

Exercicis Automatització

Exercici 1

La comanda test disposa d'un llarg nombre de funcions que permeten fer comparacions. Empleneu les tres taules següents, respectives als tres tipus d'operadors de comparació de test. Indiqueu quin valor prendria la variable #?, en funció dels valors introduïts (recordeu que 0 és cert a Linux).

Comparació numèrica	
test n1 -eq n2	
test n1 -ge n2	
test n1 -gt n2	
test n1 -le n2	
test n1 -lt n2	
test n1 -ne n2	

Comparació de cadenes de text	
test s1 = s2	
test s1 != s2	
test -n s1	
test -z s1	

Comparació de fitxers	
test -d f1	
test -e f1	
test -f f1	
test -r f1	
test -s f1	
test -w f1	

test-x f1	
test f1 -nt f2	
test f1 -ot f2	

Exercici 2

Expliqueu el funcionament de les cometes d'aquest exercici, i el per què d'això.

```
#!/bin/bash
```

```
comanda=ls
```

```
echo "$comanda"
```

```
echo ` $comanda `
```

```
echo ' $comanda '
```

Exercici 3

Indiqueu justificadament quina és la funció del següent shellsript, indicant quin és el significat més lògic dels paràmetres. Després, expliqueu una comanda equivalent a tot aquest shellsript.

```
#!/bin/bash
```

```
touch tmp
```

```
for i in *.txt
```

```
do
```

```
    grep "examen" $i >> tmp
```

```
done
```

```
wc -l < tmp
```

```
rm tmp
```

Exercici 4

Escriu un shellsript que, a partir d'un paràmetre numèric N, ordeni alfabèticament els noms dels usuaris del sistema i d'aquests mostra els N últims.

Exercici 5

Escriu un shellsript que indiqui si ens trobem als primers o als últims sis mesos de l'any. Cal que tingueu en compte que el sistema pot estar en qualsevol idioma, per tant, utilitzeu els paràmetres de la comanda date per a obtenir un valor vàlid per a qualsevol idioma.

Exercici 6

Escriu un shellsript simple que a partir d'un nombre indeterminat d'arguments, saludi a cadascun dels arguments passats. Per exemple:

```
./exercici Adrián Gerard Raúl Iker  
Hola Adrián  
Hola Raúl  
Hola Iker
```

Exercici 7

Escriu un shellsript que, a partir d'un únic paràmetre N, i utilitzant el bucle while mostra per pantalla una progressió aritmètica d'N termes (1, 2, 3 i 4...) i una progressió geomètrica d'N termes (1, 2, 4, 8, 16,...). El nombre de termes de les successions serà el \$1 d'aquest shellsript.

Exercici 8

Escriu un shellsript que, utilitzant el bucle while, mostri el factorial d'un nombre per pantalla. A continuació, feu el mateix amb un bucle until. El nombre sobre el qual calcular el factorial serà l'únic paràmetre d'aquest shellsript (\$1).

Recordatori: el factorial d'un nombre és el producte de tots els nombre naturals des de 1 fins a aquest nombre. Per exemple, el factorial de 6 (6!) és $6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$.

Exercici 9

Escriu un shellsript que s'intenti autodestruir (com a fitxer). Abans d'executar aquest shellsript, feu una còpia de seguretat de la vostra feina, per si de cas. No podeu utilitzar el nom del fitxer per a eliminar-lo. Comenteu el comportament del shellsript i si s'esborra o no el fitxer.

Exercici 10

Escriu un shellsript que, donats dos nombres diferents passats com a paràmetres mostri per pantalla els nombres que van des d'un fins l'altre (ambdós inclosos, creixentment o decreixentment). Cal que resolueu l'exercici amb un bucle until. Per exemple:

```
./exercici 5 20  
5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

```
./exercici 30 24
```

Exercici 11

Escriu un shellscript que, a partir del fitxer `/etc/passwd`, digui quin és l'interpret de comandes més utilitzat pels usuaris del sistema (és a dir, aquell que utilitzen més usuaris). Aquest shellscript s'ha de resoldre amb una única (i llarga) comanda i sense fer servir cap tipus de bucle.

Exercici 12

Elaboreu un shellscript que rebi com a paràmetres un nombre indeterminat de paraules en minúscula. El que haurà de fer el shellscript serà afegir cadascuna de les paraules a fitxers que s'anomenaran com la seva inicial.

Per exemple, si l'executem amb els paràmetres següents:

