

## Exercicis Bash (I)

### Comandes bàsiques

**1.-a)** Qui és el propietari de l'arxiu */etc/passwd*? I el seu grup? Els usuaris que no siguin ni el propietari ni pertanyin al seu grup, ¿quins permisos tenen sobre aquest arxiu?

**b)** Crea en el teu directori personal un subdirectori anomenat “copia” i copia allí l'arxiu */etc/passwd*. Qui és ara el propietari d'aquest nou arxiu? I el seu grup? I els seus nous permisos? I la seva data d'última modificació? ¿Quin paràmetre de *cp* hauries d'escriure per a què tot això (propietari, permisos i data d'última modificació) no canviés a la copia?

**2.-a)** Escriu la comanda que mostra un llistat dels noms dels arxius que hi ha dins la carpeta */etc* (incloent els possibles ocults) que tinguin l'extensió “conf”, ordenant-los a més per la darrera data de modificació de manera decreixent.

**b)** Escriu la comanda que mostra ordenats per tamany (de més petit a gran) tots els arxius amb un nom de tres lletres (i sense extensió) que estiguin dins la carpeta */usr/bin*.

**bBIS)** Escriu la comanda que mostra ordenats per tamany (de més petit a gran) tots els arxius amb un nom de tres lletres (amb qualsevol extensió) que estiguin dins la carpeta */usr/bin*.

**3.-a)** Canvia el nom de l'arxiu *passwd* del directori *copia* creat a l'exercici 1, pel nom *palabras*, utilitzant per això la comanda *mv*.

**b)** Crea un enllaç dur a l'arxiu *palabras*, en el mateix directori, anomenat *palabras2*. ¿Què ocorre si ara s'esborra *palabras*?

**c)** Crea ara un enllaç tou (un "accés directe") a l'arxiu *palabras2*, en el mateix directori, anomenat *palabras3*. ¿Què ocorre si ara s'esborra *palabras2*?

**d)** Finalment, esborra el directori *copia* sencer

**4.-a)** Crea un directori denominat *pepito* i copia allí tots els arxius de la carpeta */bin* que comencin per *a* o *b*

**b)** Crea un directori denominat *manolito* i copia allí tots els arxius de */etc* el nom dels quals sigui de 5 lletres

**c)** Crea un directori *miguelito* i copia allí tots els arxius l'extensió dels quals sigui “.sh” i estiguin a la carpeta */etc/profile.d*.

**d)** Si escrivim *mv [a-b]??ab\*.tx[!t] /home/usuari* , ¿l'arxiu “cxxab.txa” es mouria? ¿I l'arxiu “axxab.txt”?

**e)** ¿Quina diferència hi ha entre afegir el paràmetre *-u* a la comanda *cp* i afegir-ne els paràmetres *-u -b* ?

**f)** Què fa la comanda *rm !(\*.??)* ?

5.-Crea dins la teva carpeta personal una carpeta anomenada *hola*, i mou-te-hi a dins. Sense moure't d'allí, i usant sempre rutes relatives:

- a) Crea dins d'*hola* una altra amb nom *quetal*
- b) Crea dins d'*hola* un arxiu de text anomenat *prova.txt*. Inclou dintre d'aquest fitxer un text qualsevol
- c) Copia aquest arxiu a la carpeta *quetal*
- d) Esborra aquest arxiu de la carpeta *hola*
- e) Mou l'arxiu des de la carpeta *quetal* altre cop fins *hola*
- f) Oculta l'arxiu *prova.txt* (és a dir, canvia-li el nom per a què comenci amb un punt)
- g) Crea una subcarpeta dins de *quetal* anomenada *moltbe* i copia-hi l'arxiu *.prova.txt*
- h) Dins de *moltbe*, crea una altra carpeta anomenada *itu* i copia-hi també *.prova.txt*
- i) Canvia el nom de l'arxiu *.prova.txt* de la carpeta *itu* pel de *avorp.txt*
- j) Crea una carpeta anomenada *lalala* a l'escriptori teu
- k) Copia recursivament tot el contingut (arxius i carpetes) de la carpeta *hola* dins d'aqueta carpeta *lalala*
- l) Esborra recursivament la carpeta *lalala* amb tot el seu contingut (arxius i carpetes)

6.-a) Crea una ordre alias denominada *midir* que sigui equivalent a la ordre *ls -lah - -color=auto* . Què fa?

b) Crea una ordre alias denominada *midir2* que sigui equivalent a la ordre *ls -XrR --color=never* . Què fa?

c) Crea una ordre alias de la comanda *date* (denominada com vulguis) que només mostri hora i minut actual

d) Desfés tots aquests àlies que acabes de crear.

