

# Laboratorio di Elementi di Architetture e Sistemi Operativi

Soluzioni degli esercizi del 21 Marzo 2012

## Esercizio 1.

1. Scrivere un comando che fornisca i primi 15 eventi contenuti nella storia dei comandi  

```
$ history | head -n 15
```
2. Scrivere un comando che fornisca il numero di eventi contenuti nella storia dei comandi  

```
$ history | wc -l
```

**Esercizio 2.** Creare una sottodirectory `bin` all'interno della propria home directory in cui mettere gli script. Fare in modo che gli script contenuti in `bin` possano essere invocati da qualunque directory con il nome del file, senza dover specificare l'intero percorso.

```
$ cd  
$ mkdir bin  
$ export PATH=$PATH:$HOME/bin
```

L'ultimo comando aggiunge al path la directory `$HOME:/bin` alla fine del valore corrente. Il comando `export PATH=$HOME/bin` non è corretto, perché elimina da `PATH` tutti i percorsi dove si trovano i comandi di sistema (come `ls`, `cp`, ecc.), rendendoli inutilizzabili.

Alla chiusura della shell il `PATH` ritorna al valore di sistema. Se si vuole rendere il cambiamento permanente si deve aggiungere `export PATH=$PATH:$HOME/bin` alla fine del file `.profile` presente nella propria home.

**Esercizio 3.** Scrivere uno script che elenchi le seguenti informazioni:

1. il nome dell'utente e la sua home
2. la directory corrente
3. la data e l'ora corrente

```
#!/bin/bash  
echo "Il nome dell'utente e' :" $USER  
echo "La sua home:" $HOME  
echo "La directory corrente:" $PWD  
echo "La data e l'ora esatta e' ": $(date)
```

**Esercizio 4.** Scrivere uno script che accetti come argomento una lista di file e restituisca il numero totale di parole presenti nei file della lista.

```
#!/bin/bash  
cat $* | wc -w
```

**Esercizio 5.** Scrivere uno script che estragga soltanto le linee con un commento dal file java fornito come primo argomento (ossia le linee che contengono `//`). I commenti estratti devono essere salvati nel file fornito come secondo argomento.

```
#!/bin/bash  
grep "//" $1 > $2
```

**Esercizio 6.** *Creare uno script chiamato `creascript.sh` per la creazione automatica di script. Il nome dello script da creare è il primo argomento di `creascript.sh` a cui viene aggiunta l'estensione `.sh`. Lo script creato deve avere `#!/bin/bash` come prima linea ed il permesso di esecuzione attivo per tutti gli utenti. Lettura e modifica consentiti solo al proprietario.*

```
#!/bin/bash
nomefile=$1.sh
echo "#!/bin/bash" > $nomefile
chmod 711 $nomefile
```

**Esercizio 7.** *Creare uno script che prende come argomento il nome di una directory e cancella tutti i file con nome `core` dalla directory argomento e ricorsivamente in tutte le sue sottodirectory.*

```
#!/bin/bash
dir=$1
listafile=$(find $dir -type f -name core 2>/dev/null)
rm $listafile
```

**Esercizio 8.** *Creare uno script che accetti due parametri: estensione e stringa. Lo script deve elencare tutti i file presenti ricorsivamente nella cartella corrente la cui estensione è uguale al primo argomento e che contengono almeno un'occorrenza della stringa passata come secondo argomento.*

```
#!/bin/bash
ext=$1
string=$2
listafile=$(find . -type f -name "*.$ext 2>/dev/null)
grep -l $string $listafile
```