

I'Editor vi

- Negli ambienti Unix esistono molti editor di testo diversi; tuttavia vi è l'unico che siamo sicuri di trovare in qualsiasi variante di Unix.
- vi (**visual editor**) è stato scritto per essere utilizzabile con qualsiasi tipo di terminale.
- \$ vi filename
invoca vi aprendo il file `filename` (se non esiste, viene creato).
- vi ha tre **modalità**:
 1. **edit** mode (all'avvio di vi si è in questa modalità),
 2. **insert** mode,
 3. **command** mode.

Edit mode (I)

La modalità di edit è usata principalmente per muovere il cursore nel punto di interesse all'interno del file di testo che si sta editando.

Comando	Effetto
k, j, h, l (od i tasti cursore)	muove il cursore su, giù, a sinistra ed a destra
Ctrl-f, Ctrl-b	muove il cursore avanti/indietro di una pagina
H, M, L	muove il cursore alla prima riga, all'ultima od a quella nel mezzo dello schermo
w	muove il cursore all'inizio della parola successiva
e	muove il cursore alla fine della parola successiva
b	muove il cursore all'inizio della parola precedente
0	muove il cursore all'inizio della linea corrente
^	muove il cursore nella posizione del primo carattere della linea che non sia un whitespace
\$	muove il cursore alla fine della linea corrente
/string	cerca nel file la stringa <i>string</i>
?string	cerca “all’indietro” nel file la stringa <i>string</i>
n	cerca l’occorrenza della stringa successiva (in avanti o all’indietro)

Edit mode (II)

Comando	Effetto
<i>nrc</i>	rimancia <i>n</i> caratteri con <i>c</i> a partire dalla posizione del cursore
<i>nx</i>	cancella <i>n</i> caratteri dalla posizione del cursore
<i>ndw</i>	cancella <i>n</i> parole dalla posizione del cursore
<i>ndb</i>	cancella <i>n</i> parole prima del cursore
<i>ndd</i>	cancella <i>n</i> linee a partire da quella del cursore
<i>d\$</i>	cancella tutti i caratteri dalla posizione del cursore fino alla fine della linea
<i>d0</i>	cancella tutti i caratteri dalla posizione del cursore fino all'inizio della linea
<i>J</i>	unisce la linea corrente alla successiva
<i>p</i>	incolla il testo copiato/cancellato a destra del cursore
<i>P</i>	incolla il testo copiato/cancellato a sinistra del cursore
<i>yy</i>	copia la riga corrente in memoria
<i>nyy</i>	copia <i>n</i> righe in memoria a partire dalla posizione del cursore
<i>u</i>	annulla l'ultimo comando
.	ripete l'ultimo comando
<i>ZZ</i>	termina l'esecuzione di vi, salvando le modifiche

Insert mode

Siccome l'edit mode utilizza un gran numero di tasti alfanumerici, per inserire del testo in un file si rende necessaria un'altra modalità: l'**insert** mode.

Comando	Effetto
i	inserisce del testo alla sinistra del cursore
a	inserisce del testo alla destra del cursore
I	inserisce del testo all'inizio della linea corrente
A	inserisce del testo alla fine della linea corrente
o	inserisce una nuova linea sotto la posizione del cursore
O	inserisce una nuova linea sopra la posizione del cursore

Per uscire dalla modalità di inserimento basta premere il tasto Esc o Ctrl-[nei terminali senza tasto Esc.

Command mode

Tutti i comandi del **command** mode iniziano con i due punti (:); dopo aver inserito tale carattere il cursore si sposta nell'ultima riga dello schermo dove compaiono i caratteri del comando successivamente digitati. La pressione del tasto invio provoca l'esecuzione del comando.