7.-a) Si assignem a una variable creada per nosaltres (l'anomenarem "D") la ruta */etc* ,¿ com podríem anar de forma ràpida al directori anterior utilitzant aquesta variable?

b) Afegeix la ruta *"/opt"* a la llista de rutes que ja formen el valor de la variable d'entorn predefinida *PATH* sense esborrar les que ja hi són. ¿Què conseqüències té fer això?

c) Raona què fa la comanda *echo "scale=4;\$RANDOM/3" | bc*

d) Obre un terminal i executa la comanda *cd -* . Obtindràs un error. Seguidament, mou-te a alguna carpeta qualsevol i torna a executar *cd -* . ¿Què passa ara? ¿Què hi té a veure el valor de la variable *OLDPWD*?

8.-Alguns dels símbols especials que es poden introduir com a part el valor de la variable PS1 són aquests:

|  |   |  |
|--|---|--|
| \u : nom de l'usuari actual                  | \h : nom de la màquina  | \W : directori actual de treball           |
| \w : ruta actual completa                    | \d : data en format "diesetmana mes dia"                                  | \H : nom de la màquina complet             |
| \t : hora actual en format 24-hores HH:MM:SS | \T : hora actual en format 12-hores HH:MM:SS                              | \@ : hora actual en format 12-horas am/pm  |
| \\ : una contrabarra                         | \\$ : si el UID efectiu és 0 (root), imprimeix una #, sinò, un \$         | \j : N° de treballs executant-se en 2ª pla |
| \n : salt de línia                           | \e[ : caràcter d'escapament per afegir codis ANSI (del tipus \e[31m, etc) |  |

Per exemple, el valor de *PS1* = "[\u@\h \W] \\$" és el valor actual del prompt. Un cop sabut això:

- Primer copia el valor actual de *PS1* en una altra variable d'entorn pròpia anomenada *SAVE*.
- Seguidament, fes que el prompt contingui la data i hora actual, a més de la ruta completa de la carpeta actual i el nom complet de la màquina, i també un missatge de benvinguda.
- Restaura un altre cop el valor que ja tenia *PS1* anteriorment a partir de la variable d'entorn *SAVE*.
- Finalment, destrueix la variable *SAVE* i comprova que ja no existeixi
- Dedueix, abans d'executar-la, què fa aquesta línia: `export PS1="\e[36;1m\u@\e[32;1m\H> \e[0m"`

9.-a) Escriu la línia següent:

```
export LS_COLORS="di=1;4;fi=31;47;ln=34;106;ex=35:*.rpm=90:*.jpg=7;35;45"
```

Observa els seus resultats i raona el perquè d'aquests.

**NOTA:** Consulta les taules següents (o també [https://misc.flogisoft.com/bash/tip\\_colors\\_and\\_formatting](https://misc.flogisoft.com/bash/tip_colors_and_formatting)) :

#### LS\_COLORS FILE CODES

bd = Block Device (buffered)  
ca = File with Capability  
cd = Character Device (unbuffered)  
**di = Directory**  
do = Door  
ec = End Code  
**ex = Executable File (+x)**  
**fi = File**  
lc = Left Code  
**ln = Symbolic Link**  
**mh = Multi Hard Link** - regular file with more than one link  
mi = Missing File - pointed to by a symbolic link  
**or = Orphaned Symbolic Link** - points to a nonexistent file  
**ow = Directory Other-Writable(o+w)**  
pi = Fifo File (named pipe)  
rc = Right Code  
sg = File setuid (g+s)  
so = Socket File  
st = Directory Sticky (+t)  
su = File setuid (u+s)  
tw = Directory Sticky and Other-Writable(+t,o+w)  
**\*extension or \*.extension = any file that ends in extension**

| CONSOLE COLOR & FORMAT CODES (man console_codes) |      |            |               |        |
|--|------|------------|---------------|--------|
| COLOR  | TEXT | BACKGROUND | FORMAT        | FORMAT |
| Black  | 30   | 40         | reset         | 0      |
| Red  | 31   | 41         | bold/bright   | 1      |
| Green  | 32   | 42         | half-bright   | 2      |
| Yellow   | 33   | 43         | underline     | 4      |
| Blue   | 34   | 44         | blink         | 5      |
| Magenta  | 35   | 45         | inverse       | 7      |
| Cyan   | 36   | 46         | conceal       | 8      |
| Light Gray                                       | 37   | 47         | normal        | 22     |
|  |      |            | underline off | 24     |
|  |      |            | blink off     | 25     |
|  |      |            | inverse off   | 27     |

Syntax is: `file_code=format;text;background`

**b)** A partir del que has après, fes el possible per fer un “ls” que mostri els arxius amb extensió \*.png de color verd i subratllats, i els arxiu \*.txt de color blaus, en negrita i amb color de fons blanc, i que ho mostri sempre així.

