd` cuando estamos ubicados en ****?

9

?

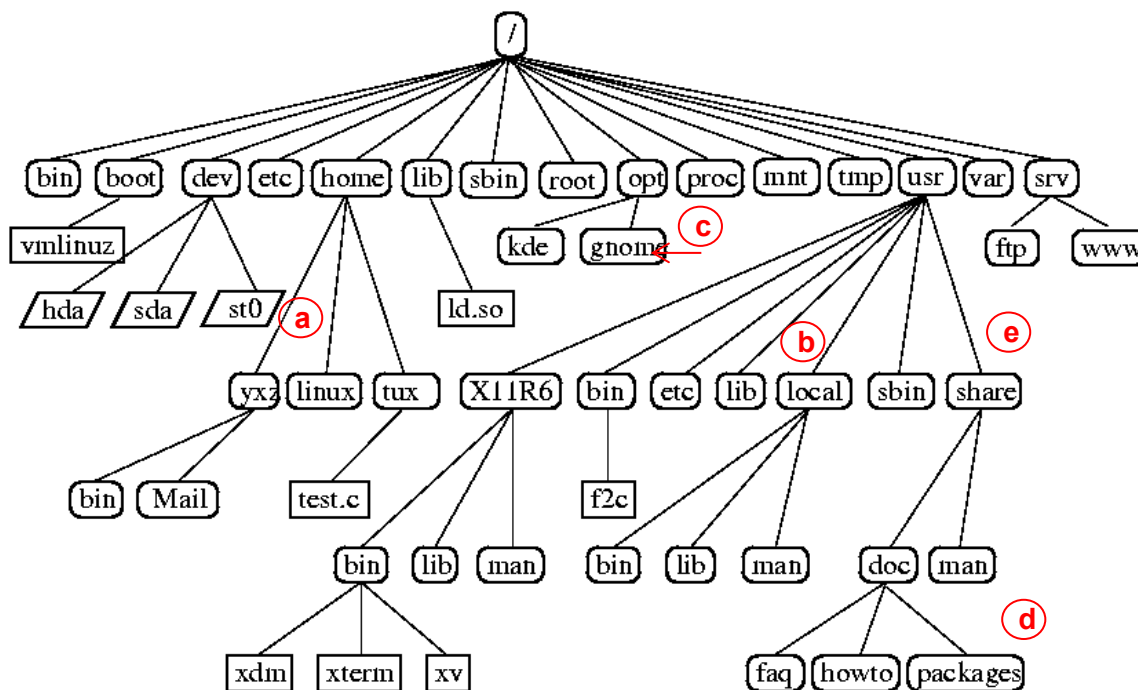
¿Cuál es el resultado de ejecutar el comando `pwd` cuando estamos ubicados en **?

¿Cuál es el comando para pasar de * a *****?

- Empleando ruta absoluta: `cd /usr`
- Empleando ruta relativa: `cd ../../usr`

MANEJO BÁSICO DE LA TERMINAL

COMANDOS CD Y PWD – EJERCICIOS DE REFUE



10

¿Cuál es el resultado de ejecutar el comando pwd cuando estamos ubicados en **d**?

11

¿Cuál es comando para pasar de **e** a **d** (de ambas formas)?

12

¿Cuál es comando para pasar de **c** a **b** (de ambas formas)?

