



Universidad de Murcia
Facultad de Informática

TÍTULO DE GRADO EN
INGENIERÍA INFORMÁTICA

Fundamentos de Computadores

Tema 4: Introducción a los Sistemas Operativos

Boletín de autoevaluación de las sesiones prácticas

CURSO 2020 / 21

Departamento de Ingeniería y Tecnología de Computadores

Área de Arquitectura y Tecnología de Computadores



Índice general

I. Cuestiones de autoevaluación	2
A4.1. Objetivos	2
A4.2. Cuestiones	3
II. Soluciones a las cuestiones de autoevaluación	9
S4.1. Soluciones	9

Cuestiones de autoevaluación

A4.1. Objetivos

El objetivo de este boletín es comprobar el grado de aprovechamiento por parte del alumno de las sesiones prácticas del tema 4 de la asignatura. Todas las preguntas planteadas en este boletín deberían poder resolverse con los conceptos aprendidos durante dichas sesiones.

A lo largo de este boletín de autoevaluación te encontrarás con fundamentalmente cuatro tipos de cuestiones diferentes:

1. Preguntas de tipo *test*: se ofrecen varias opciones, de las cuales tendrás que elegir cuál es la única correcta.
2. Preguntas de *emparejamiento*: se dan dos columnas con el mismo número de opciones en cada una, y hay que emparejar todos y cada uno de los términos de la columna primera con sus respectivos términos correspondientes en la segunda.
3. Preguntas de *rellenar huecos*: se enuncia un texto en el que existen algunos huecos, que tendrás que rellenar con el término correcto.
4. Preguntas de *razonamiento* breve: se realiza una pregunta a la que hay que contestar de modo preciso y conciso (menos de un párrafo).

Se trata de que, individualmente, en horario de trabajo en casa (fuera de las sesiones de teoría/prácticas semanales), y tras la finalización de las sesiones de prácticas dedicadas al tema 4, cada alumno intente resolver por su cuenta las preguntas del boletín, por supuesto SIN mirar previamente las soluciones, que se encuentran en el último apartado del documento. Una vez realizados los ejercicios, el propio alumno comprobará la corrección de los resultados, mirando las soluciones disponibles en dicho último apartado. Este proceso debería ayudarle a detectar sus propias carencias, en las que debe insistir en el trabajo de estudio en casa, repetición de los ejercicios propuestos en el boletín, planteamiento de dudas al profesor, etc.

Nota: Una cierta cantidad de las las preguntas planteadas en estos boletines han aparecido, planteadas de una u otra forma, en exámenes pasados de esta asignatura.

A4.2. Cuestiones

1. **[Test]** Indica cuál de los siguientes comandos se emplea para terminar una sesión de la línea de comandos en Linux:
 - a) end
 - b) quit
 - c) exit
 - d) out
2. **[Razonar]** ¿Cómo se puede copiar y pegar un determinado texto mostrado en el terminal en nuestra línea de comandos, usando el ratón? (dar una solución que sea también válida para una consola de texto, fuera del entorno gráfico).
3. **[Rellenar]** Para cambiarnos a la tercera consola de texto virtual debemos pulsar la combinación de teclas _____.
4. **[Rellenar]** Para volver al entorno gráfico cuando nos encontramos en una consola de texto debemos pulsar la combinación de teclas _____.
5. **[Emparejar]** Un terminal de Linux nos muestra el siguiente prompt: `art@funes:~/$. Empareja los siguientes términos adecuadamente:`

login	no disponible
directorío actual	art
nombre de la máquina	funes
palabra de paso	/home/art
directorío raíz	/

6. **[Test]** Los comandos `who` y `w` de Linux se emplean para:
 - a) Listar los usuarios que están conectados actualmente en el sistema.
 - b) Listar los usuarios que pueden acceder al sistema.
 - c) Determinar a qué usuarios les vamos a permitir acceder al sistema.
 - d) Lista los datos completos (nombre, apellidos, teléfono, etc.) del usuario asociado a un login determinado.
7. **[Test]** El comando `cd`, cuando se usa sin ningún argumento, nos lleva:
 - a) Al penúltimo directorio en el que estuvimos situados (justamente antes del actual).
 - b) Al nuestro directorio de inicio (nuestro *home*).
 - c) Al directorio padre del que nos encontramos.
 - d) Al directorio raíz.
8. **[Emparejar]** Emparejar los siguientes comandos con la acción correspondiente que realiza cada uno:

<code>cd -</code>	Nos lleva al directorio raíz.
<code>cd /</code>	Nos lleva al directorio en que justo anteriormente estábamos situados.
<code>cd .</code>	Nos lleva a nuestro directorio de inicio (nuestro <i>home</i>).
<code>cd</code>	Nos lleva al directorio padre del que estamos situados.
<code>cd ..</code>	No tiene ningún efecto.