```
./exercici ratolí perifèric teclat pantalla
```

Al fitxer "r" s'afegirà la paraula "ratolí", al fitxer "p" s'afegirà la paraula "perifèric", al "t" teclat i al "p", novament, "pantalla".

Exercici 13

Indiqueu justificadament quina és la funció del següent shellscript, indicant quin és el significat més lògic dels paràmetres.

```
#!/bin/bash

for i in `sort $1`
do
    if grep $2 $i > /dev/null
    then
        echo A
        cp $i /tmp
        exit
    fi
done

echo B
```

Exercici 14

Escriu un shellscript que, a partir de la ruta d'un directori determinat passada per paràmetre, si existeix, accedeixi a aquest directori.

- Si no existeix, haurà de crear aquest directori i accedir-hi.

- En cas de no poder crear el directori, mostrarà per pantalla el missatge “No s'ha pogut crear el directori” i quedar-se a la ruta original.

Resoleu aquest exercici sense utilitzar els operadors `&&` i `||`, gestionant la casuística amb estructures alternatives.

Per a comprovar que el vostre shellscript es comporta de la forma esperada, mostreu per pantalla el directori de treball actual. On queda la vostra terminal? On s'havia fet l'execució?

Exercici 15

Escriviu un shellscript que comprovi amb quin usuari s'ha executat. Si aquest s'ha executat com a superusuari, mostrarà el missatge “Sóc l'amo del món”. Si no s'ha executat inicialment com a superusuari, es tornarà a executar a si mateix com a superusuari.

En cap punt del shellscript pot aparèixer el nom del propi shellscript, caldrà que utilitzeu algun altre tipus de mecanisme o paràmetre per a poder executar-lo com a superusuari. Tingueu en compte que si s'ha executat algun `sudo` des del terminal recentment el resultat pot resultar confús: tanqueu i obriu el terminal per a assegurar-vos del resultat.

Exercici 16

Escriviu un shellscript que es mati a si mateix (com a procés). Per a comprovar que funciona correctament, poseu alguna instrucció sota de l'acabament del procés i vegeu que no s'executa.

Exercici 17

Escriviu un shellscript que a partir d'un nom de fitxer (\$1) i un número de parts (\$2), trenqui el fitxer \$1 en \$2 parts. Per a fer-ho caldrà que utilitzeu la comanda que trenca fitxers i alguna operació matemàtica. Proveu a trencar els fitxers en n parts i controleu que sempre es generi el nombre de fitxers que s'ha indicat, exactament.

Exercici 18

Escriviu un shellscript que, donat un fitxer (amb la seva ruta absoluta) i el nom d'un paquet, ens indiqui si el fitxer pertany al paquet o no (és a dir, si va ser instal·lat o requerit pel paquet).

Exercici 19

Teniu un fitxer de text on, a cada línia, hi ha dos noms de fitxer (separats per guió "-"). Escriviu un shellscript que processi totes les línies d'aquest fitxer efectuant la següent operació: si el primer fitxer existeix, el copiarà sobre el segon; si no existeix, copiarà el contingut del fitxer /etc/group sobre el segon.

Exercici 20

Indiqueu justificadament quina és la funció del següent shellscript, indicant quin és el significat més lògic dels seus paràmetres.

```
#!/bin/bash

j=0

for i in $*
do
    if grep ttt $i > /dev/null 2> /dev/null
    then
        j=`expr $j + 1`
    fi
done

