

**7.1 - bash****a) - resultado de ejecución** <sup>1</sup>

Usamos los comandos de unix para realizar un trabajo. Y cuando todo va bien puede que no escriban. Al acabar *responden* a la pregunta: - ¿Qué tal? . Y la filosofía es:

*Si no hay noticias, son buenas noticias.*                      *No news, good news.*

Los comandos *devuelven* (responden) **0** indicando **bien**, o *resultado frecuente*. En otros casos devuelven un valor entero entre 1 y 255 .

Cómo no, lo de ‘resultado frecuente’ o ‘bien’, en algunos casos es opinable.

`$?` devuelve el resultado del último comando ejecutado *síncronamente*.

Vamos a verlo: izq., der., izq., der. ...

```
cd ; cd tso/01/mio
rm -f ? ??
cp a b ; echo $? # 20
ls ; echo $?
ls ? ; echo $?
echo hola > a ; echo $?
cp a b ; echo $?
rm a ; echo $?
rm c ; echo $?
cd ../palabras
grep es dias ; echo $? # 90
grep fiesta dias ; echo $? # 100
grep '[xyz]' dias ; echo $? # 110
cd dir/dir/dir ; echo $?
cd ../progs
rm -f fi09
make fi09 ; echo $?
vi fi09.c ; make fi09 ; echo $?
cal 1000 ; echo $? # 150
```

**b) - if, escript** <sup>2</sup>

El resultado **0** se utiliza como **cierto** / **true** .

```
cd ; cd tso/01/mio
echo pasaaa > b # 200
echo ayer > f
if cp a f ; then echo " copiado a" ; else cp b f ; echo " copiado b" ; fi
ls
cat f # 240
```

No es frecuente escribir un `if` en la línea de comandos. Copia `esc07b1_e` y edita ... .

```
rm ?
cp ../progs/esc07b1_e esc07b1
vi esc07b1
bash esc07b1
```

<sup>1</sup>apuntes SSAA, cap. 21, pag. 233,234

<sup>2</sup>apuntes SSAA, cap. 21, pag. 236,237

**7.2 - sed** <sup>3</sup>

(continúa)

e) ... acción/verbo: **substituir**, indicado por `s` .

g) al final de las líneas 210 y siguientes, indica 'todas las apariciones en cada línea seleccionada'.

```
cd ; cd tso/01/palabras
sed -e 's/e/E/g' dias # 310
sed -e 'r/s/o/00/g' meses
sed -e 'r/!s/o/00/g' meses # 330
```

Vemos que la *parte dónde* indica cuando se aplica la substitución.f) **sed y expresiones regulares.** <sup>4</sup>Si las expresiones regulares amplían las posibilidades de `sed` , el apartado de ... *marcadas* hace posible substituciones sofisticadas.

```
cd ; cd tso/01/palabras
sed -e 's/[ao]/-/ ' meses # 410
sed -e 's/[ao]/-/g ' meses # 420
sed -e '/[ao]/!s/e/E/g ' meses # 430
sed -e 's././ ' dias # 440
sed -e 's/(.\\)(.*\\)/\2\1/' dias # 450
sed -e 's/(...\\)(.*\\)/\2\1/' dias # 460
sed -e 's/(.*\\)(..\\)/\2\1/' dias # 470
sed -e '2,6s/(.*\\)(..\\)/\2\1/' dias # 480
```

g) Otras acciones: `p` escribe (*print*) otra vez, `r` lee (*read*) un fichero, `a` añade un texto

```
sed -e '5,7p' dias
echo " falta menos" > falta
echo " ya falta menos" >> falta
cat falta
sed -e '3,5r falta' dias
sed -e '3,5a falta' dias
sed -e '3,5a \ \ falta' dias
rm falta
```

**7.3 - filtros**d) `tr` cambia unos caracteres por otros. <sup>5</sup>Opciones: `d` borra, `c` complemento (de param 1<sup>o</sup>), `s` elimina repetición de cambios

```
cd ; cd tso/01/palabras
cat dias | tr aeiou AEIOU # 610
tr aeiou AEIOU < dias ; tr uoiea UOIEA < dias
tr aeiou eioua < dias
tr aeiou aaaaa < dias ; tr aeiou aei < dias
tr a-t A-T < dias
tr '\012' = < dias # 660
tr -c 'a-t\012' - < dias
tr -s aeiou aaaaa < dias ; tr -s aeiou a < dias
tr -d eiou < dias
```

\012 representa un solo carácter, el CAMBIO DE LÍNEA. Las comillas se ponen para que `bash` no intervenga en lo que escribimos.<sup>3</sup>apuntes SSAA, cap. 12, pag. 81-86<sup>4</sup>apuntes SSAA, cap. 13, pag. 103-108<sup>5</sup>apuntes SSAA, cap. 12, pag. 80