9. **[Rellenar]** El comando que nos muestra la ruta completa desde la raíz hasta el directorio actual es _____.
10. **[Test]** El comando que nos muestra la fecha y la hora en el terminal es:
- a) time
 - b) date
 - c) datetime
 - d) timedate
11. **[Razonar]** Si tecleamos la cadena “fir” al comienzo de una línea de comandos, y pulsamos el tabulador, ¿qué ocurrirá?
12. **[Razonar]** Considera un directorio llamado `exámenes` que pertenece al usuario `juan` y al grupo `profesores`, y tiene los permisos `'rwx rwx ---'`. Dentro de ese directorio existen únicamente dos ficheros, llamados `examen1.txt` y `examen2.txt`, también pertenecientes al usuario `juan` y al grupo `profesores`, y con permisos `'rw- rw- r--'` y `'r-x r-x r-x'` respectivamente. Supongamos también que existen otros usuarios llamados `pedro` y `jaime` que también pertenecen al grupo `profesores`. Determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “El usuario `juan` puede modificar `examen2.txt`”.
13. **[Razonar]** En el mismo supuesto de permisos anterior, determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “Los usuarios `pedro` y `jaime` pueden borrar `examen2.txt`”.
14. **[Razonar]** En el mismo supuesto de permisos anterior, determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “Los usuarios `pedro` y `jaime` pueden modificar `examen2.txt`”.
15. **[Razonar]** En el mismo supuesto de permisos anterior, determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “Cualquier usuario del sistema puede leer `examen1.txt`”.
16. **[Razonar]** En el mismo supuesto de permisos anterior, determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “Los usuarios `pedro` y `jaime` pueden crear y borrar nuevos ficheros en ese directorio de `juan`”.
17. **[Razonar]** En el mismo supuesto de permisos anterior, determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “Cualquier usuario puede ver el contenido del directorio `exámenes`”.
18. **[Razonar]** En el mismo supuesto de permisos anterior, determina si la siguiente afirmación es verdadera o falsa, justificando brevemente la respuesta: “Cualquier usuario convertido en superusuario podría borrar el directorio `exámenes`”.
19. **[Rellenar]** En el mismo supuesto de permisos anterior, supón que queremos establecer los permisos adecuados para que los ficheros `examen1.txt` y `examen2.txt` solamente puedan ser borrados por `juan`, leídos y modificados por él mismo y los miembros de su grupo, y sólo leídos por el resto de usuarios. Los comandos a utilizar, ejecutados desde dentro del directorio, serían (rellenar con sendos números): `chmod _____ ../exámenes` (en relación a los permisos del directorio) y `chmod _____ examen1.txt` y `chmod _____ examen2.txt` (en relación a los permisos de los archivos).
20. **[Rellenar]** El comando que sirve para mostrarnos el historial de nuestros últimos comandos tecleados es _____, y el nombre del fichero donde se guarda dicho historial entre sesiones es _____, que se encuentra en nuestro directorio `home`.
21. **[Rellenar]** El principal comando utilizado en linux para encontrar ayuda sobre el uso de cualquier comando es _____, al que se le pasa como parámetro el nombre del comando sobre el que se necesita la ayuda.