**10.-a)** Usa la comanda *which inkscape* per esbrinar on està instal·lat el programa Inkscape. Un cop sabut això, crea a l'escriptori amb la comanda *ln* un enllaç tou que apunti a l'Inkscape. Si fas clic sobre ell, s'iniciarà?

**b)** Una comanda alternativa a *which* és *type*, la qual informa a més de si el programa indicat és realment un programa per sí mateix o bé un "built-in" del bash. Utilitza la comanda *type* per esbrinar si les comandes següents són executables externs o bé estan incorporats dins del Bash: *echo* , *history* , *cd* , *cp*

**c)** Si un fitxer es troba dins la carpeta “/usr/bin”, ¿què podem dir que és? ¿I si es troba dins de “/lib”? ¿I dins de “/etc”?

**11.-a)** ¿Quina diferència hi ha entre els arxius “/etc/bash.bashrc” i “~/.bashrc” a nivell de concepte? ¿I entre els arxius “/etc/bash.bashrc” i “/etc/profile”? (Consulta els apunts per saber la resposta)

**b)** Fes el necessari per a què quan obris un terminal s'executi automàticament la comanda *yes* Què passa?

**c)** Escriu el necessari per tal de què quan obris un terminal es mostri un missatge de benvinguda dit per un elefant (mitjançant la comanda *cowsay*).

**d)** Consulta les pàgines de manual *motd* i *issue* per saber per a què serveixen els arxius */etc/motd* i */etc/issue* i consulta seguidament l'apartat "Issue Escapes" de la pàgina de manual *agetty* per saber com mostrar en un terminal virtual la direcció IP de la màquina abans de qualsevol inici de sessió i per saber com mostrar un missatge de benvinguda cada cop que un usuari iniciï sessió en un terminal virtual (o sessió SSH, que seria el mateix en aquest cas), respectivament.

**12.-a)** Canvia els permisos d'un arxiu qualsevol per a què només tu el puguis llegir i escriure, i els del teu grup que puguin executar-lo. Fes-ho fent servir números (4,2,1) i lletres (r,w,x)

**b)** ¿Com es podria fer el mateix però aplicant-ho a una carpeta i tot el seu contingut? ¿Què vol dir “llegir una carpeta”, “escriure una carpeta” i “executar una carpeta”?

**c)** ¿Per a què serveix la comanda *chown*? Digues un exemple d'ús

**d)** ¿Quin permís ha de tenir la carpeta de destí per a què en una comanda *cp* es pugui copiar allà un fitxer qualsevol? ¿I la carpeta origen?

**13.-a)** Què passa si escrius `!9` al terminal? Per què en l'ordinador dels teus companys no s'executa el mateix ?

**b)** Fes el necessari per a què qualsevol comanda que comenci per `ls` no es gravi al teu historial

**c)** Busca "a Internet" per a què serveix la comanda `set +o history` ¿Quina diferència hi ha amb la comanda `history -d`?

**d)** Per què quan obres dos terminals i comences a escriure comandes diferents en cadascun, l'historial que es veu és diferent si se suposa que l'historial s'escriu en un únic fitxer, el `~/.bash_history`?

**14.-a)** A partir de la taula de codis de colors del primer exercici, dedueix què fan cadascuna de les següents línies (en aquest cas la sintaxis és: `"\e[codiFormat1;codiFormat2;...;colorText;colorFonsm"`

```
echo -e "\e[32m Hola \e[0m"
```

```
echo -e "\e[1;32m Hola \e[0m"
```

```
echo -e "\e[1;32;45m Hola \e[0m"
```

```
echo -e "\e[32;45m Hola \e[0m"
```

```
echo -e "\e[1;4;32;45m Hola \e[0m"
```

```
echo -e "\e[7;1;4;32;45m Hola \e[0m"
```

