SO sesión 7: gestión de entrada/salida (open, read, write…)

domingo, 17 de enero de 2021 19:29

Para leer en el man	Descripción básica	Opc
mknod	Comando que crea un fichero especial	
insmod	Comando que inserta un módulo en el kernel	
rmmod	Comando que descarga un módulo del kernel	
lsmod	Comando que muestra el estado de los módulos cargados en el kernel	
sudo	Comando que permite ejecutar un comando como root	
open	Abre un fichero o dispositivo	
write	Llamada a sistema para escribir en un dispositivo virtual	
read	Llamada a sistema para leer de un dispositivo virtual	
grep	Comando que busca patrones en un fichero o en su entrada estándar si no se le pasa fichero como parámetro	
ps	Comando que muestra información sobre los procesos en ejecución	
strace	Lista las llamadas a sistema ejecutadas por un proceso	

Programa < entrada.txt (redireccionar la entrada)

iones a consultar
c,p
-c
-e, -o
-e, -c

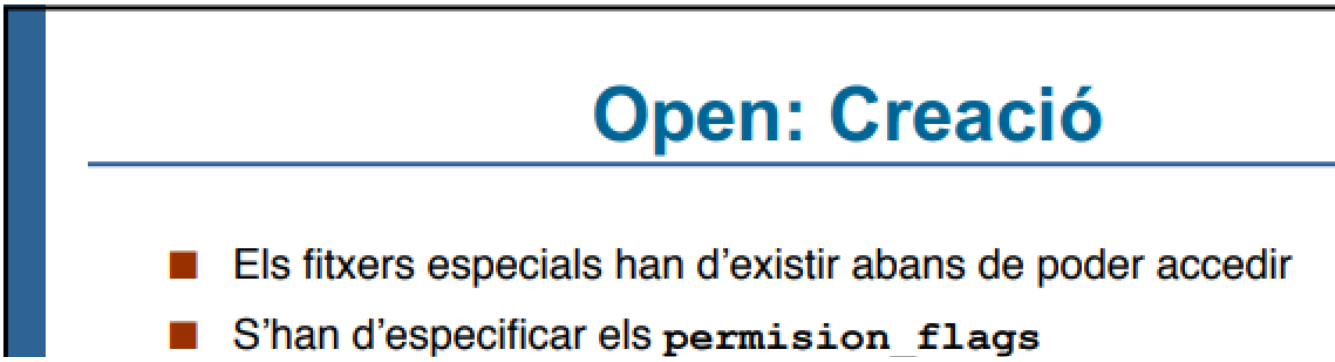
Programa > salida.txt (redireccionar la salida)
Ls -l | grep es (| es una pipe sin nombre, redirecciona la salida
la entrada de grep es)

```
char buffer[64];
int num,i=0;
// Cuando el usuario apriete CtrlD el read devolverá 0
// Como no conocemos cuantas cifras tiene el número, hay que leerlo con un buc
while (read(0,&buffer[i],1)>0) i++;
buffer[i]='\0';
num=atoi(buffer);
```

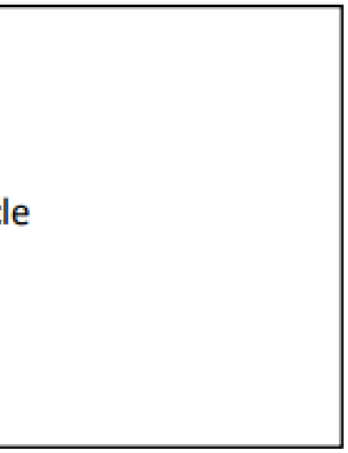
Lee un número de la consola (en ASCII) y luego lo convierte en
entero

```
char buff[64];
int num=10562;
sprintf(buff,"%d",num);
write(1,buff,strlen(buff));
```

Para escribir un numero por la consola, debemos transformarl
cadena de caracteres primero, y luego hacer un write

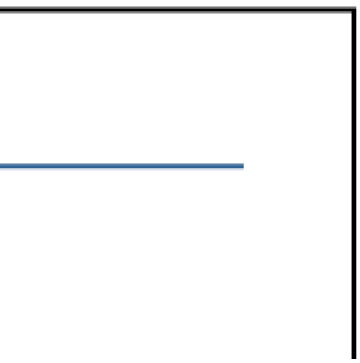


a de ls -l a



n un

lo a una



- És una OR (|) de: S_IRWXU, S_IRUSR, S_IWUSR, etc
- Els fitxers de dades es poden crear a la vegada que fem l'o
  - En aquest cas hem d'afegir el flag O_CREAT (amb una l'acces_mode
- No hi ha una crida a sistema per eliminar dades d'un fitxer només les podem esborrar totes. Si volem truncar en contin afegirem el flag O_TRUNC a l'acces_mode
  - Ex1: open("X",O_RDWRIO_CREAT, S_IRUSRIS el fitxer no existia el crea, si existia no té efecte
  - Ex2: open("X",O_RDWRIO_CREATIO_TRUNC, el fitxer no existia el crea, si existia s'alliberen les es posa el tamany a 0 bytes.