22. **[Rellenar]** El comando exacto a teclear para movernos al directorio `/home/mariano` cuando estamos situados en el directorio `/home/pelayo`, usando una ruta relativa lo más corta posible, sería _____.
23. **[Rellenar]** El comando completo que habría que teclear para listar los contenidos del directorio `bin` (que cuelga directamente de la raíz), incluyendo nombres de archivos y subdirectorios, permisos, dueños, tamaño, etc., almacenando la salida en un fichero `prueba.txt` en nuestro directorio `/home/alumno`, suponiendo que estamos situados en el mismo, sería _____ (especificar el comando de la manera más concisa posible).
24. **[Emparejar]** Emparejar las siguientes rutas absolutas de directorios con el uso habitual que se le da a cada uno en los sistemas UNIX/Linux:

<code>/usr/bin</code> y <code>/bin</code>	Directorios de trabajo de los usuarios
<code>/etc</code>	Ficheros temporales
<code>/sbin</code>	Librerías del sistema
<code>/usr/lib</code> y <code>/lib</code>	Ficheros de configuración
<code>/tmp</code>	Aplicaciones superusuario
<code>/home</code>	Aplicaciones normales del sistema

25. **[Test]** Sea un fichero llamado `ejemplo.bin` situado en el directorio de inicio de un usuario llamado `andres`. Determinar cuál de las siguientes afirmaciones es cierta:
- a) Una ruta absoluta hasta dicho fichero sería `./andres/ejemplo.bin`
 - b) Una ruta relativa desde el directorio de inicio de otro usuario del sistema sería `../andres/ejemplo.bin`
 - c) Una ruta relativa hasta el mismo sería `/home/andres/ejemplo.bin`
 - d) Una ruta absoluta hasta dicho directorio sería `/home/ejemplo.bin`
26. **[Test]** Si un usuario llamado `antonio` ejecuta el comando `ls -l ~/fichero.txt`, ¿cuál de las siguientes tareas estará intentando realizar?:
- a) Obtener un listado largo con la información del fichero llamado `fichero.txt`, que está situado en su directorio `/home/antonio`.
 - b) Obtener un listado largo con la información del fichero llamado `fichero.txt`, que cuelga directamente de la raíz.
 - c) Obtener un listado largo con la información del fichero llamado `fichero.txt`, que se encuentra directamente en su directorio actual.
 - d) Ninguna de las anteriores opciones es cierta.
27. **[Test]** Si un usuario llamado `antonio` ejecuta el comando `ls -l ./fichero.txt`, ¿cuál de las siguientes tareas estará intentando realizar?:
- a) Obtener un listado largo con la información del fichero llamado `fichero.txt`, que está situado en su directorio `/home/antonio`.
 - b) Obtener un listado largo con la información del fichero llamado `fichero.txt`, que cuelga directamente de la raíz.
 - c) Obtener un listado largo con la información del fichero llamado `fichero.txt`, que se encuentra directamente en su directorio actual.
 - d) Ninguna de las anteriores opciones es cierta.
28. **[Razonar]** Indicar toda la información que se puede determinar a partir del siguiente volcado de una sesión interactiva de terminal:

```
pedroe@juanelo:~/fc-grad/docs/tema4/exanet$ ls -l
-rw-r----- 1 pedroe profe 15558 sep 17 13:33 preguntas-boletin-3.txt
```

29. **[Razonar]** En Linux, los archivos y directorios que empiezan por el carácter _____ están ocultos (esto es, no se muestran en un listado ordinario con el comando `ls`).
30. **[Rellenar]** Supongamos que queremos listar todos los archivos y directorios que se encuentran de nuestro directorio `/home/alumno`, incluídos aquellos que están ocultos. El comando que haría esta tarea sería _____.
31. **[Emparejar]** Emparejar los siguientes comandos de Linux con la acción correspondiente:

<code>ls -R /home/blas</code>	Muestra un listado largo de los contenidos de <code>/home/blas</code> .
<code>ls -l /home/blas</code>	Muestra un listado recursivo de todo lo que cuelga de <code>/home/blas</code> .
<code>ls -a /home/blas</code>	Muestra un listado largo del propio directorio <code>/home/blas</code> .
<code>ls -dl /home/blas</code>	Muestra el listado de contenidos de <code>/home/blas</code> , incluyendo ocultos.

