



# Introducción a los Sistemas Operativos

## Examen de prácticas

28 de octubre de 2016

Grupo y profesor:	DNI:
Nombre:	Apellidos:

### Instrucciones (MUY IMPORTANTE):

- Tiene 2 horas para hacer el examen. Si decide no hacerlo o termina antes de tiempo, no abandonará el laboratorio hasta que se le indique.
- No puede sacar el examen del laboratorio. Tendrá que entregarlo junto con su nombre, DNI y grupo al profesor al finalizar el mismo.
- Los ordenadores no tienen acceso a Internet. Sólo podrá consultar las páginas de manual y cualquier otra fuente de información que haya disponible en su ordenador. Los PDFs con los boletines se han dejado en su directorio personal (/home/examen1 o /home/examen2) para su consulta.
- Tiene que entregar un fichero de texto por cada apartado de cada ejercicio con la solución del mismo. El nombre que debe dar a este fichero aparece al final del apartado correspondiente.
- Cuando lo indique el profesor, deberá subir sus ficheros con sus soluciones al Aula Virtual, a través de una tarea creada a tal efecto. Adjunte los ficheros de forma individual, sin crear un archivo comprimido con todos ellos.
- Si desea hacer alguna observación, añádala como comentario dentro de los ficheros que suba.

### Ejercicios:

1. (2 puntos) Resuelva cada uno de los siguientes apartados mediante una única línea de órdenes conectadas con tuberías, sin bucles ni condicionales de ningún tipo:

- a) (1,25 puntos) Obtenga una lista de todos los ficheros de los directorios /etc y /usr/bin (estos directorios se tienen que recorrer recursivamente) y muestre, para cada fichero, su tamaño en bytes, su nombre y el usuario propietario del mismo. Esta lista debe aparecer ordenada en primer lugar por usuario y en segundo lugar por tamaño. Cualquier mensaje de error que pueda aparecer por no poder acceder a ciertos fichero y/o directorios debe suprimirse. Para facilitar la resolución, puede suponer que los nombres de fichero no contienen espacios. La salida de la orden tiene que parecerse a la siguiente:

```
[alumno@localhost ~]$ línea de órdenes con tuberías
136 inadyn-mt.conf inadyn
93 fetchmailrc.example mail
4096 rules.d polkitd
35876608 skype root
29182288 docker root
15440888 inkscape root
... ..
```

Fichero de solución: **tuberias.a.sh**.

- b) (0,75 puntos) Continúe la orden anterior para obtener el nombre y el tamaño del fichero más grande para cada usuario. La salida de la orden tiene que tabularse para que el formato sea como el mostrado a continuación (los campos son los mismos que los mostrados en el apartado anterior):

```
[alumno@localhost ~]$ línea de órdenes con tuberías
136      inadyn-mt.conf      inadyn
93      fetchmailrc.example  mail
4096     rules.d             polkitd
35876608 skype              root
7046     tcscd.conf          tss
```

Pista: dependiendo de su solución, es posible que necesite usar la opción -f de la orden uniq.

Fichero de solución: **tuberias.b.sh**.

2. (4 puntos) Implemente un guión shell llamado `palabrasfrecuentes.sh` que, dado un fichero de texto y un número  $n$ , devuelva las  $n$  palabras diferentes más frecuentes que haya en dicho fichero, junto con el porcentaje de veces que cada palabra aparece en el fichero. No se distinguirá entre mayúsculas y minúsculas.

