NOTE PER UTILIZZO COMPILATORE C CON LINUX

Queste pagine sono estratte dalle note del corso "Abilità Informatiche: Introduzione a Unix", *Alessandra Seghini*

Per stampare questo documento si consiglia di utilizzare la versione in formato PDF.

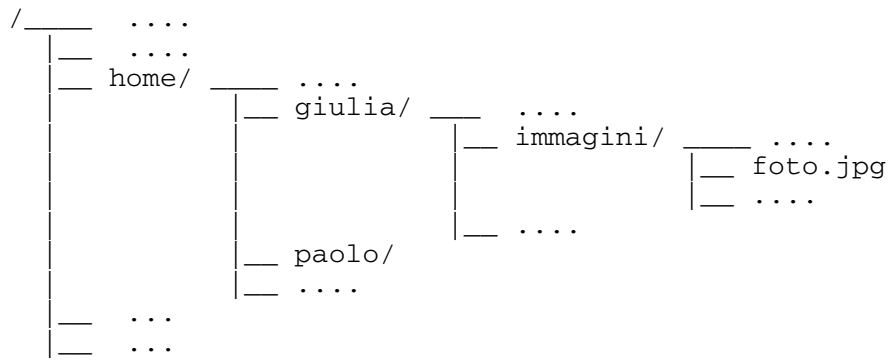
Alcuni termini da conoscere

- **Filesystem**
  Su un sistema UNIX i **file** sono suddivisi in **directory** (cartelle). Le directory sono organizzate gerarchicamente in una struttura ad albero che prende il nome di **filesystem**. La directory radice è indicata con il simbolo / e prende il nome di `root directory`.

- **Home directory**
  Al login ogni utente si trova posizionato sulla propria *home directory*.
  È possibile riferirsi alla home directory di un utente con "`~nome_utente`", ad esempio `~studente` può essere utilizzato per indicare la directory `/home.hd/studente/`.
  Ogni utente può riferirsi alla propria home directory semplicemente con il carattere `~`, senza specificare lo username.

- **Directory corrente**
  La *directory corrente* di lavoro è quella sulla quale si è posizionati. È possibile conoscere il nome della directory corrente, utilizzando il comando `pwd`. Al login, la directory corrente di un utente è la sua home directory.
  La directory corrente è indicata con il simbolo "."
  La directory "genitore" della directory corrente è indicata con il simbolo ".."

- **Pathname**
  Per individuare un file sul sistema, è necessario conoscere il suo *pathname*, costituito dal nome del file, preceduto dal nome della directory che lo contiene.



  Ad esempio, il nome completo del file `foto.jpg` contenuto nella directory `/home/giulia/immagini/` è `/home/giulia/immagini/foto.jpg`. Abbiamo così specificato il *percorso assoluto*, che permette di raggiungere il file `foto.jpg` a partire dalla radice (/) del filesystem e che, pertanto, è valido qualunque sia la directory corrente.
  È possibile individuare un file anche specificando il *percorso relativo*, rispetto alla directory sulla quale ci si trova posizionati, ovvero rispetto alla directory corrente. Se ad esempio siamo sulla directory `/home/giulia/`, possiamo riferirci al file `/home/giulia/immagini/foto.jpg` semplicemente indicando `immagini/foto.jpg` (senza il carattere / iniziale). In questo caso, il percorso per raggiungere il file `foto.jpg` non parte dalla radice del filesystem, ma dalla directory corrente e, pertanto, è valido solo se la directory corrente è `/home/giulia/`.
  Nello specificare il pathname di un file, si possono utilizzare i caratteri . , .. e ~ per indicare rispettivamente la directory corrente, la directory di livello superiore rispetto alla directory corrente e la home directory di un utente.
  Ad esempio, un percorso assoluto per individuare il nostro solito file `foto.jpg`, è anche

`~giulia/immagini/foto.jpg`.
Se la directory corrente è la home directory dell'utente paolo `/home/paolo/` (o `~paolo`), il percorso relativo per individuare il file foto.jpg sarà `../giulia/immagini/foto.jpg`.

Alcuni comandi utili

- **`ls [nome-dir]`**
  riporta l'elenco dei file contenuti nella directory `nome-dir`. Se non è specificato il nome di una directory, riporta l'elenco dei file contenuti nella directory corrente.

```
archimede%ls
Mail            bookmarks.html  help            help-alex       lq
```

  Qualche opzione del comando `ls` (è possibile combinare insieme più opzioni):

  `ls -a [nome-dir]`
  riporta l'elenco di tutti (*all*) i file contenuti nella directory `nome-dir`, compresi i file *nascosti* il cui nome inizia con il carattere ".". Nell'elenco sono compresi anche i file . e .. , che indicano rispettivamente, la directory corrente e la directory "genitore" della directory corrente.

```
archimede%ls -a
.               .Mathematica    .history        bookmarks.html  help-alex
..              .cshrc          Mail            help            lq
```