32. **[Razonar]** Indicar qué hace exactamente el siguiente comando:

```
pedroe@juanelo:~$ find /usr/lib -type f -iname "*z"
```

33. **[Razonar]** Explica brevemente la diferencia, y qué ocurriría en cada caso, entre ejecutar los comandos

```
$ gcalc & kcalc &
```

y ejecutar los comandos

```
$ gcalc ; kcalc
```

34. **[Razonar]** Dado el primer caso anterior (`gcalc & kcalc &`): si a continuación introducimos en el terminal el comando `fg` ¿qué ocurriría?
35. **[Razonar]** ¿Y si seguidamente pulsamos `Ctrl-z`?
36. **[Rellenar]** Supongamos que lanzamos el comando `kcalc` en segundo plano, y `bash` nos contesta así:

```
pedroe@juanelo:~$ kcalc &
[1] 9177
```

El comando que habría que usar a continuación para matar dicho proceso `kcalc` sería _____.

37. **[Razonar]** Explica con una frase qué hace exactamente el siguiente comando de Linux:

```
pedroe@juanelo:~$ find /usr -type f -size +120k -iname "[e-m]*.???"
```

38. **[Test]** Considérese el directorio `/home/pelayo/fotos`. Queremos establecer los permisos necesarios para que a dicha carpeta de fotos de la cuenta de pelayo todos los miembros del grupo al que pertenece pelayo puedan acceder y contemplar qué archivos contiene, pero sin poder añadir ni borrar ninguno, y que el resto de usuarios no tengan acceso ni visibilidad ninguna. El propio pelayo deberá seguir teniendo todos los permisos (posibilidad de acceder, contemplar y modificar los contenidos de dicha carpeta). El comando adecuado para realizar dicha tarea sería:

- a) `chmod 750 /home/pelayo/fotos`
- b) `chmod 741 /home/pelayo/fotos`
- c) `chgrp 730 /home/pelayo/fotos`
- d) `chown 540 /home/pelayo/fotos`

39. **[Rellenar]** Suponer que tenemos un directorio `/home/alumno/directorio` sobre el que ejecutamos el siguiente comando linux desde un terminal:

```
alumno@juanelo:~$ chmod 754 /home/alumno/directorio
```

Si a continuación ejecutamos un comando `ls -ld /home/alumno/directorio`, el resultado de dicho comando sería (rellenar la cadena que se mostraría exactamente en la línea continua):

```
alumno@juanelo:~$ ls -ld /home/alumno/directorio
_____ 2 alumno alumno 4096 2008-05-16 10:16 /home/alumno/directorio
```

40. **[Rellenar]** Suponer que estamos situados en un directorio en el que tenemos permisos de escritura, y del que cuelga un subdirectorio llamado `seguridad`, del que a su vez cuelgan varios ficheros y subdirectorios adicionales. Si queremos crear un fichero comprimido con todos esos contenidos, llamado `seguridad.tgz`, el comando que tendremos que teclear sería _____.

41. **[Emparejar]** Empareja las siguientes opciones de la orden `tar` con su correspondiente cometido:

<code>czvf</code>	Lista los contenidos de un archivo comprimido (sin descomprimirlo realmente).
<code>xzvf</code>	Crea un archivo comprimido.
<code>tzvf</code>	Descomprime un archivo comprimido.

42. **[Razonar]** Indicar el cometido exacto del siguiente comando:

```
cp -r /home/mariano /tmp/mariano
```

43. **[Razonar]** ¿Qué ocurriría entonces si a continuación ejecutáramos el siguiente comando?

```
rm -rf /tmp/mariano
```

44. **[Rellenar]** Una expresión con comodines que englobaría a los nombres de fichero `aaa.txt`, `acd.tar`, `aef.tt` y `agh.t`, sin englobar a los ficheros `aaaa.txt`, ni `abc.d` y (que podría usarse, por ejemplo, en combinación con un comando `ls -l` para hacer un listado largo de dichos ficheros) sería _____.

45. **[Emparejar]** Empareja los siguientes comandos Linux con su correspondiente cometido:

<code>cat</code>	Matar un proceso.
<code>less</code>	Mostrar un fichero de texto por el terminal (pudiendo paginarlo).
<code>hexdump</code>	Mostrar un fichero de texto por el terminal (todo seguido).
<code>ps</code>	Mostrar el contenido en hexadecimal de un fichero por el terminal.
<code>touch</code>	Mostrar los procesos en ejecución.
<code>df</code>	Crea un archivo nuevo, vacío (o bien modifica su fecha si ya existía).
<code>mv</code>	Mover un archivo/directorio de sitio, o renombrarlo.
<code>du</code>	Chequear espacio libre en discos.
<code>kill</code>	Chequear el espacio ocupado por un(os) determinado(s) archivo(s)/directorio(s).

