



Laboratorio di Programmazione

Laurea in Bioinformatica

Web: <http://www.scienze.univr.it/foi/main?ent=oi&id=39990>

Docente: *Carlo Drioli*
Email: *drioli@sci.univr.it*

2008/2009

Lucidi a cura di
Carlo Drioli

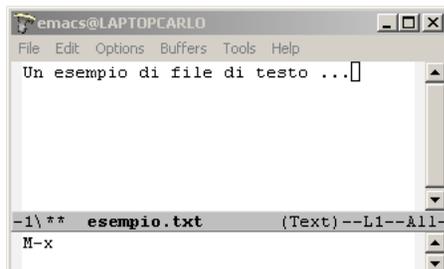
Tool di elaborazione testi in Unix - Emacs

Tool di elaborazione testi in Unix

- In ambiente Unix è particolarmente importante disporre di strumenti efficaci per poter leggere, modificare e scrivere file di testo.
- Molte operazioni di **configurazione e manutenzione** del sistema richiedono la modifica di file testuali.
- Lo **sviluppo di applicazioni** e di **nuove componenti** avviene attraverso la manipolazione di file di testo contenenti istruzioni dei linguaggi di programmazione
- Alcuni programmi di elaborazione di file di testo molto diffusi in ambiente Unix: **vi, emacs, ioe.**

Elaborazione di testi con *emacs*

- Creare o modificare un documento
`$ emacs nomefile`
- Lo spazio dello screen editor è diviso in tre parti
 - Area di testo
 - Riga di stato
 - Area di comando



← Area di testo

← Riga di stato

← Area di comando

Elaborazione di testi con *emacs* (2)

- Emacs opera su tre componenti principali:
 - **File:** è un file memorizzato sul disco. Non viene mai manipolato direttamente, tutte le operazioni vengono eseguite copiando i file in dei buffer di memoria e salvando il risultato delle manipolazioni sui buffer in un file.
 - **Buffer:** è una struttura interna che contiene il testo da elaborare. Possono esserci più buffer attivi allo stesso tempo.
 - **Finestre:** una finestra corrisponde alla visualizzazione di un buffer. E' possibile visualizzare uno o più buffer per volta aprendo e chiudendo finestre durante una sessione di elaborazione del testo.

La riga di stato di *emacs*

- Visualizza informazioni relative al testo corrente.

■ Struttura:



The diagram shows the Emacs status line: `-1 ** esempio.txt (Text) --L1--A11`. Red circles and lines connect parts of the status line to their descriptions below. `-1` is connected to 'Linea', `**` to 'St', `esempio.txt` to 'Nomebuffer', `(Text)` to '(major minor)', `--` to 'L1', and `A11` to 'A11'.

St Nomebuffer (major minor) Linea Posizione

- **St**: indica se il file è stato salvato dopo l'ultima modifica.
“* *” (non salvato), “--” (salvato), “%%” (file di sola lettura)
- **Nomebuffer**: indica il nome del buffer corrente
- **(major minor)**: modalità di editina del file.
major fa riferimento a confiaurazioni di editina per linuacai particolari (es. Lisp, C, testo semplice.etc.)
minor fa riferimento a modalità di inserimento testo particolari
- **Linea**: numero di linea su cui è posizionato il cursore
- **Posizione**: posizione del cursore in relazione all'inizio del file

Comandi principali di *emacs*

- In Emacs i comandi vengono invocati attraverso la combinazione dei tasti CTRL o ALT con altri tasti.