echo $# $j
```

Digues quin efecte tindria substituir la línia en blau per: `if grep ttt $i`

Exercici 21

Escriviu un shellscript que a partir d'un directori i una paraula, passats com a paràmetres:

- comprova que el primer paràmetre sigui un directori:
- En cas que no ho sigui, mostrarà un missatge d'error.
- En cas que ho sigui cercarà a tots els fitxers del directori indicat (només el directori, no els seus subdirectoris) la paraula passada com a paràmetre i indicarà, per a cada fitxer, el nombre d'ocurrències de la paraula que s'ha indicat.

Resoleu aquest exercici utilitzant un bucle **for**. No és permet l'ús de la comanda **find**.

Exercici 22

Escriviu un shellscript que rebi com a paràmetre el nom d'un directori i doni com a resultat un únic fitxer, a partir de la compressió i empaquetament del directori i tots els seus subdirectoris.

- Feu les comprovacions necessàries per tal que el shellscript comprovi si el directori a comprimir existeix.
- Feu les comprovacions necessàries per tal que només admeti directoris (no fitxers, per tant).

- c) Feu que el nom del fitxer acabi amb la data actual, en el format AAAA_MM_DD. Per exemple, el backup del 10 de maig de 2021, seria backup_2021_05_10.tar.gz. Caldrà que s'emmagatzemi en la ruta que crida el shellscript.

Si a l'hora de comprimir se us mostren missatges d'error (típicament referents a la falta de permisos), caldrà que silencieu la sortida.

Exercici 23

Escriuiu un shellscript que rebi com a paràmetre un valor alfanumèric V. Aquest shellscript haurà de posar el sistema en el **runlevel** indicat. Abans de posar-lo en aquest runlevel, caldrà que comprovi que, efectivament, aquest és un runlevel legal. En cas que aquest runlevel no existeixi, el sistema mostrarà un missatge per pantalla indicant que no s'ha pogut executar l'acció. En cas que existeixi, el sistema canviarà el runlevel a l'indicat.

Tingueu en compte el tipus d'usuari per a executar aquest shellscript i les comprovacions necessàries per a poder executar-lo.

Exercici 24

Indiqueu justificadament quina és la funció del següent shellscript, indicant quin és el significat més lògic dels paràmetres.

```
#!/bin/bash

for j in ../directori/*
do
    if tail $j | grep $1 > /dev/null
    then
        mv $j .
    fi
done
```

Exercici 25

Escriuiu un shellscript que a partir d'una ruta de directori (absoluta o relativa) passada per paràmetre, examini tots els fitxers d'extensió *.c del directori indicat i compti el nombre de línies no blanques. La sortida del shellscript haurà de ser un fitxer "linies.txt" que contindrà els noms de cada fitxer, junt al nombre de línies no buides d'aquest, ordenat de forma decreixent segons el nombre de línies buides.

```
$ ./exercici ../practiquesC
$ cat linies.txt
practica3.c 1530
```

practica2.c 720
practica1.c 378

Exercici 26

Escriuiu un shellscript que, per a totes les paraules d'un fitxer hipotètic, paraules.txt, cerqui si hi ha alguna de les paraules del fitxer dins els fitxers del vostre directori actual (i el seu arbre de directoris complets) que contingui qualsevol de les paraules del fitxer indicat. En concret, caldrà que, per a cada paraula, digueu quants fitxers la contenen. Podeu utilitzar la comanda **find** per fer la cerca.

```
$ ./exercici paraules.txt  
$ arbre 7  
$ casa 10  
$ cotxe 4
```

Exercici 27

Indiqueu justificadament quina és la funció del següent shellscript, indicant el significat més probable dels seus arguments.