## Exercicis Bash (II)

### Filtres

- 1.-Crea un arxiu en el teu directori \$HOME denominat *fich.sal* que contingui el nom de tots els arxius que hi hagi als directoris */bin* i */sbin*. Utilitza per fer això una sola comanda (amb comodí i redireccionador).
- 2.-Ordena numèricament l'arxiu */etc/passwd* segons el camp tercer i graba el resultat en un arxiu denominat *uid.ord*. Seguidament fes el mateix però usant el quart camp, i afegeix aquesta nova sortida al mateix arxiu.
- 3.-a) Com podem usar la comanda *cut* per quedar-nos només amb la informació relativa a l'hora, de tota la que treu la comanda *date*?  
b) Amb la comanda *date* i el seu paràmetre *-s* podem canviar l'hora i data del sistema. Consulta el document de teoria per aprendre exactament com fer-ho, i prova-ho.
- 4.-Crea un àlies, anomenat com vulguis, que conti el número d'arxius -i només els arxius- que hi ha en el directori */usr/bin*.
- 5.-a) Utilitza l'ordre *grep* per visualitzar, de tots els arxius inclosos a la carpeta */etc*, només les línies que tinguin la paraula "file", mostrant a més el seu respectiu número de línia dins del fitxer on estan  
b) ¿Com es faria per fer exactament el contrari, per veure totes les línies que NO tinguin la paraula "file"? Quantes línies són?
- 6.-Crea un arxiu denominat *persones* que contingui, separats per una coma, el nom, cognom i edat de 7 persones (una diferent a cada línia). A partir d'aquí:  
a) Llista el nom i cognom de totes les persones de l'arxiu anterior l'edat dels quals sigui 27 anys, mostrant a més la posició de la seva línia corresponent dins de l'arxiu.  
b) Llista l'edat de totes les persones el nom de les quals tingui una S majúscula. ¿Quantes persones són?  
c) Ordena alfabèticament per cognoms l'arxiu anterior i genera un nou arxiu al directori HOME que contingui aquesta ordenació denominat *persones.ordre.alfabetic*.  
d) Crea un altre arxiu denominat *persones2* amb un contingut similar al de *persones* però que contingui només 2 persones Uneix el contingut dels 2 fitxers en un de sol, anomenat *persones3*  
e) Esborra amb una sola ordre tots quatre fitxers de cop
- 7.-a) ¿Què fan les comandes *ls -ltu | grep "root" | wc -l* ?  
b) ¿Què fan les comandes *ls -a | grep -E "\.+\$"* ?  
c) Escriu les ordres entubades necessàries per veure per pantalla les carpetes -i només les carpetes!- que hi ha dins de la carpeta */etc* .  
d) Escriu les ordres entubades necessàries per veure per pantalla només la contrasenya encriptada de l'usuari "usuari" (és el segon camp de les línies de l'arxiu */etc/shadow*)
- 8.-a) Utilitzant la comanda *head*, *tail*, i les canonades, mostra les línies de la 5ena a la 8ena d'un arxiu qualsevol. PISTA: consulta els apunts i fixa't quina diferència hi ha entre escriure *head -n 5 fitxer.txt* i *head -n -5 fitxer.txt* i entre *tail -n 5 fitxer.txt* i *tail -n +5 fitxer.txt*  
b) ¿Què fan les comandes *head -n 3 /etc/fstab ; tail -n 2 /etc/fstab* ?
- 9.-Canvia, amb la comanda *sed*, totes les paraules "usuari" que hi ha dins del fitxer */etc/passwd* per la paraula "PRINGAT" i grava el resultat en un altre fitxer diferent.

10.-De la sortida de la comanda *mount*, sel.lecciona només les línies que comencen per *"/dev"*, i, un cop sel.leccionades, talla-les per tal de mostrar només el punt de muntatge (és a dir, el valor que apareix entre "on" y "type")

11.-a) Conta els fitxers de dins la teva carpeta personal el nom dels quals comenci per *".bash"*

b) Conta quantes línies hi ha dins l'arxiu */etc/passwd* que continguin la paraula "nologin" al final de la línia. ¿I les que no la contenen?

