

ExamenSOM1Resuelto1.pdf



Anónimo



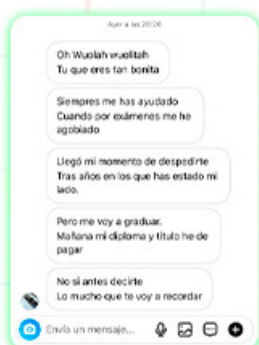
Sistemas Operativos



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada



**Que no te escriban poemas de amor
cuando terminen la carrera**



*(a nosotros por
suerte nos pasa)*

WUOLAH

WUOLAH

Oh Wuolah wuolithah
Tu que eres tan bonita

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

```
[root@localhost ~]# chage -W 6 maria
[root@localhost ~]# chage -W 5 antonio
```

//Comprobamos que han cambiado con /etc/shadow

```
antonio:$6$XsUZJAID$lvvI7hR502K8F0rjzgpkkJwGToKQJiKtAIDw9ndmJHjYa0IvJrLwbrS49p/I2TSVFhZrzB9ygz2rTFhvyFIJ1:19646:0:15:5:::
maria:$6$QdISXSW/$Kpeqh/KsldvrGAPGClSa7znG4oS8KqgYfBatCwErayUn45nQ2E1ni2VA5GYV.TR0wZX2bTH7jLYK5HgEF/zl1:19646:0:10:6:::
```

c)

// /etc/fstab es la lista de los sistemas de archivos montados actualmente

```
[root@localhost ~]# cat /etc/mtab
LABEL=ROOT / auto rw,noatime 0 0
proc /proc proc rw 0 0
sysfs /sys sysfs rw 0 0
devpts /dev/pts devpts rw,gid=5,mode=620 0 0
tmpfs /dev/shm tmpfs rw 0 0
/tmp /tmp tmpfs rw,mode=1777 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
```



/etc/mtab → El archivo `/etc/mtab` es un archivo de texto que contiene información sobre los sistemas de archivos montados en el sistema en un momento dado. Cada línea en este archivo representa un sistema de archivos montado y proporciona detalles sobre el punto de montaje, el dispositivo de almacenamiento subyacente y las opciones de montaje.

Ejercicio 2:

- a) Crea un archivo de 35MB y asócialo al `/dev/loop3`.
- b) Crea una partición con `fdisk` y crea un SA de tipo `ext3`.
- c) Monta el sistema de archivos anterior en `/mnt`, con permisos para solo lectura y no ejecución de binarios.

a)

//Creamos dispositivo virtual `/dev/loop3`

//Creación de archivo de X MB → `dd`

`if=/dev/zero of=/root/nombre_archivo bs=?k count=?` (aquí ya se pone 100, 1000, 10000, según sea bs)

```
[root@localhost ~]# mknod /dev/loop3 b 7 3
[root@localhost ~]# dd if=/dev/zero of=/root/archivoexamen bs=34k count=1000
1000+0 records in
1000+0 records out
34816000 bytes (35 MB) copied, 0.030764 s, 1.1 GB/s
[root@localhost ~]# losetup /dev/loop3 /root/archivoexamen
```

b)