```
#!/bin/bash  
  
if test -f $1  
then  
    if test `wc -c < $1` -gt $2  
    then  
        gzip $1  
    fi  
fi
```

Exercici 28

Escriuiu un shellscript que, donat un nom d'usuari mostri per pantalla a quins grups pertany, ja sigui de forma primària o secundària, és a dir, escriuiu una comanda que faci el mateix que la comanda `groups`, per a qualsevol usuari. Heu de resoldre l'exercici manipulant els fitxers `/etc/passwd` i `/etc/group` i no podeu utilitzar les comandes `group` i `id`.

Exercici 29

Escriuiu un shellscript que mostri els noms dels fitxers ordinaris (no altres directoris) del directori actual amb mida igual o superior a la mida especificada per l'usuari com a paràmetre.

Recomanació: escriuiu un bucle que iteri mostrant els noms de tots els fitxers del directori actual. A continuació, feu que a cada iteració es prengui la mida del fitxer. Finalment, si aquesta mida és superior al paràmetre del shellscrip, mostreu per pantalla el nom del fitxer.

Exercici 30

Escriuiu un shellscrip que faci una còpia al directori updated tots els fitxers ordinaris del directori actual i els seus subdirectoris que hagin estat modificats al llarg de les darreres 48 hores (per fer proves pot resultar-vos útil la comanda touch que permet modificar les dades associades a un fitxer). Assegureu-vos que el directori updated existeix i, si no, creeu-lo.

Exercici 31

Feu un shellscrip que cada N segons comprovi si s'ha muntat un nou volum al sistema. En cas que el shellscrip detecti que s'ha muntat un nou sistema de fitxers, mostrarà per pantalla un llistat del directori base on s'hagi muntat (fitxers i directoris). N serà un paràmetre del shellscrip.

Pista: us pot ser útil monitoritzar el resultat de la comanda df, de manera que si d'una iteració a la següent hi ha més línies, utilitzeu les línies afegides (tail), combinant-lo amb la comanda cut per tal de llistar el contingut del directori base on s'hagi muntat.

Us imagineu que aquest shellscrip, en comptes de llistar, esborrés recursivament tot el contingut del volum muntat?

Exercici 32

Escriuiu un shellscrip que cada N segons comprovi si, al darrer minut, la part entera de la càrrega mitjana del sistema (% CPU) és superior a M. En cas afirmatiu, hauria d'imprimir un missatge indicant-ho. N i M seran arguments del shellscrip (per fer proves, incrementeu la càrrega del sistema amb un shellscrip que tingui un bucle infinit amb una crida a date, per exemple).

Exercici 33

Escriuiu un shellscrip que indiqui quantes particions primàries, quantes esteses i quantes lògiques hi ha al vostre sistema. Tingueu en compte el tipus d'usuari que caldrà per a executar aquest shellscrip.

Exercici 34

Escriuiu un shellscript que, a partir d'un nom de fitxer, un identificador d'usuari (per exemple, "buddy") i un permís (per exemple, r); indiqui si l'usuari té el permís indicat sobre el fitxer. Per a comprovar-ho, caldrà que estúdieu tant els permisos de l'usuari com els del grup o grups als quals pertanyi.

Exercici 35

Escriuiu un shellscript que cada N segons comprovi si el fitxer o directori M té més de P links. En aquest cas, caldrà que mostri un missatge indicant que s'ha superat el màxim de links. N, M i P són paràmetres d'aquest shellscript.

NOTA: Utilitzau `ln -s` per crear enllaços simbòlics dins el directori M i fer proves

Exercici 36

Escriuiu un shellscript que farà de frontend dels compressors/descompressors típics de Linux (gzip i bzip2). El shellscript rebrà un únic paràmetre i es comportarà de la següent manera:

1. Si el paràmetre passat és un fitxer de tipus gz o bz2, el descomprimirà utilitzant el descompressor adequat.
2. Si el paràmetre passat és un fitxer de tipus tar.gz o tar.bz2, farà la descompressió i descompactació pertinents (en un únic pas).
3. Si el paràmetre passat és un fitxer de qualsevol altre tipus o un directori, el sistema mostrarà per pantalla el següent menú:
 - Gzip
 - Bzip2

De manera que es comprimirà en format gz o bz2 en el cas d'un fitxer; o en format tar.gz o tar.bz2 en el cas d'un directori. El fitxer generat haurà de tenir el mateix nom que l'original.

Totes les comprovacions anteriors es poden fer en funció de l'extensió del fitxer. Suposarem que si els fitxers tenen la extensió correcta, són del format que indica la seva extensió.

Exercici 37

Indiqueu justificadament quina és la funció del següent shellscript:

```
#!/bin/bash

temps=5
limit=40

while sleep $temps
do
    procs=`ps h -ef | wc -l`
    echo `date` : Processos = $procs
    if test $procs -gt $limit
```