Open

- Per tant, com s'associa un nom a un dispositiu virtual?
- `fd = open(nom, access_mode[,permision_flags]`
  - Vincula nom de fitxer a dispositiu virtual (descriptor de f
    - A més, permet fer un sol cop les comprovacions de cop verificat, ja es pot executar *read/write* múltiples es torna a comprovar
  - Se li passa el **nom** del dispositiu i retorna el dispositiu v *descriptor* o canal) a través del qual es podran realitzar de lectura/escriptura ,etc
  - El **access_mode** indica el tipus d'accés. Sempre ha d'a d'aquests tres com a mínim :

:

pen:

a OR de bits) a

parcialment,
ngut d'un fitxer

S_IWUSR) → Si

S_IRWXU)→ Si
s seves dades y

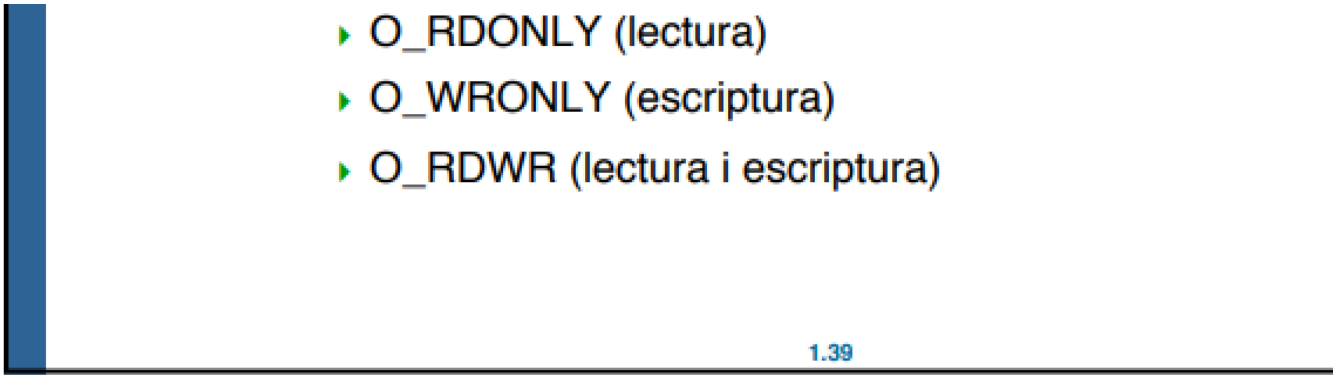
) ;

fitxer o canal)