Para probar el funcionamiento de su guión shell, puede usar el siguiente fichero de texto:

```
[alumno@localhost ~]$ cat elquijote.txt
01 En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha mucho
02 tiempo que vivía un hidalgo de los de lanza en astillero, adarga antigua,
03 rocín flaco y galgo corredor.
```

Implemente este guión shell resolviendo los siguientes apartados:

- a) (1 punto) El guión shell debe comprobar el número de parámetros, el permiso de lectura del fichero y si el segundo parámetro es realmente un número entero positivo. En caso de error, debe mostrar un mensaje y devolver un código de retorno apropiados. A continuación se muestran algunas ejecuciones incorrectas del guión shell:

```
[alumno@localhost ~]$ bash palabrasfrecuentes.a.sh noexiste
Uso: palabrasfrecuentes.a.sh fichero número
[alumno@localhost ~]$ echo $?
1
[alumno@localhost ~]$ bash palabrasfrecuentes.a.sh noexiste 10
El fichero noexiste no existe o no se puede leer
[alumno@localhost ~]$ echo $?
2
[alumno@localhost ~]$ bash palabrasfrecuentes.a.sh elquijote.txt a
a no es un número o no es mayor que 0
[alumno@localhost ~]$ echo $?
3
```

Fichero de solución: **palabrasfrecuentes.a.sh**.

- b) (0,5 puntos) Amplíe el guión shell para que muestre el total de palabras que contiene el fichero dado. Se entiende por palabra cualquier cadena de caracteres alfabéticos separada de otras palabras por cualquier signo de puntuación o espacio (ya sea un espacio en blanco, un tabulador, un retorno de carro, etc.). Las cadenas que contengan algún número no se considerarán palabras. A continuación se muestran el resultado de la ejecución del guión shell (observa que el segundo parámetro se ignora por ahora):

```
[alumno@localhost ~]$ bash palabrasfrecuentes.b.sh elquijote.txt 5
33
```

Fichero de solución: **palabrasfrecuentes.b.sh**.

- c) (1,0 puntos) Amplíe de nuevo el guión shell para que muestre también el número de veces que cada palabra aparece en el fichero dado. La lista debe aparecer ordenada por número de repeticiones. Recordemos que tampoco se debe distinguir entre mayúsculas y minúsculas.

A continuación se muestran el resultado de la ejecución de esta nueva versión del guión shell (observa que el segundo parámetro se ignora de nuevo):

```
[alumno@localhost ~]$ bash palabrasfrecuentes.c.sh elquijote.txt 5
33
    4 de
    2 un
    2 no
    2 en
    1 y
    ... ..
```

Fichero de solución: **palabrasfrecuentes.c.sh**.

- d) (1,5 puntos) Finalmente, modifique y amplíe el guión shell para que sólo se quede con las  $n$  palabras más frecuentes, siendo  $n$  el segundo parámetro del guión shell, y, para cada una de ellas, muestre el porcentaje de veces que aparece en el fichero dado. A continuación se muestra una ejecución de la versión final del guión shell:

```
[alumno@localhost ~]$ bash palabrasfrecuentes.d.sh elquijote.txt 5
de: 4 (12%)
un: 2 (6%)
no: 2 (6%)
en: 2 (6%)
y: 1 (3%)
```

Fichero de solución: **palabrasfrecuentes.d.sh**.

3. (4 puntos) Implemente un guión shell llamado `usodiscoporusuario.sh` que, dada una lista de directorios, calcule el tamaño total de todos los ficheros (de cualquier tipo) que se encuentren a partir de ellos de forma recursiva y muestre además, para cada usuario propietario de ficheros, el tamaño total de todos sus ficheros. Si se da el caso, el guión shell también debe mostrar al final una lista de todos los ficheros y directorios que no se han podido contabilizar por falta de permisos.

No es necesario que el guión shell realice ninguna comprobación de parámetros.

Para implementar este guión shell, resuelva los siguientes apartados:

- a) (0,5 puntos) Haga que el guión shell devuelva una lista de todos los usuarios con ficheros en los directorios dados como argumentos. Suprima cualquier mensaje de error que pueda aparecer. Lo siguiente es una ejecución del guión shell donde se muestra el tipo de salida que debe producir:

```
[alumno@localhost ~]$ bash usodiscoporusuario.a.sh /etc/ /usr/bin/ /mnt
inadyn
mail
polkitd
root
tss
```

Fichero de solución: **usodiscoporusuario.a.sh**.