46. **[Rellenar]** Supongamos que teníamos un subdirectorio llamado `Musica`, con tres subdirectorios, cada uno de ellos a su vez con un número indeterminado de archivos `.mp3`. Nos encontrábamos en ese momento en el directorio `/home/alumno/`, y queríamos saber el espacio en disco ocupado por cada uno de dichos tres subdirectorios, con un sólo comando. El comando ejecutado para obtener la salida indicada a continuación fue _____:

```
27M      /home/pedroe/Musica/Rollings
22M      /home/pedroe/Musica/Amaral
78M      /home/pedroe/Musica/Beatles
```

Considérese la siguiente sesión con la línea de comandos de Linux:

```
mariano@qosmio:~/Documentos$ ls -l
drwxr-xr-x 2 mariano profe 4096   sep 15 18:59 kig
dr--r--r-- 1 mariano profe 4096   sep  5 23:56 dir
-r--r--r-- 1 mariano profe 155509 sep 23 19:46 print.pdf
mariano@qosmio:~/Documentos$ du -hs kig
48K kig
```

A la vista de la sesión interactiva anterior, contestar a las siguientes preguntas:

47. **[Rellenar]** El directorio `/home/mariano/Documentos` contiene _____ subdirectorios (indicar el número).
48. **[Rellenar]** El comando completo que teclearías a continuación para crear un fichero comprimido llamado `kig.tgz` con los contenidos de todo lo que cuelga (recursivamente) del directorio `kig` sería _____.
49. **[Razonar]** Dicho archivo `kig.tgz` en ningún caso ocupará más de _____ Kilobytes. El archivo `print.pdf`, por su parte, ocupa aproximadamente _____ Kilobytes.
50. **[Razonar]** ¿Qué crees que ocurriría si se ejecutase a continuación el comando `cp kig.tgz dir`?
51. **[Rellenar]** El comando completo que teclearías para que el archivo `print.pdf` pudiese ser leído y escrito por mariano, sólo leído por sus compañeros del grupo profe, y ni leído ni escrito por el resto de usuarios, sería _____.
52. **[Razonar]** Indica si es cierta o falsa la siguiente afirmación, y explica brevemente por qué: “El archivo `print.pdf` fue creado, necesariamente, el 23 de octubre de 2009, a las 19:46”.
53. **[Rellenar]** El comando completo que teclearías para generar un listado de todos los procesos que se están ejecutando en el sistema, pero escribiendo dicho listado justo al final del fichero `/home/alumno/informe.txt` (que ya existía previamente, y tenía un determinado contenido que no queremos machacar) sería _____.
54. **[Razonar]** Indicar exactamente lo que hace el comando siguiente:
- ```
find /tmp -type d -printf "El directorio %p es propiedad del usuario %u\n"
```
55. **[Rellenar]** El comando utilizado para crear un subdirectorio es \_\_\_\_\_, y para borrarlo, el comando \_\_\_\_\_.

## Soluciones a las cuestiones de autoevaluación

### S4.1. Soluciones

1. *Solución:* c)
2. *Solución:* Se marca el texto a copiar de principio a fin pulsando el botón izquierdo del ratón, y después, con el cursor situado en el lugar en que se quiere pegar el texto, se pulsa el botón central.
3. *Solución:* Ctrl-Alt-F3.
4. *Solución:* Ctrl-Alt-F7.
5. *Solución:*

|                      |               |
|----------------------|---------------|
| login                | art           |
| directorio actual    | /home/art     |
| nombre de la máquina | funes         |
| palabra de paso      | no disponible |
| directorio raíz      | /             |

6. *Solución:* a)
7. *Solución:* b)
8. *Solución:*

|       |                                                                        |
|-------|------------------------------------------------------------------------|
| cd /  | Nos lleva al directorio raíz.                                          |
| cd -  | Nos lleva al directorio en que justo anteriormente estábamos situados. |
| cd    | Nos lleva a nuestro directorio de inicio (nuestro <i>home</i> ).       |
| cd .. | Nos lleva al directorio padre del que estamos situados.                |
| cd .  | No tiene ningún efecto.                                                |

