

FONAMENTS D'ENGINYERÍA

Pràctica 1

Arnau Marcos Almansa (1354223)

Jairo Villarroel Rodríguez (1571069)

Grup 45

Curs 2019-20

Objectius de la pràctica

L'objectiu d'aquesta pràctica és aprendre i familiaritzar-se amb la programació en *bash*. En concret, familiaritzar-se amb la sintaxi del llenguatge, algunes comandes essencials i el tractament de fitxers i dades.

Plantejament

La pràctica, consisteix en el desenvolupament d'un script en *bash* que permeti que un usuari pugui obtenir dades d'un fitxer *csv*. Aquest fitxer contindrà un llistat de sèries i pel·lícules. L'usuari ha de poder, mitjançant diversos menús, obtenir un registre aleatori del fitxer, aplicar un filtre de puntuació (*rating*) i interactuar amb un sistema de preferències que seran utilitzades a l'hora de seleccionar el registre aleatori.

Per desenvolupar aquest script hem seguit les pautes de l'enunciat. Cada opció del script s'ha programat independentment en funcions separades que es criden des de el bucle principal o des de altres funcions. D'aquesta manera, ens queda un codi modular i reutilitzable.

Descripció funcional del script

En iniciar el *script*, s'agafa tot el contingut del fitxer *netflix.csv* excepte la primera línia, s'eliminen els duplicats i s'introdueix a un nou fitxer *netflix_unique.csv*.

Si es detecta el fitxer de preferències, es generarà un fitxer *filtrat_preferencies.csv* que contindrà les sèries i pel·lícules que compleixin les preferències de l'usuari.

La base del *script* és un bucle amb una estructura de control *case*. Això fa la funció de menú que, segons l'opció que seleccioni l'usuari, crida a una funció o a una altra.

Si es selecciona l'opció 5, el programa finalitza.

Les funcions principals són: *recomanacio_rapida*, *llistar_per_any*, *llistar_per_rating* i *criteris_de_cerca*.

A *recomanacio_rapida*, mostrem per pantalla una sèrie/pel·lícula aleatòria. Si hi ha preferències, *recomanacio_rapida* només utilitzarà les sèries/pel·lícules que coincideixin amb els criteris de cerca.

Si el fitxer amb les sèries filtrades per preferències no té registres, avisarà a l'usuari.

Per escollir un número de registre aleatori, generem un nombre aleatori i calculem el residu d'aquest entre el número de línies del fitxer i li afegim 1. D'aquesta manera ens queda un número entre l'1 i la longitud del fitxer que podem utilitzar per obtenir el registre.

La funció *llistar_per_any* llegeix un any introduït per l'usuari i l'utilitza per mostrar per pantalla un llistat amb totes les series/pel·lícules que hagin sigut publicades aquell any.

Per fer el filtre per any hem utilitzat la comanda *grep*, aquesta comanda ens permet filtrar les línies que compleixin una expressió regular. En aquest cas, busquem els registres que tinguin l'any escollit a l'antepenúltim camp. Això és així per tractar correctament certs registres que inclouen una coma (el separador de camps) al títol o a la descripció. Si l'expressió regular estigues escrita per filtrar per la cinquena columna des de el principi de la línia en comptes de l'antepenúltima, els registres anteriors es filtrarien incorrectament.

La funció *llistar_per_rating* mostra un menú on l'usuari pot escollir un nombre d'estrelles per filtrar o sortir. La funció mostrarà per pantalla les series/pel·lícules que tinguin el mateix nombre d'estrelles escollides per l'usuari o més.

Per fer això, quan l'usuari introdueix un número, s'assigna un *rating* mínim i un *rating* màxim i es filtren les sèries que estiguin entre aquests dos *ratings*. Un cop s'han filtrat les sèries, s'ordenen per *rating* de menor a major i els donen format per mostrar-les. En donar format, es reemplaça el camp del *rating* numèric del 0 al 100 per un *rating* d'estrelles de l'1 al 5. Finalment es mostren per pantalla.

La última funció, *criteris_de_cerca*, obre un menú on l'usuari pot escollir entre modificar les preferències, eliminar-les, o mostrar-les.

Si escull modificar les preferències, haurà de seguir les instruccions i introduir els valors per a cada criteri (*any*, *rating*, *stars*), si no introdueix cap valor a un criteri, aquest serà ignorat.

Si l'usuari escull l'opció d'eliminar les preferències, s'eliminarà el fitxer on s'emmagatzemen els valors de les preferències.

Si escull la tercera opció, mostrarà les preferències actuals, amb el seu nom i el seu valor.

Problemes

Un dels primers problemes que vam tenir, que era part de l'enunciat, va ser que el document *netflix.csv* tenia moltes sèries/pel·lícules repetides. Això ho hem arreglat amb la següent comanda:

```
tail +2 netflix.csv | sort -u > netflix_unique.csv
```

Això treu la capçalera del fitxer amb el *tail +2* i elimina registres repetits amb *sort -u*. Al treure la capçalera, evitem que es barregi amb la resta de registres en eliminar repetits i ens facilita tractar el fitxer a diferents parts del programa, com quan mostrem un registre aleatori, on no ens hem de preocupar que ens mostri la capçalera per accident.

Un altre problema que vam tenir, va ser quan fèiem el filtre a *listar_per_any*. Vam observar que la següent comanda:

```
grep "$any" netflix_unique.csv | cut -d',' -f1,2 | less
```

mostrava registres que tenien el valor *\$any* a camps que no eren el d'any, sinó el títol, la descripció o el rating.

Això ho hem solucionat canviant la comanda anterior per:

```
grep ",$any,[0-9]*,[0-9]*$" netflix_unique.csv | cut -d',' -f1,2 | column -t -s " " | less
```

Fent això ens assegurem que la informació l'agafarà de l'antepenúltima columna del document *netflix_unique.csv*, que és l'any de publicació. Si no féssim això, la cerca es faria per qualsevol camp.

Un problema bastant interessant que ens ha succeït ha sigut que, quan utilitzàvem el valor d'una variable que contenia espais o caràcters separadors sense posar-la entre cometes dobles, aquesta variable es convertia en tantes variables com paraules separades entre espais o altres caràcters.

Per exemple, si la variable *frase* conté «Hello World» i passem *\$frase* (sense cometes dobles) com argument a una funció, «Hello» serà el primer argument i «World» serà el segon. No s'enviaran com un únic argument.

Això ens ha passat a diversos punts del script i ens ha obligat a buscar-ho a internet fins a entendre que, quan una variable conté múltiples paraules separades per espais o altres caràcters, cal expandir aquesta variable entre cometes dobles o es poden produir errors.

Execució

L'*script* és interpretat, i per executar-lo l'hem de cridar en un terminal *bash*. Abans d'executar-lo cal donar-li permisos d'execució si no els té amb la comanda *chmod*.

Conclusions

La pràctica ha estat molt didàctica. Ha estat prou guiada per a no perdre el fil i prou lliure perquè haguem necessitat buscar informació a internet.

La part de programació ha sigut curta, ja que una part important del temps l'hem invertit investigant la sintaxi de *bash*, comandes i solucionant problemes que ens han sorgit per falta de coneixement sobre aquest llenguatge de *scripting*.

Després d'haver acabat aquesta pràctica, creiem que no hem adquirit un coneixement massa exhaustiu sobre *bash*, però si el suficient per a poder dur a terme projectes més complexos en un futur. També hem après com buscar informació a internet en cas de tenir un dubte o problema i creiem que això és una part essencial per a poder continuar aprenent de forma independent i per a projectes futurs inclús d'altres assignatures.