MANEJO BÁSICO DE LA TERMINAL

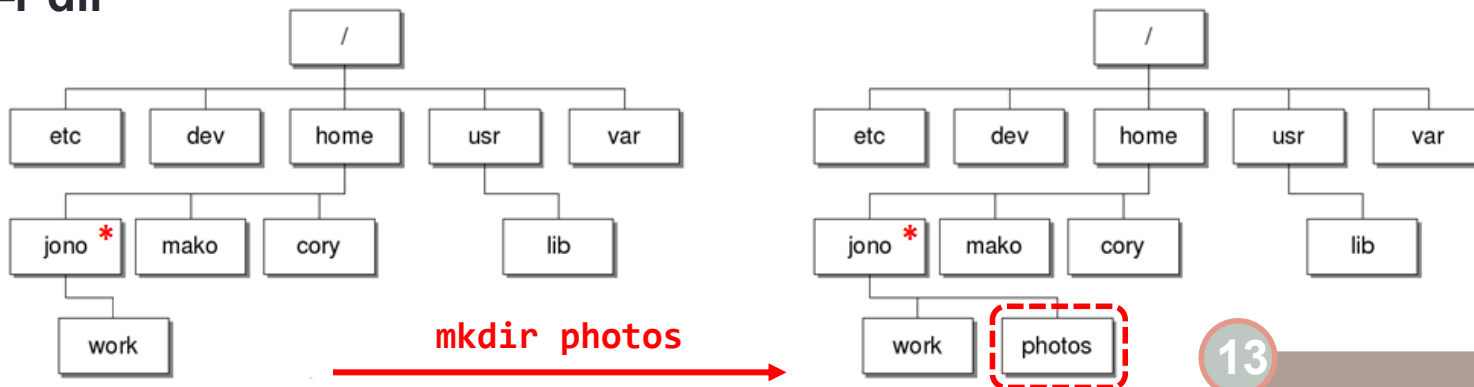
CREANDO Y REMOVIENDO DIRECTORIOS

`mkdir dir1 dir2 dir3 ...`

Crea directorios con los nombres dados

`rmdir dir1 dir2 dir3 ...`

Remueve los directorios dados. Solo funciona para directorios (no para archivos) cuando estos son vacíos. Si un directorio no esta vacío se usa la opción **-r**, así: **rm -r dir**



¿Cual es el comando para pasar crear una carpeta llamada lab dentro de var?

Forma 1: Cambiando primero la ubicación

1. `cd /var`
2. `mkdir lab`

Forma 2:

- Permaneciendo en ***
1. `mkdir /var/lab`

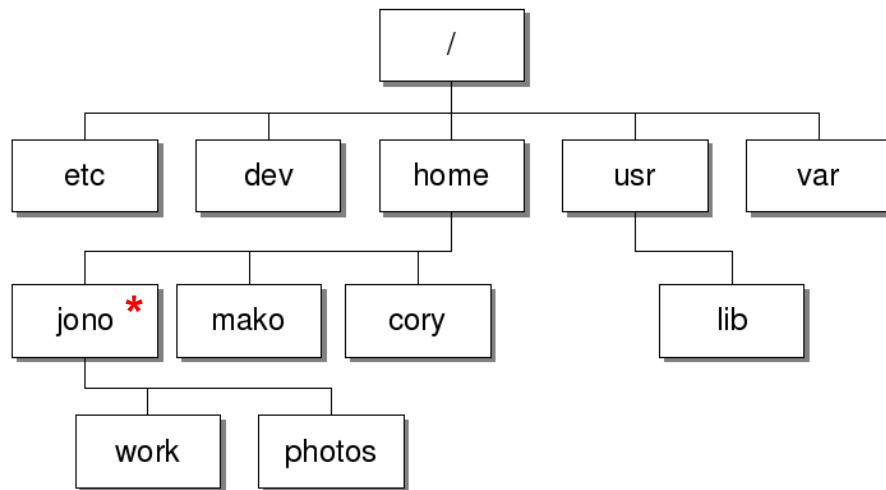
13

Muestre otra manera por medio de rutas relativas



MANEJO BÁSICO DE LA TERMINAL

CREANDO Y REMOVIENDO DIRECTORIOS - PREGUNTA



Para responder las preguntas asuma que el directorio actual es el directorio *

16

¿Si el directorio 1 creado en el punto anterior esta vacío con que comando o secuencia de comandos se elimina?

17

¿Cómo se elimina el directorio c creado en 14 si este no esta vacío?

14

¿Cuál es el comando (o secuencia de comandos) para crear 3 directorios llamados **a**, **b** y **c** dentro de **jono**?

15

¿Cuál es el comando (o secuencia de comandos) para crear 2 directorios llamados **1** y **2** dentro de **photos**?

MANEJO BÁSICO DE LA TERMINAL

LISTANDO ARCHIVOS

`ls <opciones> <dir>`

Lista los archivos del directorio `dir`, en orden alfanumerico, exceptuando los archivos que empiezan con el carácter “.” (archivos ocultos). El directorio `dir` es opcional y cuando no aparece los archivos listados son los del directorio actual.

`ls -a <dir>`

Lista todos los archivos incluyendo aquellos que empiezan por el carácter “.” (archivos ocultos).

`ls -l <dir>`

Lista los archivos desplegando con detalle la información asociada a cada uno de estos (tipo, fecha, tamaño, propietario, permisos).

El posible combinar opciones, por ejemplo: **`ls -al`**.

Para mas información puede consultar el manual del comando: **`man ls`**.