9. *Solución:* pwd
10. *Solución:* b)
11. *Solución:* El sistema busca todos los comandos instalados (situados en el PATH) que comiencen por dicha cadena (p.e., `firefox`) y, si existe sólo uno, lo completa (ahorrándonos tener que teclear el resto de la cadena), y si existen varios, nos mostrará las distintas opciones, completando siempre hasta la subcadena en que coinciden (p.e., si en nuestro sistema estuvieran los comandos `firefox`, `firefox-3.0`, y `firefox-3.5`, entonces completaría hasta la cadena "firefox", mostrando entonces las tres opciones disponibles).
12. *Solución:* Falsa, ya que el archivo `examen2.txt` no tiene permiso w en su posición 2/9 en los permisos.
13. *Solución:* Cierta, ya que el directorio `exámenes` tiene permiso w en su posición 5/9 en los permisos.
14. *Solución:* Falsa, ya que el archivo `examen2.txt` no tiene permiso w en su posición 5/9 en los permisos.

15. *Solución:* Falsa, ya que aunque el archivo `examen1.txt` tiene permiso `r` en su posición 7/9 en los permisos, el directorio en el que está contenido no tiene ningún permiso para usuarios que no pertenezcan al grupo profesores.
16. *Solución:* Cierta, ya que el directorio `exámenes` tiene permiso `w` en su posición 5/9 en los permisos.
17. *Solución:* Falsa, ya que el directorio `exámenes` no tiene permiso `r` en su posición 7/9 en los permisos.
18. *Solución:* Cierta, ya que el superusuario siempre tiene todos los permisos.
19. *Solución:* 755 664
20. *Solución:* `history .bash_history`
21. *Solución:* `man`
22. *Solución:* `cd ../mariano`
23. *Solución:* `ls -l /bin > prueba.txt`
24. *Solución:*

|                              |                                        |
|------------------------------|----------------------------------------|
| <code>/usr/bin y /bin</code> | Aplicaciones normales del sistema      |
| <code>/etc</code>            | Ficheros de configuración              |
| <code>/sbin</code>           | Aplicaciones superusuario              |
| <code>/usr/lib y /lib</code> | Librerías del sistema                  |
| <code>/tmp</code>            | Ficheros temporales                    |
| <code>/home</code>           | Directorios de trabajo de los usuarios |

25. *Solución:* b)
26. *Solución:* a)
27. *Solución:* c)
28. *Solución:* Un usuario llamado `pedro`, ejecuta un comando para listar los contenidos de un directorio cuya ruta es `/home/pedro/fc-grad/docs/tema4/exanet/` en el que se encuentra, dentro de la máquina `juanelo`. En el mismo sólo se halla un fichero llamado `preguntas-boletin-3.txt`, de 15558 bytes de longitud, modificado por última vez a las 13:33 del 17 de septiembre, perteneciente a ese mismo usuario y al grupo `profe`. Dicho archivo tiene permisos de lectura y escritura para el dueño (`pedro`), de sólo lectura para el resto de miembros del grupo `profe`, y ningún permiso (ni lectura, ni escritura, ni ejecución) para el resto de usuarios del sistema.
29. *Solución:* `.` (carácter punto)
30. *Solución:* `ls -a /home/alumno`
31. *Solución:*

|                                |                                                                                   |
|--------------------------------|-----------------------------------------------------------------------------------|
| <code>ls -l /home/blas</code>  | Muestra un listado largo de los contenidos de <code>/home/blas</code> .           |
| <code>ls -R /home/blas</code>  | Muestra un listado recursivo de todo lo que cuelga de <code>/home/blas</code> .   |
| <code>ls -dl /home/blas</code> | Muestra un listado largo del propio directorio <code>/home/blas</code> .          |
| <code>ls -a /home/blas</code>  | Muestra el listado de contenidos de <code>/home/blas</code> , incluyendo ocultos. |