proteccions. Un
cops però ja no

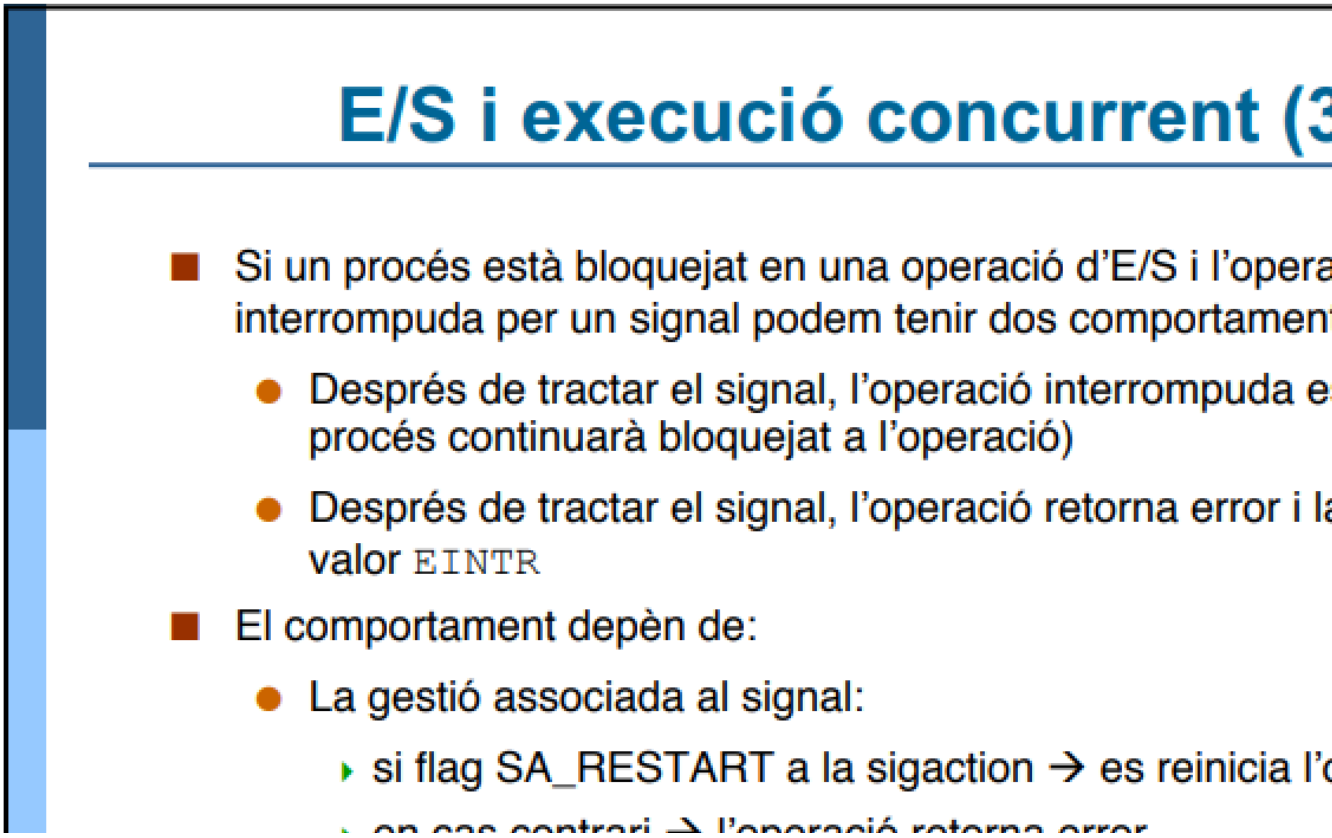
virtual (*fd: file*
les operacions

anar un



El major identifica el tipo de driver y el minor identifica la insta
concreta.
Para redireccionar terminales: ./es1 > /dev/pts/1

While => espera activa, consume CPU
En un read, si no escribes nada, hace una espera activa, no co
CPU
Errno indica error => habitual (errno == EINTR), indica interrur
signal



ancia

nsume

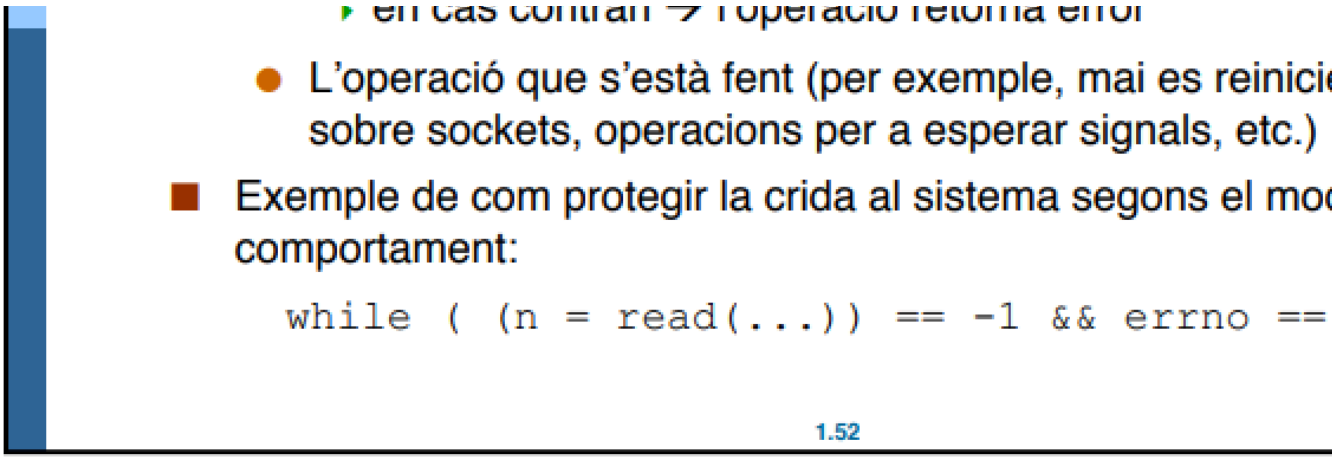
mpido por

3)

ació és
ts diferents:
s reinicia (el

a `errno` pren per

operació



Cat /proc/devices => dispositivos instalados en la maquina
Ls -l /dev => carpeta con los devices
Lsmod => modulos en la maquina

en operacions

del de

```
 EINTR );
```