MANEJO BÁSICO DE LA TERMINAL

LISTANDO ARCHIVOS

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~/Documents$ ls
informatica1      pantallazos.odt
lab_0_images.doc  Screenshot at 2012-08-08 23:23:15.png
Lab50             Screenshot at 2012-09-26 18:55:23.png
pantallazo.png
tigarto@my-fuck-pc:~/Documents$
```

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~/Documents$ ls -a
.                .oculto
..               pantallazo.png
informatica1     pantallazos.odt
lab_0_images.doc Screenshot at 2012-08-08 23:23:15.png
Lab50            Screenshot at 2012-09-26 18:55:23.png
~lock.lab_0_images.doc#
tigarto@my-fuck-pc:~/Documents$
```

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~/Documents$ ls -l
total 1200
drwxrwxr-x 4 tigarto tigarto 4096 2012-07-30 14:32 informatica1
-rw-rw-r-- 1 tigarto tigarto 628736 2012-09-26 20:30 lab_0_images.doc
drwxrwxr-x 6 tigarto tigarto 4096 2012-08-08 09:17 Lab50
-rw-rw-r-- 1 tigarto tigarto 66632 2012-07-30 14:35 pantallazo.png
-rw-rw-r-- 1 tigarto tigarto 168989 2012-05-01 16:30 pantallazos.odt
```

18

?

¿Cuál es la diferencia entre las 2 figuras?

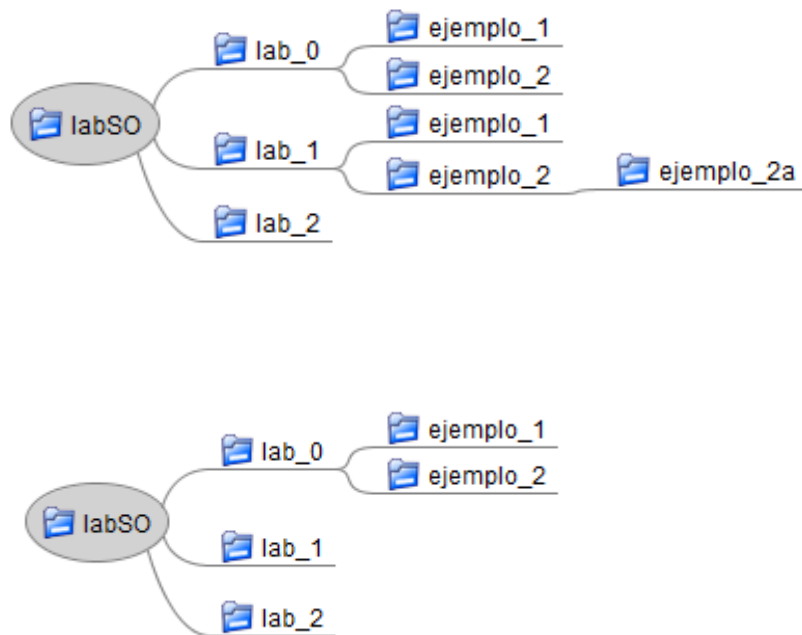
19

?

Dada la siguiente figura como se lograría que el archivo .oculto aparezca desplegado con su información detallada

MANEJO BÁSICO DE LA TERMINAL

EJERCICIO COMPLETO

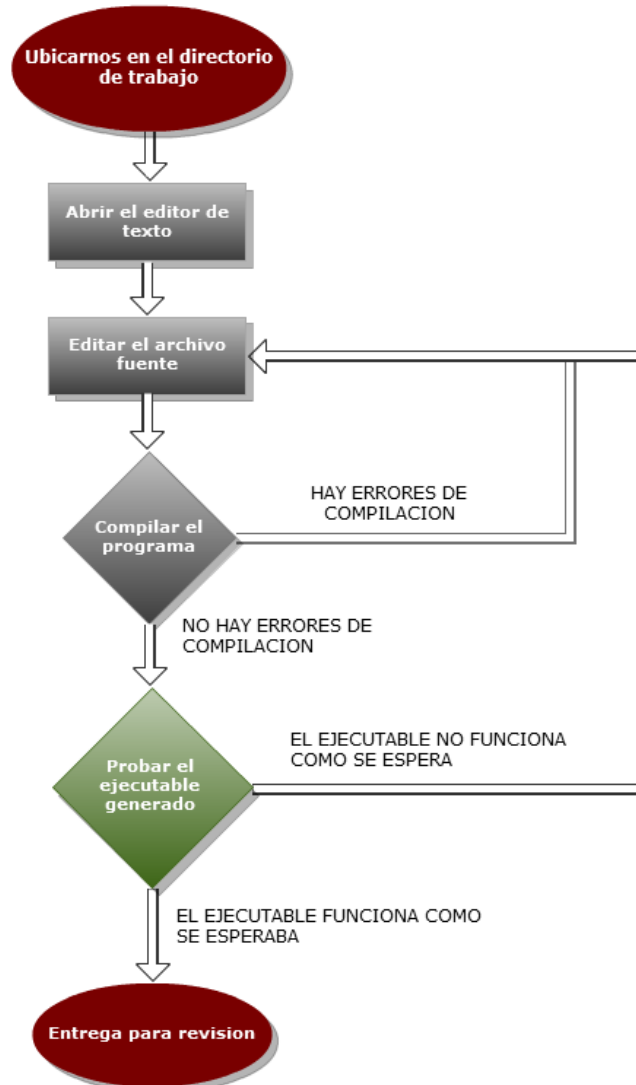


Si el comando **tree** no está disponible, se debe instalar con:
sudo apt-get install tree

20

1. Abra una terminal.
2. Averigüe el usuario del equipo dentro del que se encuentra.
3. Ubíquese en la carpeta **Documentos** (o **Documents**) y cree allí la jerarquía de directorios mostradas en la figura de arriba.
4. Ingrese al directorio **labSO**. Verifique esta jerarquía con el comando **tree**.
5. Elimine los directorios **ejemplo_1** y **ejemplo_2** dentro de la carpeta **lab_1**, después de hacer esto la jerarquía de directorios debe quedar como se muestra en la figura de abajo.
6. **Ahora que se familiarizo un poco mas con la consola ya esta listo para compilar su primer programa.**

COMPILANDO EL PRIMER PROGRAMA



COMPILANDO EL PRIMER PROGRAMA

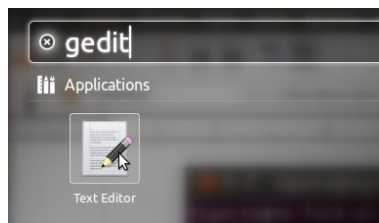
Ubicarnos en el directorio de trabajo