  `ls -l [nome-dir]`
  riporta l'elenco dei file contenuti nella directory `nome-dir`, in formato esteso (*long*).

```
archimede%ls -la
total 84
drwxr-xr-x   5 giulia  users        512 Jan 26 08:52 .
drwxr-xr-x  31 root    root        2560 Jan 26 09:11 ..
drwxr-xr-x   8 giulia  users        512 Jun  7  2004 .Mathematica
-rw-r--r--   1 giulia  users       1487 Nov 29 08:52 .cshrc
-rw-------   1 giulia  users       2607 Jan 25 15:23 .history
drwx------   2 giulia  users        512 Apr 13  2000 Mail
-rw-------   1 giulia  users      28244 Jul 17  2002 bookmarks.html
drwxr-xr-x   2 giulia  users        512 Jan 12  2004 help
lrwxrwxrwx   1 giulia  users         15 Jan 26 08:52 help-alex -> /home/alex/help
-rwxr-xr-x   1 giulia  users        280 Jun 15  2004 lq
```

- **`mkdir nome-dir`**
  crea una nuova directory

- **`rmdir nome-dir`**
  rimuove la directory specificata (deve essere vuota)

- **`cd [nome-dir]`**
  cambia directory, spostandosi nella directory indicata. Se `nome-dir` non è specificato, si sposta nella home directory dell'utente

- **`cp file1 file2`**
  copia il file `file1` sul file `file2`. Se `file2` non esiste, crea un nuovo file, altrimenti lo sovrascrive.
  Nell'indicare `file1` e `file2` è necessario specificare, ove necessario, il *pathname* completo.

- **`mv file1 file2`**
  sposta il file `file1` sul file `file2`. Se `file2` non esiste, crea un nuovo file, altrimenti lo sovrascrive. Questo comando può essere utilizzato per cambiare il nome di un file o per spostarlo in un'altra directory.
  Nell'indicare `file1` e `file2` è necessario specificare, ove necessario, il *pathname* completo.

- **`rm file1 [file2 file3 ...]`**
  elimina il/i file indicati.

- **`more nome-file`**
  visualizza sul terminale il contenuto di un file di testo, mostrando una schermata alla volta. Per scorrere all'interno del documento utilizzare i tasti:

[INVIO]	– avanza di una riga
barra spaziatrice	– avanza di una schermata
b	– torna indietro di una schermata
q	– esce

Stampa di un file

Il comando **`"lpr nome-file"`** invia il file `nome-file` alla stampante *principale* collegata al sistema.
Se il file da stampare è un file in formato testo, come ad esempio il sorgente di un programma C o fortran, potrà essere necessario trasformare il file in formato *postscript*, prima di mandarlo in stampa. Ad esempio, per stampare il file `mioprog.c`, si potrà utilizzare la sequenza di comandi:

```
a2ps mioprog.c -o mioprog.ps
lpr mioprog.ps
```

Il comando `a2ps` trasforma il file testo ricevuto in input in formato *postscript*. Se non vengono specificate opzioni, i fogli del file di output saranno orientati in senzo orizzontale e su ogni foglio saranno stampate due pagine. Se si desidera che il file di output abbia i fogli orientati in senso verticale e che su ogni foglio venga stampata una sola pagina, si dovrà utilizzare il comando:

```
a2ps mioprog.c -o mioprog.ps --chars-per-line=80 --columns=1 --portrait
```

Prima di inviare in stampa il file `mioprog.ps`, è possibile visualizzarlo a schermo utilizzando il comando:

```
gv mioprog.ps &
```

Cenni sull'utilizzo dell'editor emacs

Per editare un file con emacs, inviare il comando:

```
emacs nome-file &
```

Il file *nome-file* sarà caricato nella finestra di *editing*.

Per uscire da emacs, selezionare "Exit Emacs" alla voce "Files" del *pop menù*, oppure digitare la sequenza di tasti <CTRL>x <CTRL>c. Se sono state apportate modifiche al file aperto, emacs chiederà la conferma, prima di salvare il file.

Per salvare le modifiche, senza uscire dall'editor, selezionare "Save Buffer" alla voce "Files" del *pop menù*, oppure digitare la sequenza di tasti <CTRL>x <CTRL>s.

`emacs` è un editor di facile utilizzo. Tutto funziona in maniera molto semplice. Il testo viene inserito automaticamente nel punto in cui si trova posizionato il cursore. È possibile spostarsi nel testo utilizzando sia il mouse sia la tastiera. Dai *pop menù* sono disponibili la maggior parte dei comandi tipici dei programmi di editing (copy, cut, paste, search, etc.)
Per maggiori informazioni si rimanda all'help in linea, richiamabile dal *pop menù* di emacs alla voce "Help".