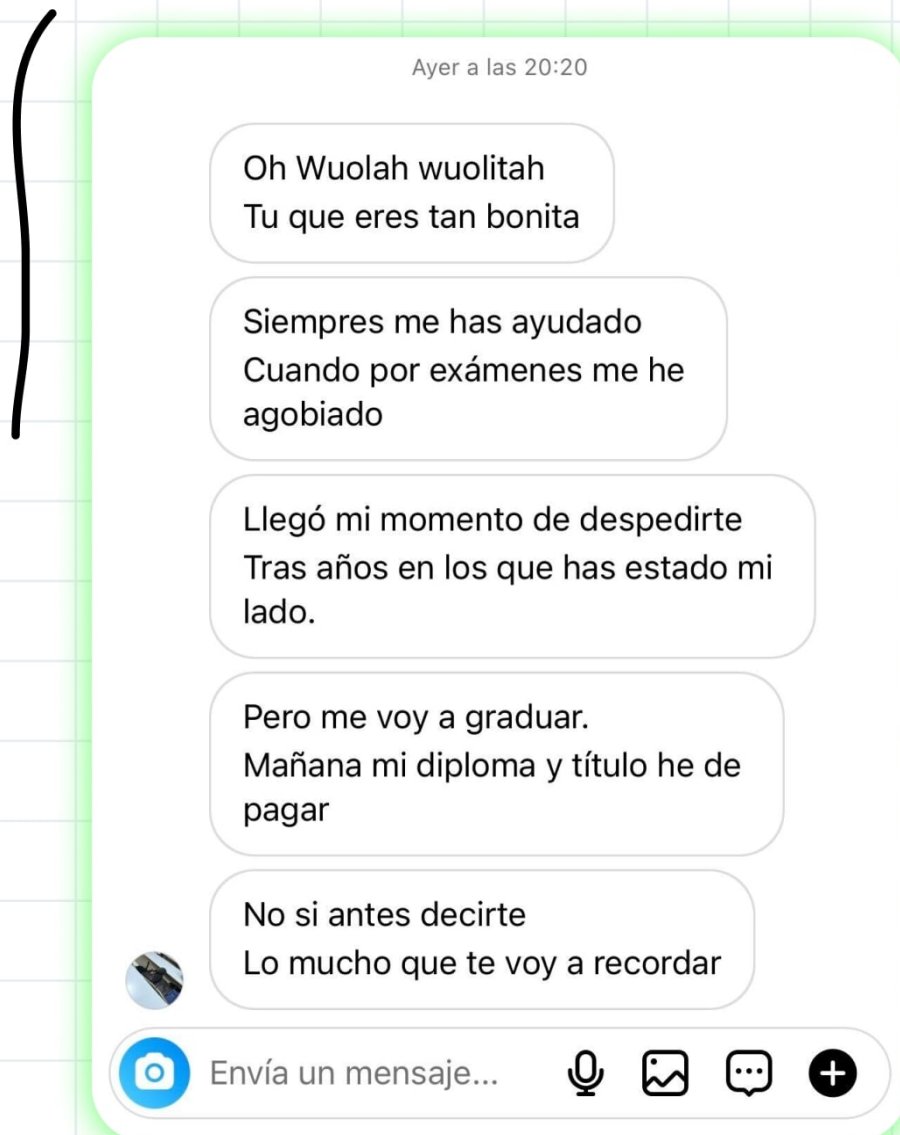
//Creamos una partición

```
[root@localhost ~]# fdisk /dev/loop3
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel
Building a new DOS disklabel with disk identifier 0x0ff90667.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n
Command action
```

**Que no te escriban poemas de amor
cuando terminen la carrera ▶▶▶▶▶▶▶▶**
(a nosotros por suerte nos pasa) 😊



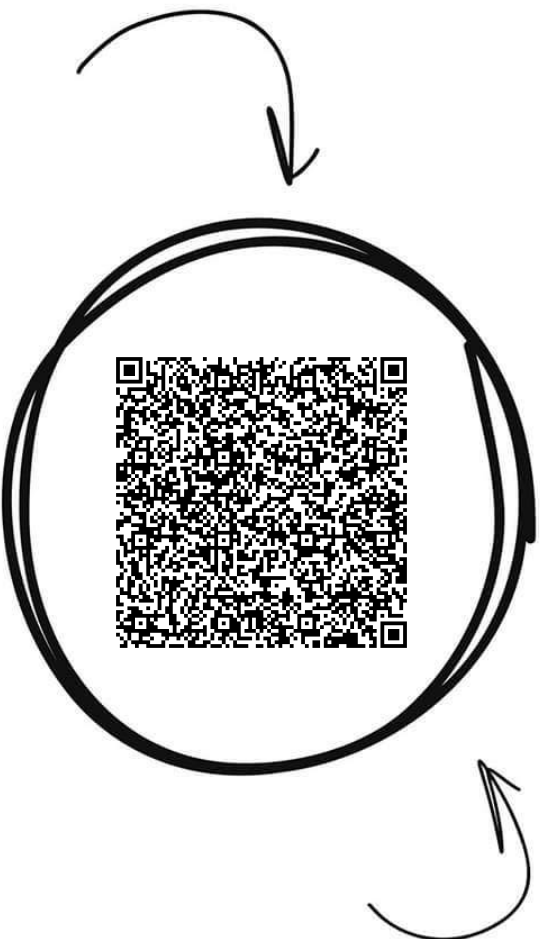
WUOLAH



Sistemas Operativos



Comparte estos flyers en tu clase y consigue más dinero y recompensas



Banco de apuntes de la

WUOLAH

- 1** Imprime esta hoja
- 2** Recorta por la mitad
- 3** Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes

- 4** Llévate dinero por cada descarga de los documentos descargados a través de tu QR



```

e   extended
p   primary partition (1-4)

Command action
e   extended
p   primary partition (1-4)
p
Partition number (1-4, default 1):
Using default value 1
First sector (2048-67999, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-67999, default 67999):
Using default value 67999

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 22: Invalid argument.
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.

