

Pràctica 2

Març 2020

En aquest document es mostren els exercicis a programar per a la pràctica 2. Per tal d'intentar seguir la “filosofia” dels scripts a Unix els arguments es passen per línia de comandes i la sortida s'imprimeix per pantalla.

A continuació hi ha dues seccions: la primera secció explica els scripts a programar i la segona indica l'informe que s'ha d'entregar amb els scripts. La puntuació dels scripts és d'un 80% i la de l'informe un 20%.

1 Scripts a programar

Tots els scripts necessiten arguments per línia de comandes. En cas que no s'especifiquin, en executar, el nombre necessari d'arguments l'script haurà de mostrar un missatge d'error per pantalla i sortir amb un codi d'error (`exit 1`). En cas que s'especifiquin tots els arguments, podeu suposar que s'han especificat de forma correcta. Si l'script s'executa correctament, caldrà que aquest surti amb un codi d'execució correcta, `exit 0`.

Per exemple, suposeu un script en què s'han d'especificar dos arguments: el primer argument ha de ser un número i el segon una cadena. Si no s'especifiquen els dos arguments, caldrà donar un missatge d'error per pantalla i sortir de l'script amb el codi d'error. Si s'especifiquen els dos arguments, podeu suposar que el primer argument és un número i el segon una cadena.

Tots els noms de fitxers, directoris i cadenes a tractar no tindran espais ni caràcters especials i, a més, podeu suposar que tots els fitxers a tractar són fitxers de text pla. En alguns d'aquests exercicis es passa com a paràmetre un directori: l'script ha de funcionar independentment del directori que es passi com a paràmetre. També podeu suposar que no hi ha enllaços simbòlics en els fitxers analitzats. Al campus teniu disponible un fitxer comprimit que conté un gran nombre de fitxers de text amb els quals podeu fer proves.

Cada script té la mateixa puntuació. En cas que es faci servir redirecció per implementar un exercici la puntuació màxima serà la meitat de la puntuació associada a l'exercici. Feu servir canonades! Per fer aquests exercicis només està permès fer servir les instruccions que hi ha al document del tutorial de la pràctica 2 i 1 llevat que s'indiqui el contrari. Podeu fer servir qualsevol opció de la comanda encara que no estigui indicada al tutorial. En cas que es faci servir una instrucció no permesa la puntuació de l'exercici serà de 0 punts.

Es recomana fer els exercicis junts, en parella, ja que hi ha exercicis que són difícils de resoldre sense haver resolt l'anterior. No us repartiu els exercicis per treballar de forma independent!

Exercici 1

Fer un script anomenat `exercici1.sh` que només té un argument: un directori. L'script ha d'analitzar, de forma recursiva i per a cada subdirectori que trobi, la mida (en bytes) dels fitxers que hi ha dintre d'aquell directori. Per a cada subdirectori, s'ha d'imprimir per pantalla el nom del subdirectori així com el nom i la mida (en bytes) dels 3 fitxers més grans que s'hi hagin trobat. Els fitxers s'han imprimir en per ordre de mida (descendent). *Es pot fer servir com a màxim un bucle per implementar l'script.*

Es mostra a continuació un exemple d'execució. Sou lliures que el resultat es mostri en el format que us sigui més fàcil.

```
$ ./exercici1.sh gutenber
Directori:  gutenber
46764 master_list.csv
268840 authors.html
268840 titles.html
Directori:  gutenber/etext00
1766775 1cahe10.txt
1899162 dscmn10.txt
4651897 00ws110.txt
Directori:  gutenber/etext01
1373362 ironm11h.htm
1391054 idiot10.txt
3282452 wrnpc10.txt
<continua>
```

Exercici 2

Fer un script anomenat **exercici2.sh** que té tres arguments: un directori i dues cadenes, *cadena1* i *cadena2*. L'script ha d'analitzar, de forma recursiva, tots els fitxers que hi hagi dintre del directori. Per a cada fitxer trobat ha de comptar quantes paraules contenen les dues cadenes (*cadena1* i *cadena2*) en una mateixa paraula i en l'ordre especificat a la línia d'arguments (primer *cadena1* i després *cadena2*). Així, per exemple, si s'especifiquen com a argument les cadenes “ad” i “en”, la paraula “nonadjacent” compleix les condicions. En canvi, la paraula “tapenade” no les compleix atès que apareixen en ordre invers a l'especificat als arguments. *Es pot fer servir com a màxim un bucle per implementar l'script.*

Es mostra a continuació un exemple d'execució. Sou lliures que el resultat es mostri en el format que us sigui més fàcil.

```
$ ./exercici2.sh gutenber ad en
gutenber/etext00/00ws110.txt 113
gutenber/etext00/1cahe10.txt 75
gutenber/etext00/1vkip11.txt 47
gutenber/etext00/2cahe10.txt 42
gutenber/etext00/2yb4m10.txt 15
gutenber/etext00/8rbaa10.txt 3
<continua>
```

Exercici 3

Fer un script anomenat **exercici3.sh** que té tres arguments: un directori, una extensió i un valor sencer. L'script ha d'analitzar tots els fitxers que hi hagi a dins del directori arrel especificat com a argument. En cas que el fitxer tingui l'extensió i una mida superior a les especificades als argument, l'ha de copiar a un directori temporal. Podeu suposar que el directori **fitxers** està disponible com a directori temporal i que tots els fitxers que s'hi copien tenen noms diferents¹. Un cop s'hi han copiat els fitxers, comprimiu el directori en format **tar.gz** tal i com s'ha realitzat a la pràctica 1. *No es permet utilitzar bucles per implementar aquest script.*

En cas que no s'hagi copiat cap fitxer al directori temporal cal mostrar un missatge d'error per pantalla.

```
$ ./exercici3.sh gutenber txt 100000
No hi ha cap fitxer que compleixi amb les condicions especificades.
```

¹En cas que hi hagi fitxers amb el mateix nom, podeu sobreescrivre l'antic fitxer pel nou, podeu deixar l'antic o adoptar la solució que més us agradi!