32. *Solución:* Encontrar todos los ficheros regulares (e.d., no directorios) que cuelgan del subdirectorio `lib` (ubicado dentro del subdirectorio `usr`, éste último ya colgando directamente de la raíz), y cuyo nombre acaba con un carácter “z” o una “Z”.
33. *Solución:* La primera expresión lanzaría las dos aplicaciones en segundo plano, dejando el terminal libre. La segunda ejecutaría `gcalctool` en primer plano y cuando ésta se cerrara se ejecutaría `kcalc`, también en primer plano, quedando todo el tiempo el terminal ocupado hasta el cierre de `kcalc`.
34. *Solución:* La aplicación `kcalc` pasaría a primer plano y el terminal quedaría ocupado.
35. *Solución:* `kcalc` se pararía (quedaría dormido) y el terminal quedaría libre.
36. *Solución:* `kill 9177` (o bien `kill -9 9177`).
37. *Solución:* Busca y muestra los ficheros (no directorios) que cuelguen a cualquier profundidad del directorio `/usr`, que tengan más de 120 KBytes, y cuyo nombre empiece alfabéticamente entre la “e” y “m” (mayúscula o minúscula) y con una extensión (tras el carácter “.”) de exactamente tres caracteres.
38. *Solución:* a)
39. *Solución:* `drwxr-xr--`
40. *Solución:* `tar czvf mariano.tgz mariano`
41. *Solución:*
- |                   |                                                                               |
|-------------------|-------------------------------------------------------------------------------|
| <code>czvf</code> | Crea un archivo comprimido.                                                   |
| <code>xzvf</code> | Descomprime un archivo comprimido.                                            |
| <code>tzvf</code> | Lista los contenidos de un archivo comprimido (sin descomprimirlo realmente). |
42. *Solución:* Copia todos los contenidos que cuelgan del directorio `/home/mariano` (ficheros y subdirectorios) recursivamente, al directorio `/tmp/mariano`
43. *Solución:* Que se borraría toda la copia generada (es decir, el directorio `/tmp/mariano` y todos sus subdirectorios).
44. *Solución:* `a???.t*` (aunque puede haber alguna otra expresión que realice también el trabajo, ésta es la más corta posible).
45. *Solución:*
- |                      |                                                                                 |
|----------------------|---------------------------------------------------------------------------------|
| <code>cat</code>     | Mostrar un fichero de texto por el terminal (todo seguido).                     |
| <code>less</code>    | Mostrar un fichero de texto por el terminal (pudiendo paginarlo).               |
| <code>hexdump</code> | Mostrar el contenido en hexadecimal de un fichero por el terminal.              |
| <code>ps</code>      | Mostrar los procesos en ejecución.                                              |
| <code>kill</code>    | Matar un proceso.                                                               |
| <code>mv</code>      | Mover un archivo/directorio de sitio, o renombrarlo.                            |
| <code>df</code>      | Chequear espacio libre en discos.                                               |
| <code>du</code>      | Chequear el espacio ocupado por un(os) determinado(s) archivo(s)/directorio(s). |
| <code>touch</code>   | Crea un archivo nuevo, vacío (o bien modifica su fecha si ya existía).          |
46. *Solución:* `du -hs Musica/*`
47. *Solución:* `dos`

48. *Solución:* `tar czvf kig.tgz kig`
49. *Solución:* 48 KB el archivo `kig.tgz` (puesto que los contenidos del directorio `kig` ocupaban 48 KB en total), y 155 KB el archivo `print.pdf` (puesto que el listado largo indica que su tamaño es de 155509 bytes, es decir, 155 KB aprox).
50. *Solución:* Se produciría un error de permisos, ya que no tenemos permiso de escritura en el directorio `dir` (es decir, no podemos crear nuevos ficheros dentro de él).
51. *Solución:* `chmod 640 print.pdf`
52. *Solución:* Falso, ya que la fecha y hora que muestra por defecto el listado largo es la de última modificación, que no coincide necesariamente con la de creación.
53. *Solución:* `ps -A >> /home/alumno/informe.txt`
54. *Solución:* Busca todos los subdirectorios que cuelguen del directorio `/tmp`, y para todos ellos indica quién es el usuario propietario, con el mensaje apropiado.
55. *Solución:* `mkdir, rmdir` (aunque en este último caso también puede usarse `rm -rf`).