

Sistemes Operatius 1

Sessió de problemes 3 – 4 de març del 2020

Introducció

En aquesta sessió de problemes es presenten exercicis relacionats amb el segon tutorial. Aquesta sessió acaba amb un exercici que s'haurà d'entregar a través del campus i que serà avaluat per els vostres companys.

Exercici 1

La comanda "find" és una de les comandes més potents en l'entorn Linux per buscar fitxer o directoris. Aquí teniu diversos exemples del seu ús, analitzeu el que imprimeix per pantalla i les opcions de execució usades.

```
find gutenber/ -type d
find gutenber/ -type f
find gutenber/ -name "*.txt"
find gutenber/ -name "*.txt" -size +512k -size -1024k
find gutenber/ -name "*.txt" -size +512k -size -1024k > fitxer.txt
wc -l fitxer.txt
find gutenber/ -name "*.txt" -size +512k -size -1024k | wc -l
find gutenber/ -name "*.txt" -size +512k -size -1024k -exec wc -l {} \;
find gutenber/ -name "*.txt" -type f -size +512k -exec ls -l {} \;
```

Exercici 2

La comanda "ps aux" ens retorna tota informació sobre els processos que s'estan executant al sistema. Després de llegir i revisar el man de la comanda:

1. Quina diferència hi ha entre la crida de la comanda "ps" i "ps aux"?
2. Trobeu tots els processos executats **només** per l'usuari root. Feu servir una canonada per trobar-los, eviteu fer servir bucles.
3. Trobeu tots els processos executats pel vostre usuari. Us podeu ajudar de la comanda "whoami" que us retorna el vostre nom d'usuari. Feu servir una canonada per trobar-los, eviteu fer servir bucles.

Exercici 3

A la pràctica 1 heu fet ús de la comanda `awk`. La comanda `awk` inclou, de fet, tot un llenguatge de programació propi que permet fer un munt de coses. Com a exemple anem a veure les comandes `BEGIN` i `END`. Aquestes ens permeten incloure accions abans i després de l'acció a realitzar.

Analitzeu les següents comandes (les dues darreres són dues línies que són una única comanda)

```
ls -l gutenbergetext00/ | grep -F '.txt' | awk '{print $9}'

ls -lh gutenbergetext00/ | grep -F '.txt' | awk 'BEGIN {print "Fitxers .txt:"} {print $9}'

ls -lh gutenbergetext00/ | grep -F '.txt' | awk 'BEGIN {print "Fitxers .txt:"} {sum+=$5; print $9} END {print "Total:" sum}'
```

Exercici a entregar

Es proposa a continuació un exercici que forma part de les activitats presencials (AP) de l'assignatura. Els detalls de l'avaluació s'indiquen al final.

El problema consisteix en fer un script que rep com argument un directori. Donat aquest directori pare, l'script ha de recórrer tots els subdirectoris que hi ha per sota i fer el recompte del nombre de fitxers a cada un d'ells. Per a cada subdirectori ha d'indicar el nombre de fitxers que hi ha en aquell subdirectori.

```
$ ./exercici.sh gutenbergetext00/
directori: gutenbergetext00 nfiles: 13
directori: gutenbergetext00/etext00 nfiles: 47
directori: gutenbergetext00/etext01 nfiles: 38
directori: gutenbergetext00/etext02 nfiles: 52
directori: gutenbergetext00/etext02/andvll1h nfiles: 140
directori: gutenbergetext00/etext02/mtdtll1h nfiles: 6
...
```

No és necessari que el resultat aparegui en l'ordre mostrat en aquest exemple.

Per a la implementació d'aquest exercici només es permeten fer servir les funcions descrites al tutorial de la pràctica 2 i 1, amb totes les opcions que tinguin disponibles les comandes encara que no estiguin descrites al tutorial.

L'exercici ha de realitzar les següents comprovacions:

- Comprovar que el nombre de paràmetres sigui correcte. Si no són correctes el programa haurà de sortir amb un codi d'error.
- Comprovar que el paràmetre passat com a argument existeix i és un directori. En cas que no existeixi o no sigui un directori, el programa haurà de sortir amb un codi d'error.

Si l'execució s'ha realitzat de forma correcta, l'script haurà de sortir amb un codi d'execució correcta (típicament és un 0).

Avaluació de l'exercici

L'avaluació d'aquest exercici només tindrà els valors 10, 5 i 0. Per poder obtenir la qualificació de 10 cal entregar l'exercici i la rúbrica. L'exercici ha d'intentar resoldre el problema proposat fent servir només les instruccions presentades al primer i segon tutorial. En cas que el codi d'un grup no funcioni i la rúbrica corresponent no ho reflecteixi, el grup que ha fet la rúbrica obtindrà un 5. Finalment, en cas que no s'entregui l'exercici o la rúbrica, o la solució entregada faci servir instruccions no permeses o la solució no tingui res a veure amb l'exercici proposat l'avaluació serà de zero.

Es demana entregar, en grups de dues persones, l'exercici proposat aquí a través del campus. S'entregarà fent servir el número de grup que se us ha assignat a classe presencial. Així, per exemple, si sou el grup 31 entregueu l'exercici amb un únic fitxer que s'anomeni grup31.sh. Eviteu posar noms a l'exercici per assegurar l'anonimat de l'entrega.

Hi ha tot el dia d'avui, dia 4 de març, per entregar l'exercici. Dijous 5 es faran públics al matí els exercicis entregats a l'enllaç següent

<https://drive.google.com/open?id=1yQ4zWcWq7dnDozb0UtohNmtZNuWkzEEa>

Cada grup d'estudiants haurà d'avaluar dos exercicis diferents: en particular, cada grup avaluarà els dos exercicis amb un número de grup immediatament inferior al que tinguin assignats ell@s. El grup 31 haurà d'avaluar doncs els grups 30 i 29 (o els que hi hagi immediatament inferiors).

L'avaluació es farà fent servir la rúbrica que hi ha disponible al campus. Feu servir l'exemple proporcionat per assegurar el bon funcionament de l'exercici entregat. Avaluar cada exercici no hauria de portar més de 15 minuts. Cada grup entregarà les seves dues rúbriques via campus. Per facilitar la revisió de les rúbriques es demana que cada rúbrica tingui el nom rubrica_grupoXX_grupoYY.ods on grupoXX és el grup avaluat i grupoYY és el grup que fa l'avaluació.

La data termini per entregar les dues rúbriques assignades és el proper divendres. L'avaluació assignada per vosaltres ens ajudarà a decidir quines solucions es mostren a la propera sessió de problemes.