- b) (1,5 puntos) Modifique el guión shell para que muestre el tamaño total, en bytes, de todos los ficheros de cada uno de los usuarios obtenidos anteriormente. Al igual que antes, se debe suprimir cualquier mensaje de error que se pueda producir. El resultado que se debe obtener ahora debe parecerse a lo siguiente:

```
[alumno@localhost ~]$ bash usodiscoporusuario.b.sh /etc/ /usr/bin/ /mnt
inadyn: 136
mail: 93
polkitd: 4096
root: 915098478
tss: 7046
```

Fichero de solución: **usodiscoporusuario.b.sh**.

- c) (1 punto) Modifique de nuevo el guión shell para que se muestre también el tamaño total de todos los ficheros de todos los usuarios. Además, haga que los tamaños inferiores a un megabyte se den en bytes y los que sean mayores o iguales se den en megabytes, indicando, en cada caso, la unidad utilizada. Lo siguiente es un ejemplo de ejecución de esta nueva versión del guión shell:

```
[alumno@localhost ~]$ bash usodiscoporusuario.c.sh /etc/ /usr/bin/ /mnt
inadyn: 136 bytes
mail: 93 bytes
polkitd: 4096 bytes
root: 872 MiB
tss: 7046 bytes
Total de todos los usuarios: 872 MiB
```

Fichero de solución: **usodiscoporusuario.c.sh**.

- d) (1 punto) Finalmente, modifique el guión shell para mostrar al final una lista de todos los ficheros y directorios que no se han podido contabilizar por falta de permisos para acceder a ellos. Si no se da ningún caso, el guión shell no mostrará nada. A continuación se muestran un par de ejecuciones de esta última versión del guión shell:

```
[alumno@localhost ~]$ bash usodiscoporusuario.d.sh /etc/ /usr/bin/
inadyn: 136 bytes
mail: 93 bytes
polkitd: 4096 bytes
root: 872 MiB
tss: 7046 bytes
Total de todos los usuarios: 872 MiB
El contenido de los siguientes directorios no se han podido contabilizar:
  '/etc/audit'
  '/etc/ntp/crypto'
  '/etc/cups/ssl'
....
[alumno@localhost ~]$ bash usodiscoporusuario.d.sh /mnt
root: 8192 bytes
Total de todos los usuarios: 8192 bytes
```

Fichero de solución: **usodiscoporusuario.d.sh**.

# Soluciones

1. Las líneas serían las siguientes:

- a) `find /usr/bin/ /etc/ -printf "%s %f %u\n" 2>/dev/null | sort -k3,3 -k1,1rn`
- b) `find /usr/bin/ /etc/ -printf "%s %f %u\n" 2>/dev/null | sort -k3,3 -k1,1rn | uniq -f2 | column -t`

2. Las distintas versiones del guión `palabrasfrecuentes.sh` se muestran a continuación:

a) `#!/bin/bash -u`

```
[ $# -ne 2 ] && echo Uso: $0 fichero número >&2 && exit 1

[ ! -r "$1" ] && echo El fichero $1 no existe o no se puede leer >&2 && exit 2

if echo "$2" | grep -qvx '[0-9]\+' || [ $2 -lt 1 ]
then
    echo $2 no es un número o no es mayor que 0 >&2
    exit 3
fi
```

**Script 1: palabrasfrecuentes.a.sh**

b) `#!/bin/bash -u`

```
[ $# -ne 2 ] && echo Uso: $0 fichero número >&2 && exit 1

[ ! -r "$1" ] && echo El fichero $1 no existe o no se puede leer >&2 && exit 2

if echo "$2" | grep -qvx '[0-9]\+' || [ $2 -lt 1 ]
then
    echo $2 no es un número o no es mayor que 0 >&2
    exit 3
fi

grep -wo '[a-zA-Z]\+' "$1" | wc -l
```