Editare, compilare ed eseguire un programma C

La creazione di un programma scritto in un **linguaggio ad alto livello**, come il C ed il fortran, prevede diverse fasi per arrivare ad ottenere un codice eseguibile:

- Creazione del **codice sorgente**, ovvero scrittura, tramite un editor, del/dei files contenenti le istruzioni che il programma dovrà eseguire.
- Compilazione del codice sorgente. In questa fase, il compilatore individua e segnala gli eventuali errori di sintassi. È allora necessario utilizzare l'editor per correggere gli errori, salvare il file corretto e quindi ricompilare il programma. Una volta eliminati tutti gli errori, il compilatore traduce il codice sorgente in **codice oggetto**.
- Linking (collegamento) di tutti i file oggetto e delle librerie che concorrono a formare il **programma eseguibile**.

A questo punto, sarà possibile eseguire il programma.

Vediamo un esempio.

1. Utilizzando un editor creare il file `gr2rad.c++`, contenente le seguenti righe di codice sorgente:

```
/* Questo programma converte la misura di un angolo da gradi e radianti */
#include <stdio.h>
int main ()
{
   float Pi =  3.14159;
   float gradi, radianti;
   printf("Misura dell'angolo in gradi?  ");
   scanf("%f", &gradi);
   radianti = Pi*gradi/180.;
   printf("%f gradi = %f radianti.\n", gradi, radianti);
}
```

   Può essere conveniente aprire la finestra di editing in background, utilizzando, ad esempio, il comando

```
emacs gr2rad.c++ &
```

   In questo modo, la shell resterà attiva e sullo schermo saranno disponibili due finestre: la finestra di editing, per correggere gli errori e salvare il file modificato e la shell per compilare.

2. Per compilare utiliziamo *GNU project C and C++ Compiler*, richiamabile con il comando:

```
g++ gr2rad.c++
```

   Verranno segnalati eventuali errori, che dovranno essere corretti.
   `gcc` eseguirà quindi le operazioni di compilazione e di linking e salverà il programma eseguibile nel file `a.out`.
   È possibile assegnare un nome diverso da `a.out` al programma eseguibile, utilizzando l'opzione `-o` (*output*) nel comando di compilazione:

```
g++ gr2rad.c++ -o  gr2rad
```

   In questo caso, il programma eseguibile verrà salvato nel file `gr2rad`.

3. Per eseguire il programma, digitare semplicemente il nome del file:

```
gr2rad
```

Riepilogo delle operazioni da effettuare per l'esercitazione in laboratorio

L'esercitazione si svolge nel Laboratorio didattico di Calcolo, utilizzando il *GNU project C and C++ Compiler (gcc, g++)* in ambiente Linux.

OPERAZIONI INZIALI

1. Accendere il computer, attivare il s.o. linux e collegarsi al sistema.
   Aprire una *shell*, ovvero una finestra nella quale è attivo l'interprete dei comandi (*shell*) che consente di interagire con il sistema operativo.

2. Creare sull'hard disk una directory di lavoro e posizionarcisi:

   ```
   mkdir miadir
   cd miadir
   ```

3. Se necessario, *montare* la pendrive usb e copiare il sorgente del programma sul quale si deve lavorare sulla directory di lavoro *miadir*:

   ```
   cp /media/nome_usb_device/mioprog.c++  .
   ```

ESERCITAZIONE

4. Editare, compilare ed eseguire il programma:

`emacs mioprog.c++ &`	per editare
`g++ mioprog.c++ -o mioprog`	per compilare
`mioprog`	per eseguire

   Attenzione: ogni volta che si modifica il sorgente con l'editor, è necessario salvare il file modificato e poi ricompilare.

   Per stampare utilizzare l'apposita voce dal menù di emacs, oppure i comandi:

   ```
   a2ps mioprog.c++ -o mioprog.ps
   lpr mioprog.ps
   ```

OPERAZIONI FINALI

5. Se necessario, copiare il sorgente del programma sul quale si è lavorato, dalla directory di lavoro alla pendrive usb:

   ```
   cp mioprog.c++ /media/nome_usb_device
   ```

   Per salvare il file sulla pendrive usb, assegnandogli un nuovo nome:

   ```
   cp mioprog.c++ /media/nome_usb_device/newprog.c++
   ```

   Per copiare tutti i file con estensione .c++ dalla directory di lavoro alla pendrive usb:

   ```
   cp *.c++ /media/nome_usb_device
   ```

6. Prima di estrarre la pendrive usb *smontarla dal filesystem*: con il tasto destro del mouse cliccare sull'icona della pendrive che si trova sul desktop e selezionare la voce *Eject*.

7. Spegnere il computer utilizzando l'apposito bottone che appare sulla barra nella parte bassa del monitor.