```
then
    echo -e "Límit superat"
fi
done
```

Exercici 38

Escriu un shellscript que, cada N segons, comprovi quin és el nombre total de processos en execució a la màquina. En cas que superi el valor M, el shellscript haurà d'enviar el senyal SIGTERM a tots els processos que estiguin executant el programa P i afegir al fitxer F els pids dels processos a qui ha enviat el signal. N, M, P i F seran paràmetres del shellscript (i en aquest ordre); per exemple, 60 100 firefox /tmp/log.

Exercici 39

Escriu un shellscript tal que, cada N segons, comprovi si hi ha algun dispositiu d'emmagatzemament amb un percentatge d'ocupació total superior al M%; en cas afirmatiu, ha de mostrar un missatge indicant el nom i el punt de muntatge del dispositiu. N i M seran dos paràmetres d'entrada del shellscript.

Exercici 40

Escriu un shellscript tal que, cada N segons, verifiqui si existeix/en, al directori actual, fitxer(s) tal que el seu nom comenci per core. En cas afirmatiu, comprovarà les seves mides; si la mida d'algun d'aquests fitxers supera el valor M, caldrà esborrar-lo i afegir al fitxer F el nom i la mida del fitxer esborrat. N M i F seran paràmetres del shellscript (i en aquest ordre); per exemple, 60 1000000 /tmp/log.

Exercici 41

Escriu un shellscript que rebi com a paràmetres la ruta d'un directori passat, R, i un valor numèric, N. Aquest shellscript, per a cada fitxer del directori actual (i no dels seus directoris fills), comprovarà el seu ús de disc i si aquest és superior a N bytes, mostrarà el missatge següent per pantalla: **"NOM_FITXER: Esborra / Manté / Comprimeix?"**

I en la mateixa línia permetrà la introducció de cadascuna de les opcions següents:

- **E:** esborrarà el fitxer indicat.
- **M:** no farà cap canvi al fitxer indicat.
- **C:** guardarà el nom de fitxer indicat per a comprimir-lo.

En acabar les qüestions a l'usuari, el sistema generarà un únic fitxer exercici.tar.gz amb tots els fitxers que s'hagi decidit comprimir. Si no hi ha cap fitxer a comprimir, no generarà el fitxer exercici

.tar.gz. Finalment, caldrà que el shellscript també guardi la informació de què s'ha fet amb cada fitxer en el fitxer exercici.log. Tant el fitxer de log com el tarball s'han de generar dins el mateix directori que comprovem.

```
./exercici_directori_prova 1024 # fitxers de més d'un kb
```

```
fitxer_prova1: Esborrar / Mantenir / Comprimir? E
```

```
fitxer_prova2: Esborrar / Mantenir / Comprimir? M
```

```
fitxer_prova4: Esborrar / Mantenir / Comprimir? M
```

```
fitxer_prova7: Esborrar / Mantenir / Comprimir? C
```

```
fitxer_prova8: Esborrar / Mantenir / Comprimir? C
```

```
# Fem algunes proves per a veure que s'ha executat bé
```

```
# Llistem el contingut del fitxer de log
```

```
cat directori_prova/exercici.log
```

```
fitxer_prova1: Esborrat
```

```
fitxer_prova2: Mantingut
```

```
fitxer_prova4: Mantingut
```

```
fitxer_prova7: Comprimit
```

```
fitxer_prova8: Comprimit
```

Exercici 42

Escriu un shellscript tal que, cada N segons, comprovi quants fitxers hi ha al directori actual que siguin modificables per tots els usuaris de la màquina. Si el nombre d'aquests fitxers supera el valor M i la suma de les mides d'aquests fitxers supera el valor P, caldrà esborrar el fitxer més gran d'aquests i afegir al fitxer F el nom i la mida del fitxer esborrat. N, M, P i F seran paràmetres del shellscript (i en aquest ordre), P està especificat en bytes; per exemple, 60 5 100000 /tmp/log.

Exercici 43

Escriu un shellscript que demani a l'usuari que introdueixi el nom d'un fitxer (aquest contindrà una llista de noms personals i dates de naixement) i, a partir d'aquest, indiqui quina és la persona més vella i la més jove.

El fitxer seguirà el següent format:

Manuel 15/07/89

Jimena 02/03/92

Gregori 01/07/2015

Cristina 15/02/1999

Nota: recordau que l'usuari no ha d'introduir informació per paràmetre