En cas que se s'hagi copiat algun fitxer al directori temporal es mostra un missatge informatiu com el següent

```
$ ./exercici3.sh gutenbergtxt 1000
He comprimit els fitxers a fitxers.tar.gz.
```

Exercici 4

Fer un script anomenat `exercici4.sh` que té un únic argument, un nom d'usuari. Aquest script ens donarà informació sobre aquest usuari fent servir la informació que hi ha emmagatzemada als fitxers `/etc/passwd` i `/etc/group`. Aquests dos fitxers són fitxers essencials que contenen informació sobre els usuaris de l'ordinador i els grups als quals pertanyen.² L'script a programar ha d'extreure informació sobre l'usuari fent servir aquests dos fitxers. *Es permet utilitzar com a màxim un bucle per realitzar la implementació d'aquest script.*

En cas que s'especifiqui un usuari que no existeix l'script ha de donar un missatge error

```
$ ./exercici4.sh jordi
Aquest usuari no existeix en aquest ordinador.
```

En cas que l'usuari existeixi, s'ha de mostrar la informació que s'ha extret dels dos fitxers. La informació que s'imprimeix tindrà una estructura similar a aquesta

```
$ ./exercici4.sh lluis
Nom de l'usuari:  Lluís Garrido
Identificador de l'usuari:  1000
Grups als quals pertany l'usuari:  100 (users), 463 (docker), 486 (lp), 26 (postgres)
Directori arrel de l'usuari:  /home/lluis
Intèrpret per defecte:  /bin/bash
```

Per poder fer aquest exercici es proporcionen, en cas que faci falta, els fitxers `passwd` i `group` de l'ordinador d'en Lluís Garrido.

Exercici 5

La comanda `ps` ens permet obtenir informació dels processos que s'executen a l'ordinador. Executant la comanda amb `ps aux` s'obté molta informació sobre els processos que s'hi executen. Dues de les informacions que es mostren són la memòria virtual (VSZ) i memòria física (RSS) que el procés està ocupant en aquell moment.

L'objectiu d'aquest exercici es fer un script, anomenat `exercici5.sh`, que té dos arguments: un nom d'usuari i un d'addicional que és una cadena que només pot ser VSZ o RSS. L'objectiu d'aquest script és que mostri per pantalla els 10 primers processos d'aquell usuari ordenats de forma descendent per VSZ o RSS (dependent del que s'hagi especificat per argument). L'script només haurà de mostrar tres columnes: la VSZ, la RSS i el nom del procés associat. *No es permet utilitzar bucles per implementar aquest script.*

En cas que s'especifiqui un nom d'usuari que no té cap procés executant-se s'ha de mostrar un missatge d'error.

```
./exercici5.sh jordi VSZ
En jordi no te cap procés associat.
```

En cas que el segon argument no sigui ni VSZ ni RSS també s'ha de mostrar un missatge d'error.

```
./exercici5.sh lluis S01
El segon argument ha de ser VSZ o RSS.
```

Us mostrem a continuació un exemple d'execució en què es mostra un d'exemple d'excursió per a l'usuari lluis en què els processos s'ordenen per VSZ. Recordar que només s'han de mostrar per pantalla les columnes VSZ, RSS i el nom del procés.

²Podeu trobar informació de l'estructura d'aquests fitxers en aquests dos enllaços: enllaç per `passwd`, enllaç per `group`

```

$ ./exercici5.sh lluis VSZ
VSZ      RSS      Proces
3899652 208484 /usr/bin/plasmashell
3820300 366220 /usr/lib64/firefox/firefox
3395820 622360 /usr/lib64/firefox/firefox
3354796 227772 /usr/share/spotify/spotify
3289320 126840 kwin_x11
2912624 250128 /usr/lib64/firefox/firefox
2636588 88544 /usr/sbin/mysqld
2585372 87108 /usr/lib64/firefox/firefox
2567068 41976 akonadiserver
2554068 20616 /usr/bin/pulseaudio

```

2 Informe

A més dels scripts caldrà entregar un informe sobre la feina feta. L'informe a entregar ha d'estar en format PDF o equivalent (no s'admeten formats com `odt`, `docx`, ...). Aquest informe ha de mostrar les proves que s'han realitzat per assegurar el bon funcionament del codi. No inclogueu una descripció de l'algorisme implementat llevat que cregueu rellevant comentar algun detall. Inclogueu, preferentment en format text (tal com es fa en aquest document), la comanda que heu executat així com la sortida obtinguda i algun comentari breu descrivint el resultat si ho creieu necessari. En cas que no us funcioni el codi indiqueu també el problema detectat així com possibles sospites de quin pot ser la font del problema. Sigueu breus i clars a les vostres respostes, no fa falta que us esteneu en el text escrit.

En cas que preferiu incloure captures de pantalla en comptes d'incloure el resultat en format text, assegureu-vos que el text de la captura es pot llegir bé (és a dir, que tingui una mida similar a la resta del text del document) i que totes les captures siguin uniformes (és a dir, que totes les captures tinguin la mateixa mida de text).

El document ha de tenir una llargada màxima de 3 pàgines (sense incloure la portada). El document s'avaluarà amb els següents pesos: proves realitzades i comentaris associats, un 60%; escriptura sense faltes d'ortografia i/o expressió, un 20%; paginació del document feta de forma neta i uniforme, un 20%.