```
tigarto@my-fuck-pc: ~/Documents/LabSO
tigarto@my-fuck-pc:~/Documents/LabSO$
```

Abrir el editor de texto

Se abre el editor de texto o IDE favorito: gedit, kwrite, emacs, eclipse, netbeans.

Desde el menú del SO



Desde la consola (Con ejecución en background o

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~/Documents$ gedit
```

```
tigarto@my-fuck-pc: ~/Documents
tigarto@my-fuck-pc:~/Documents$ gedit &
[1] 2047
tigarto@my-fuck-pc:~/Documents$
```

COMPILANDO EL PRIMER PROGRAMA

Editar el archivo fuente

Una vez abierto el editor, se codifica el programa y se guarda en la ruta deseada, el nombre de este debe tener extensión

.c.

Después de que el programa ha sido codificado y guardado se verifica por seguridad que si se encuentre en el directorio de trabajo.

```
ejemplo1.c (~/.Documents/LabSO) - gedit
Open Save Undo
ejemplo1.c
#include <stdio.h>
#include <unistd.h>

//Comentario de una linea
char nombre[50];

/*
Comentario de
varias lineas
*/

int main() {
    int pid = getpid();
    printf("Ingrese su nombre por favor: ");
    scanf("%s", nombre);
    printf("Hola %s, su id de proceso es %d", nombre, pid);
    return 0;
}
```

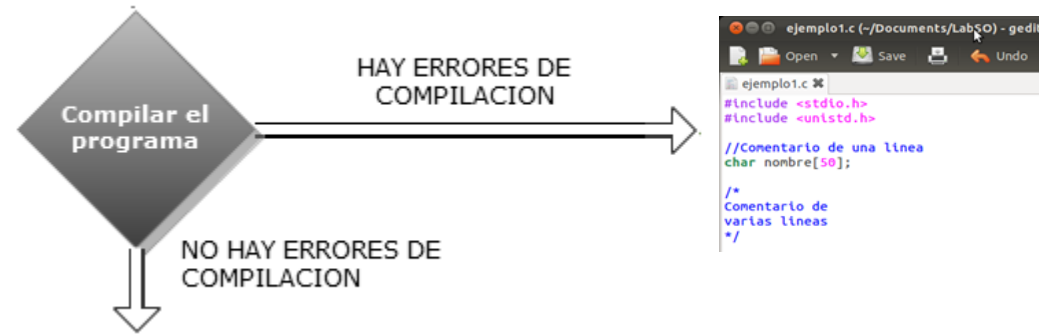
```
tigarto@my-fuck-pc: ~/.Documents/LabSO
tigarto@my-fuck-pc:~/.Documents/LabSO$ ls
ejemplo1.c
tigarto@my-fuck-pc:~/.Documents/LabSO$
```

