

# Laboratorio di Elementi di Architetture e Sistemi Operativi

Soluzioni degli esercizi del 7 Marzo 2012

Nelle soluzioni le righe che iniziano con \$ rappresentano i comandi da dare al prompt. Le altre i risultati dei comandi.

**Esercizio 1.** Esplorate il vostro file system:

1. \$ pwd  
/home/bresolin
2. \$ cd /
3. \$ pwd  
/
4. \$ cd  
\$ pwd  
/home/bresolin
5. Il percorso della home directory è quello visualizzato dal comando pwd al punto precedente. In questo caso /home/bresolin
6. \$ cd /
7. \$ cd /home/bresolin  
\$ pwd  
/home/bresolin
8. \$ cd /
9. \$ cd home/bresolin  
\$ pwd  
/home/bresolin

**Esercizio 2.** \$ ls -t

**Esercizio 3.** & ls -R

**Esercizio 4.**

1. con un comando unico: \$ ls /usr/bin/lo\*  
con due comandi: \$ cd /usr/bin ; ls lo\*
2. con un comando unico: \$ ls /usr/include/???.h  
con due comandi: \$ cd /usr/include ; ls ???h

**Esercizio 5.**

1. \$ mkdir prova
2. \$ cp /usr/share/vim/vimrc prova oppure  
\$ cd prova ; cp /usr/share/vim/vimrc .
3. \$ cp /usr/share/vim/vimrc prova/filecopia  
oppure (se ci si è spostati in prova al passo precedente)  
\$ cp /usr/share/vim/vimrc fileprova
4. \$ cat vimrc oppure \$ cat prova/vimrc

**Soluzioni alternative:**

usare un edito di testo: \$ gedit vimrc

usare il comando more o less: \$ more vimrc

5. per svolgere questo punto è necessario prima creare la directory prova2 (da qui in poi si assume di essere nella home):  
\$ mkdir prova2  
quindi si possono copiare i file:  
\$ cp prova/\* prova2

6. \$ rm prova/\*
7. \$ rmdir prova
8. \$ rm -r prova2

### Esercizio 6.

1. \$ mkdir prova
2. \$ cp /usr/include/stdio.h prova oppure  
\$ cd prova ; cp /usr/include/stdio.h .
3. \$ cd prova (se non lo si è fatto al passo precedente)  
\$ ln stdio.h stdio-hard.h
4. \$ ln -s stdio.h stdio-link.h
5. \$ cat stdio-hard.h  
\$ cat stdio-link.h

#### Soluzioni alternative:

usare un edito di testo: \$ gedit stdio-hard.h

usare il comando more o less: \$ more stdio-link.h

6. \$ ls -l  
total 88  
-r--r--r-- 2 davide davide 16756 8 Mar 11:11 stdio-hard.h  
lrwxr-xr-x 1 davide davide 7 8 Mar 11:12 stdio-link.h -> stdio.h  
-r--r--r-- 2 davide davide 16756 8 Mar 11:11 stdio.h
7. \$ rm stdio-\* oppure \$ rm stdio-hard.h stdio-link.h
8. \$ cd .. ; rm -r prova

### Esercizio 7. Alcuni modi per creare un file:

1. creare il file con un editor di testo come gedit:  
\$ gedit fileprova.txt
2. creare una copia di un file esistente:  
\$ cp fileprova.txt fileprova2.txt
3. usare il comando touch, che crea un file vuoto se il nome del file dato come argomento non esiste:  
\$ touch fileprova3.txt
4. usare il comando cat, usando la redirectione dell'output:  
\$ cat > fileprova4.txt (premere ctrl-D per terminare l'inserimento del file)

### Esercizio 8.

1. creare un file prova.txt contenente almeno 15-20 righe di testo
2. mostrare la differenza nei risultati dei comandi:  
\$ cat prova.txt mostra tutto il file  
\$ head prova.txt mostra solo le prime righe  
\$ tail prova.txt mostra solo le ultime righe

### Esercizio 9. Sì, un link simbolico può puntare ad un altro link che a sua volta punta ad un file:

```
$ ln -s file_name link1
```

```
$ ln -s link1 link2
```

eseguendo il comando cat link2 viene stampato a video il contenuto del file file\_name.

Il limite al numero di link simbolici in catena dipende dal sistema operativo. Nel caso dei computer del laboratorio è 8:

```
$ ln -s file_name link1
```

```
$ ln -s link1 link2
```

```
$ ln -s link2 link3
$ ln -s link3 link4
$ ln -s link4 link5
$ ln -s link5 link6
$ ln -s link6 link7
$ ln -s link7 link8
$ ln -s link8 link9
```

Eseguendo il comando `cat link8` viene visualizzato il contenuto di `file_name`, mentre eseguendo il comando `cat link9` si ottiene il messaggio d'errore

```
cat : link9: Too many symbolic links encountered.
```