Comando	Effetto
:q	termina vi se non vi sono delle modifiche non salvate
:q!	termina vi perdendo le eventuali modifiche non salvate
:w	salva il file originale
:wq	salva il file originale e termina vi (stesso effetto di zz)
:w <i>file</i>	salva il contenuto nel file <i>file</i>
:r <i>file</i>	legge il contenuto del file <i>file</i> inserendolo dopo la posizione del cursore
:e <i>file</i>	edita il file <i>file</i> , sostituendo il contenuto corrente
:f <i>file</i>	cambia il nome del contenuto corrente in <i>file</i>
:f	stampa il nome e lo stato del testo corrente
:n	sposta il cursore alla linea <i>n</i>
:/str/	sposta il cursore alla prossima linea contenente <i>str</i>
:s/ <i>str1</i> / <i>str2</i> /	sostituisce la prima occorrenza di <i>str1</i> sulla linea con <i>str2</i>
:set <i>option</i>	definisce un'opzione, e.g. :set number aggiunge i numeri di linea

L'editor Emacs/XEmacs

Emacs, E(ditor) Mac(ro)s, è un applicativo che non fa parte di Unix; è stato scritto da R. Stallman nel 1975. Digitando al prompt

```
$ emacs <file>
```

si apre una finestra in cui viene visualizzato il contenuto del file <file>, che può essere editato normalmente. Tale contenuto è memorizzato in un buffer (struttura dati interna di Emacs).

Ci sono due modi per accedere ai comandi di editing:

1. barra dei menu,
2. **caratteri di controllo**.

Emacs: caratteri di controllo

Ci sono due categorie di caratteri di controllo:

1. sequenze che iniziano con **Ctrl** (abbrev. **C**), e.g., **C-h t** (esegue il tutorial)
2. sequenze che iniziano con **meta** (tasto **Alt** o **Esc**, abbrev. **M**), e.g., il prefisso **M-x**.

Esistono dei **prefissi** standard di Emacs, ovvero delle sequenze di caratteri, che, se completate opportunamente, costituiscono un comando:

C-c

C-x

C-h

C-x 4

Esempio: la finestra principale di Emacs può essere suddivisa in due finestre con la sequenza **C-x 2**. Per ingrandire la finestra su cui si sta lavorando e chiudere l'altra si usa la sequenza **C-x 1**.

Emacs: la mode line

La **mode line** si trova sotto la finestra e fornisce informazioni su:

- lo stato del buffer contenente il file che si sta editando,
- il nome del file,
- la percentuale di file visibile (i.e., che sta nella finestra),
- il **modo**:
 - Fundamental (default)
 - text (testo inglese)
 - Lisp mode
 - C mode
 - ...

Esistono:

- major mode (e.g. text)
- minor mode (e.g. Auto Fill: va a capo automaticamente se una riga è troppo lunga)

Emacs: il minibuffer ed alcuni comandi utili

Sotto la mode line si trova il **minibuffer**, che è una parte dell'interfaccia di Emacs che si occupa di visualizzare e di permettere all'utente di editare/completare i comandi.

Comando	Effetto
M-x text-mode	cambia il major mode in text (modalità testo inglese)
C-h m	informazioni sul modo corrente
C-x C-f <i>file</i>	apre il file <i>file</i>
C-x C-s	salva il buffer corrente
C-<Spacebar>	imposta il mark (inizio di una regione di testo su cui eseguire un comando in seguito)
M-w	copia la regione di testo compresa fra il mark e la posizione del cursore
C-w	taglia la regione di testo compresa fra il mark e la posizione del cursore
C-y	incolla la regione di testo copiata/tagliata in precedenza
C-x u	undo
C-x C-c	termina Emacs

Esercizi

- Aprire un file di testo con *vi* ed eseguire le seguenti operazioni:
 - inserire una nuova linea di testo in fondo al file;
 - copiare le ultime 4 linee del file all'inizio del file;
 - sostituire tutte le occorrenze della stringa *are* con il carattere - ;
 - salvare le modifiche.
- Ripetere l'esercizio precedente in *Emacs* utilizzando i caratteri di controllo.
- Eseguire alcune parti del tutorial di *Emacs*.