COMPILANDO EL PRIMER PROGRAMA

La compilación se hace empleando el comando gcc

```
tigarto@my-fuck-pc: ~/Documents/LabSO
tigarto@my-fuck-pc:~/Documents/LabSO$ gcc ejemplo1.c -o exe1
tigarto@my-fuck-pc:~/Documents/LabSO$
```

Después de compilar, si la compilación fue exitosa es recomendable verificar que el ejecutable fue correctamente generado para lo cual se emplea el comando **ls**.



gcc archivoFuente -o nombreEjecutable

Uso mas frecuente del comando gcc para compilar y enlazar un archivo fuente (archivo .c) de modo que se genera un ejecutable con nombre dado por nombreEjecutable. Para mas información puede consultar:

http://www.mhe.es/universidad/informatica/8448198441/archivos/apendice_general_1.pdf

COMPILANDO EL PRIMER PROGRAMA

Probar el ejecutable generado

Para probar el ejecutable generado se coloca el nombre del ejecutable antecedido por `./`, en este ejemplo como el ejecutable se llama **exe1**, entonces para su ejecución tecleamos **./exe1**.

```
tigarto@my-fuck-pc: ~/Documents/LabSO
tigarto@my-fuck-pc:~/Documents/LabSO$ ./exe1
Ingrese su nombre por favor: Ramon
Hola Ramon, su id de proceso es 1990tigarto@my-fuck-pc:~/Documents/LabSO$
```

21

Como se debe ejecutar el comando `gcc` si se tiene un archivo fuente llamado `main.c` y se quiere generar un ejecutable llamado `myexe`.

22

Como será el comando para probar el ejecutable anteriormente generado.

23

¿Qué sucede si se ejecuta el comando `gcc` sin la opción `-o`?

COMPILANDO EL PRIMER PROGRAMA

Entrega para revision

GET TO
KNOW:

YOUR ADVISOR'S
NEGATION FIELD

A Professor's Negation Field is the unexplained phenomenon whereby mere spatial proximity to an experimental set-up causes all working demonstrations to fail, despite the apparent laws of Physics or how many times it worked right before he/she walked into the room.

BEWARE
ALSO:

The Sphere of Death.
Allowing your experiment
within arm's reach of your
Advisor risks the possibility
of immediate destruction.

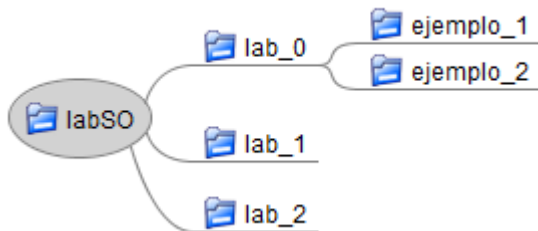


JORGE CHAM © 2007

WWW.PHDCOMICS.COM

MANEJO BÁSICO DE LA TERMINAL

EJERCICIO COMPLETO



24

1. Continuando con el ejercicio ubíquese dentro de la carpeta ejemplo_1.
2. Abra el editor de texto y codifique el programa mostrado en la figura . Modifique el programa de tal manera que se despliegue **Hola mundo**.
3. Guárdelo en el directorio de trabajo como ejemplo1_lab0.c.
4. Compile el programa de manera que el nombre del ejecutable resultante se llame hola.
5. Pruebe el ejecutable.
6. **Con esto ya quedamos listos para empezar a estudiar el lenguaje C.**

ejemplo1.c (~/.Documents/LabSO) - gedit

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

//Comentario de una linea
char nombre[50];

/*
Comentario de
varias lineas
*/

int main() {
    int pid = getpid();
    printf("Ingrese su nombre por favor: ");
    scanf("%s", nombre);
    printf("Hola %s, su id de proceso es %d", nombre, pid);
    return 0;
}
```

C Tab Width: 8 Ln 17, Col 12 INS