```

b)

//Crea una partición con fdisk y crea un SA de tipo ext3.

```

[root@localhost ~]# mke2fs -t ext3 /dev/loop3
mke2fs 1.41.12 (17-May-2010)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
8520 inodes, 34000 blocks
1700 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=34865152
5 block groups
8192 blocks per group, 8192 fragments per group
1704 inodes per group
Superblock backups stored on blocks:
    8193, 24577

Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 32 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.

```

c)

//Monta sistema de archivos

```

[root@localhost ~]# mkdir /mnt
mkdir: cannot create directory '/mnt': File exists
[root@localhost ~]# mount -r -o noexec /dev/loop3 /mnt
[ 4479.160000] EXT3-fs: barriers not enabled
[ 4479.160000] kjournald starting.  Commit interval 5 seconds
[ 4479.160000] EXT3-fs (loop3): mounted filesystem with writeback data mode

```

Ejercicio 3:

- Explica que hace la segunda línea de la orden top. Ordena los procesos mostrados según su PID.
- Genera 10 informes cada 3 segundos con vmstat. Expresa los datos en MiB. Explica cuál es el significado de las columnas r y free.

Que no te escriban poemas de amor
cuando terminen la carrera ▶▶▶▶▶▶▶▶



WUOLAH

(a nosotros por suerte nos pasa)

No si antes decirte
Lo mucho que te voy a recordar

Pero me voy a graduar.
Mañana mi diploma y título he de
pagar

Llegó mi momento de despedirte
Tras años en los que has estado mi
lado.

Siempre me has ayudado
Cuando por exámenes me he
agobiado

Oh Wuolah wuolilah
Tu que eres tan bonita

c) Crea un script que, dado el PID de un proceso, aumente su prioridad en 2. Además, tiene que producir la siguiente salida (para el proceso pasado como argumento)

```
Tasks: 49 total, 1 running, 48 sleeping, 0 stopped, 0 zombie
```

Muestra información sobre las estadísticas de los procesos del sistema (número de procesos, procesos en ejecución, durmiendo, parados o zombies)

Para ordenarlo según su PID hay que pulsar 'N' mientras top está en ejecución y 'R' para que sea en orden creciente

b)

```
[root@localhost ~]# vmstat -S M 3 10
procs -----memory----- --swap-- --io-- --system-- --cpu-----
r  b   swpd   free   buff   cache   si   so   bi   bo   in   cs   us   sy   id   wa   st
1  0     0    917     3    53     0     0     2     5   99    3   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   99    2   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   98    2   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   98    1   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   98    2   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   98    1   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   99    3   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   99    1   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   99    2   0   0  100   0   0
0  0     0    917     3    53     0     0     0     0   99    1   0   0  100   0   0
```

Columna **r** → muestra procesos que están actualmente en la cola de ejecución.

Columna **free** → indica que la memoria libre se puede estar agotando.

c)

```
[root@localhost ~]# cat aumentar_prioridad.sh
#!/bin/bash

#Verificamos que se proporcione PID como argumento
PID=$1
//Aumentamos la prioridad con renice
renice +2 -p $PID
PID=$(ps -o pid= -p $PID)
PPID=$(ps -o ppid= -p $PID)
PRI=$(ps -o pri= -p $PID)
NICE=$(ps -o nice= -p $PID)
USUARIO=$(ps -o user= -p $PID)
memoria_usada_k=$(ps -o rss= -p $PID)
TERMINAL=$(ps -o tty= -p $PID)
COMANDO=$(ps -o comm= -p $PID)

echo "PID PPID PRI NICE USUARIO memoria_usada_k TERMINAL COMANDO"
echo "$PID $PPID $PRI $NICE $USUARIO $memoria_usada_k $TERMINAL $COMANDO"
```



EJEMPLO: PID=\$(ps -o pid= -p \$PID) , -o es para definir formato dónde pid = -p \$PID, -p es para seleccionar el proceso.

Ejercicio 4:

a) Crea un script que encuentre todos los archivos especiales de dispositivo de caracteres del directorio /dev y los guarde en el archivo /root/especiales_CAR.

b) Programe el script para que se ejecute mañana a las 17:30.

c) Programe el script para que se ejecute periódicamente todos los sábados y domingos cada 15 minutos.

```
#!/bin/bash
// Buscar todos los archivos y guardarlos en archivo /root/especiales_CAR
find /dev -type c > /root/especiales_CAR
```

b y c)

```
[root@localhost ~]# crontab -e
30 17 * * * /root/find_especiales.sh
0,30 * * * 6,7 /root/find_especiales.sh
```