**NOTA:** Recorda que les expressions regulars només es poden fer servir si utilitzem *grep* o *sed*. Per tota la resta de comandes, caldrà fer servir comodins

c) Utilitza l'ordre *grep* per visualitzar, de tots els arxius amb extensió *.conf* inclosos a la carpeta */etc*, només les línies que comencin amb una lletra majúscula

12.- Analitza línia a línia el que fa aquest conjunt entubat i digues quin és el significat de la seva sortida global: *history | sed -r "s/^ +//;s/ / /" | cut -d" " -f2- | sort | uniq -c -i | sort -rn | head -n 5*

13.-Digues quina diferència hi ha en executar les següents comandes (consulta els apunts per saber-ho):

*ls /root 2> registre.log*

*ls /root &> registre.log*

*ls /root | tee registre.log*

14.-a) Escriu la següent línia: *ls && cp \* && ls* . ¿Què s'executa i què no i per què? Pots consultar els apunts

b) Escriu ara això: *ls ; cp \* ; ls* . ¿Què és el que passa ara i per què?

c) ¿I si escrius això *ls || cp \* || ls*?

15.- a) ¿Quina diferència hi ha entre escriure *grep "^....\$" fitxer* i *grep "^\....\$" fitxer*?

b) ¿Per què és diferent el que veus en executar *echo "scale=4;3/5" | bc | grep -E "[0-9]\*\.[0-9]+"* del que veus en executar *echo "scale=4;3+5" | bc | grep -E "[0-9]\*\.[0-9]+"* ? (l'únic canvi que hi ha és que al primer exemple dividim i al segon sumem). ¿I si fem *echo "scale=4;3+5" | bc | grep -E "[0-9]\*\.[0-9]+"* ?

c) Sabent que l'arxiu */var/log/auth.log* guarda, entre altres coses, els intents de fer "sudo", per què veus el que veus quan executes *grep "COMMAND" /var/log/auth.log | sed -r "s/ +//g" | cut -f 14- -d " " | sort -u* ?

d) A quin patró correspon aquesta exp.reg: *^(http://|https://)\. +* ? I aquesta: *[a-z\.-]+@[a-z\.-]+\.[a-z]{2,4}* ?

16.-Si tenim un arxiu de text amb el següent contingut:

*hola hoola ala*

*ala alo halo*

*aaloo oollaa*

*ylu ili ole ole*

Digues, abans de provar-ho, quin hauria de ser el resultat esperat d'executar les següents comandes, i per què. (Després de fer-ho, podràs comprovar si no t'has equivocat, però no abans):

a) *grep -vc "[ola]" nomFitxer*

b) *grep -iE "^ola" nomFitxer*

c) *grep -niE "o(la)+\$" nomFitxer*

d) *grep -nE "ol.\*a.\*" nomFitxer*

e) *grep -nvE "(ola|ila)\*" nomFitxer*

f) *grep -rE "[^ol]la" nomFitxer*

- 17.- a)** ¿Com visualitzaries en majúscules, utilitzant la comanda sed, el contingut de l'arxiu /etc/fstab?  
**b)** ¿Què fa la comanda `ls | sed -nr "/^.$$/p" ?` I `ls | sed -nr "/^{2,7}$/p" ?`  
**c)** ¿Què fa la comanda `sed -r "s/( |\t)/n/g" /etc/crontab | sed -r "/^$/d" | sort | uniq -i -c | sort -nr` ?

**18.-a)** Crea un fitxer anomenat "hola.sed" amb el següent contingut...:

```
#!/usr/bin/sed -f
3d
4d
```

...i executa seguidament aquesta comanda: `sed -f "hola.sed" /etc/fstab` . ¿Què passa?

- b)** ¿Seria el mateix que executar `sed "3d;4d" /etc/fstab`?  
**c)** ¿I que executar `sed -e "3d" -e "4d" /etc/fstab`?  
**d)** ¿Què fa la comanda `sed -e "3i HOLA GUAPO" -e "3d" /etc/fstab`?

**19.-** Crea un fitxer anomenat "a.txt" amb el següent contingut:

*Els Beatles són el millor  
grup de música  
de la història.*

I un altre fitxer anomenat "b.txt" amb el següent contingut:

*The Who són el millor  
grup de música  
del món.  
O potser els Pink Floyd.*

- a)** Executa ara la comanda `diff {a,b}.txt` i explica el significat del que surt a pantalla (consulta els apunts per conèixer com funciona la comanda diff)  
**b)** ¿Hi ha alguna diferència entre executar la comanda anterior i aquesta: `diff {a..b}.txt` ? Per què?  
**c)** D'altra banda, ¿què creus que sortirà a la pantalla si executes `echo {-50..50}`? (No l'executis abans de respondre mentalment)