



Primi passi con il sistema operativo Linux

Quando l'utente entra nel sistema gli viene assegnata un'area di disco dedicata ai propri file che può essere divisa in *directory*. Una *directory* rappresenta una divisione logica dell'area di disco. Le *directory* hanno un nome che segue le stesse convenzioni dei nomi dei file.

Una *directory* può contenere altre *directory* in un albero gerarchico. L'albero delle *directory* si può navigare separando i nomi delle *directory* con uno *slash*.

Per eseguire le operazioni l'utente dispone di comandi da invocare, attraverso la tastiera, in un terminale (che appare come una finestra) attraverso la *shell*: un ambiente nel quale i comandi sono riconosciuti dal sistema operativo ed eseguiti.

Per creare una *directory*, nella *shell* si usa il comando

```
mkdir my_dir
```

dove *my_dir* è il nome che s'intende dare a una *directory*. Per entrare in una *directory* si usa il comando

```
cd my_dir
```

In questo modo si può creare una nuova *directory* all'interno di *my_dir*, ad esempio con

```
mkdir sub_dir
```

Per tornare nella *directory* principale si può dare il comando

```
cd ~
```

Il simbolo *~* infatti rappresenta la propria *home directory*. Per tornare nella *directory* superiore invece si può dare il comando

```
cd ..
```

I due punti rappresentano la directory che si trova al livello superiore di quella attuale. Il punto singolo invece rappresenta la directory corrente. Se dalla home directory si dà il comando

```
cd my_dir/sub_dir
```

si scende prima in `my_dir` e poi in `sub_dir`. Da qui si può risalire in `my_dir` con il comando

```
cd ..
```

e nella home directory con

```
cd ../..
```

Per creare un file di testo si può usare il comando `emacs`. Dopo aver salvato il file così creato, per vedere i file su disco si usa il comando

```
ls
```

(list) oppure, per avere più dettagli su ciascun file

```
ls -l
```

Se al comando `ls` si passa il nome di un file o di una directory il comando mostra solo il file o la directory indicati. Il nome si può esprimere usando la *wildcard* `*`: un carattere speciale che corrisponde a tutti i possibili caratteri. Per esempio

```
ls -l a*
```

mostra, con tutti i dettagli, la lista dei file il cui nome inizia per `a`.

In generale tutti i comandi Unix si compongono di una parola chiave (`ls`) cui seguono zero o più parametri (`a*`) e/o dei modificatori (`-l`) che ne alterano il comportamento di default. Per conoscere tutti i possibili usi di un comando si può usare il comando `man` come in

```
man ls
```

La copia di un file si esegue con il comando `cp` (copy) che accetta due parametri: il nome del file da copiare e il nome che assumerà la copia. Il comando

```
cp a.b c.d
```

assume che il file `a.b` esista e crea un file `c.d` il cui contenuto è identico a quello di `a.b`. Il comando

```
cp a.b ../c.d
```

esegue le stesse operazioni, ma il file di destinazione è creato nella directory superiore rispetto a quella in cui si trova il file `a.b`.

Il nome di un file si cambia con il comando `mv` (move):

```
mv a.b c.d
```

che cambia il nome del file da `a.b` a `c.d` senza modificarne il contenuto. Il comando `mv` si può usare anche per spostare il file da una directory all'altra. Ad esempio:

```
mv a.b sub_dir/c.d
```

sposta il file `a.b` dalla directory corrente a quella di nome `sub_dir`, presente nella directory corrente.

Il comando `rm` (remove) elimina il file passato come argomento come in

```
rm a.b
```

Il modificatore `-f` impone al comando di rimuovere il file senza chiedere conferma all'utente. Il modificatore `-r` si usa per rimuovere una directory (che deve essere vuota).

In generale ogni file è univocamente individuato sul disco specificandone il *percorso* o *path* completo per giungere nella directory nella quale si trova partendo dalla directory gerarchicamente più importante detta *root* e indicata col simbolo `/`. Le aree di disco degli utenti in genere si trovano all'interno di una directory con il nome dell'utente che, a sua volta, si trova in una directory dedicata agli utenti. In laboratorio il percorso completo del file `a.b` che si trova nella directory `my_dir` della home directory dell'utente `lcgo23` è

```
/home/studenti/lcgo23/my_dir/a.b
```

Le home directory degli utenti si trovano infatti sotto la directory `/home/studenti` e prendono il nome dello user cui appartengono.

Quindi, per ottenere i dettagli di questo file stando nella propria home directory l'utente `lcgo23` deve dare il comando

```
ls -l my_dir/a.b
```

oppure

```
ls -l /home/studenti/lcgo23/my_dir/a.b
```

o ancora

```
ls -l ~/my_dir/a.b
```

Negli ultimi due casi il comando funziona indipendentemente dalla directory nella quale ci si trova. Nel primo caso funziona solo se l'utente si trova, nel momento in cui invoca il comando, nella propria home directory.