**Script 2: palabrasfrecuentes.b.sh**

c) `#!/bin/bash -u`

```
[ $# -ne 2 ] && echo Uso: $0 fichero número >&2 && exit 1

[ ! -r "$1" ] && echo El fichero $1 no existe o no se puede leer >&2 && exit 2

if echo "$2" | grep -qvx '[0-9]\+' || [ $2 -lt 1 ]
then
    echo $2 no es un número o no es mayor que 0 >&2
    exit 3
fi

grep -wo '[a-zA-Z]\+' "$1" | wc -l

grep -wo '[a-zA-Z]\+' "$1" | tr [:upper:] [:lower:] | sort | uniq -c | sort -nr
```

**Script 3: palabrasfrecuentes.c.sh**

d) `#!/bin/bash -u`

```
[ $# -ne 2 ] && echo Uso: $0 fichero número >&2 && exit 1

[ ! -r "$1" ] && echo El fichero $1 no existe o no se puede leer >&2 && exit 2

if echo "$2" | grep -qvx '[0-9]\+' || [ $2 -lt 1 ]
then
    echo $2 no es un número o no es mayor que 0 >&2
    exit 3
fi
```

```
totalpalabras=`grep -wo '[a-zA-Z]\+' "$1" | wc -l`

grep -wo '[a-zA-Z]\+' "$1" | tr [:upper:] [:lower:] | sort | uniq -c | sort -nr |
  head -$2 | while read linea
do
  repeticiones=$(echo $linea | tr -s ' ' | cut -f1 -d' ')
  palabra=$(echo $linea | tr -s ' ' | cut -f2 -d' ')
  let porcentaje=(repeticiones*100)/totalpalabras
  echo "$palabra: $repeticiones ($porcentaje%)"
done
```

**Script 4: palabrasfrecuentes.d.sh**

### 3. La distintas versiones del guión usodiscoporusuario.sh son las siguientes:

a) #!/bin/bash -u

```
find "$@" -printf "%u\n" 2> /dev/null | sort | uniq
```

**Script 5: usodiscoporusuario.a.sh**

b) #!/bin/bash -u

```
for u in $(find "$@" -printf "%u\n" 2>/dev/null | sort | uniq)
do
  totalusuario=0
  for s in $(find "$@" -user $u -printf "%s\n" 2> /dev/null)
  do
    let totalusuario+=s
  done
  echo $u: $totalusuario
done
```

**Script 6: usodiscoporusuario.b.sh**

c) #!/bin/bash -u

```
total=0
for u in $(find "$@" -printf "%u\n" 2>/dev/null | sort | uniq)
do
  totalusuario=0
  for s in $(find "$@" -user $u -printf "%s\n" 2> /dev/null)
  do
    let totalusuario+=s
  done
  let total+=totalusuario
  if [ $totalusuario -ge 1048576 ]
  then
    let totalusuario/=1048576
    totalusuario="$totalusuario MiB"
  else
    totalusuario="$totalusuario bytes"
  fi
  echo $u: $totalusuario
done
if [ $total -ge 1048576 ]
then
  let total/=1048576
  total="$total MiB"
else
  total="$total bytes"
fi
echo Total de todos los usuarios: $total
```

**Script 7: usodiscoporusuario.c.sh**

d) #!/bin/bash -u

```
FICHTEMP=$(mktemp)
total=0
for u in $(find "$@" -printf "%u\n" 2>$FICHTEMP | sort | uniq)
do
```

```

totalusuario=0
for s in $(find "$@" -user $u -printf "%s\n" 2> /dev/null)
do
    let totalusuario+=s
done
let total+=totalusuario
if [ $totalusuario -ge 1048576 ]
then
    let totalusuario/=1048576
    totalusuario="$totalusuario MiB"
else
    totalusuario="$totalusuario bytes"
fi
echo $u: $totalusuario
done
if [ $total -ge 1048576 ]
then
    let total/=1048576
    total="$total MiB"
else
    total="$total bytes"
fi
echo Total de todos los usuarios: $total
if [ -s $FICHTEMP ]
then
    echo El contenido de los siguientes directorios no se han podido contabilizar:
    cut -f2 -d: $FICHTEMP
fi
rm $FICHTEMP

```

Script 8: usodiscoporusuario.d.sh