Ad esempio per **uscire** da Emacs si può usare la sequenza **CTRL-x CTRL-c**

Comandi di manipolazione dei file

CTRL-x	CTRL-f	apre un file esistente
CTRL-x	CTRL-s	salva il file corrente
CTRL-x	CTRL-w	salva il file con nome

Comandi principali di *emacs* (2)

Comandi di manipolazione dei buffer

CTRL-x	b	seleziona un buffer attivo o crea un buffer nuovo
CTRL-x	CTRL-b	elenca i buffer attivi
CTRL-x	k	elimina un buffer

Comandi di manipolazione delle finestre

CTRL-x	o	seleziona un'altra finestra tra quelle attive
CTRL-x	0	chiudi la finestra corrente
CTRL-x	1	chiudi tutte le finestre eccetto quella corrente
CTRL-x	2	divide la finestra del buffer corrente in 2 (vert.)
CTRL-x	3	divide la finestra del buffer corrente in 2 (orizz.)
CTRL-v		scorrimento del testo in avanti
ALT-v		scorrimento del testo all'indietro

Comandi principali di *emacs* (3)

Comandi di spostamento del cursore

CTRL-a	sposta il cursore a inizio riga
CTRL-b	sposta il cursore a sinistra di 1 carattere
CTRL-n	sposta il cursore alla riga sottostante
ESC 6 CTRL-b	cursore a sinistra di 6 caratteri
ESC <	sposta il cursore a inizio buffer
ESC >	sposta il cursore a fine buffer

Comandi principali di *emacs* (4)

Comandi di selezione di blocchi

CTRL-barra spazio seana l'inizio del blocco
ESC h definisce come blocco il paragrafo corrente
CTRL-x CTRL-p definisce come blocco la pagina
CTRL-w cancella un blocco
ESC w copia un blocco in un buffer di memoria

Comandi principali di *emacs* (5)

Comandi di cancellazione

CTRL-d cancella il carattere a destra del cursore
BACKSPACE cancella il carattere a sinistra del cursore

Comandi di cancellazione con memorizzazione

CTRL-k cancella la parte della riga a destra del cursore
ESC d cancella parola dopo il cursore
ESC BACKSPACE cancella parola prima del cursore
CTRL-v inserisce dopo il cursore il testo cancellato
CTRL- annulla il comando precedente

Esempi

- **Esempio:** Sequenza che sposta la riga corrente in alto di tre righe.

```
CTRL-k  
ESC 3 CTRL-p  
CTRL v
```

- **Esempio:** Operazioni su paragrafi.

```
ESC h CTRL-w        cancella un paragrafo  
ESC h ESC w         copia un paragrafo
```

- **Esempio:** Copia e incolla. Sequenza che copia la pagina corrente nel buffer di memoria e la incolla all'inizio del file.

```
CTRL-x CTRL-p ESC w  
ESC <  
CTRL-v
```

Comandi principali di *emacs* (6)

Comandi di ricerca di stringhe

CTRL-s cerca un stringa in avanti
ESC CTRL-s cerca un'espressione regolare in avanti
ESC x replace-string eseque una sostituzione globale
ESC x replace-regexp eseque una sostituzione con espressioni regolari
ESC % eseque una sostituzione condizionale
 (auerv-replace)
ALT-x occur cerca un'espressione regolare e salva il risultato

Comandi principali di *emacs* (6)

- Espressioni Regolari
- . Un singolo carattere qualsiasi
 - * Occorrenza del carattere precedente zero o piu' volte
 - + Occorrenza del carattere precedente una o piu' volte
 - ? Occorrenza del carattere precedente zero o una volta
 - {n.m} Occorrenza del carattere precedente almeno n e al max. m volte
 - s|t L'elemento s oppure l'elemento t
 - [...] Insiemi di caratteri
 - [^...] Negazione l'insieme di caratteri fra parentesi
 - [a-z] Un qualunque carattere tra a e z
 - ^ Inizio della riga
 - \$ Fine della riga
 - \< Inizio parola
 - \> Fine parola
 - \ Annulla il significato speciale del metacarattere seguente

Esempio

```
ESC % stringa 1 [Invio] stringa 2 [Invio] opzione
```

- Questa sequenza di comandi permette di sostituire le occorrenze nel testo di **stringa 1** con **stringa 2**, con una procedura interattiva. Dopo il secondo comando di [Invio], all'utente viene chiesto di selezionare per l'occorrenza corrente un'opzione fra le seguenti:
 - **Y** o **barra spazio**
Sostituisce e passa alla prossima occorrenza
 - **N** o **Canc**
Non sostituisce e passa alla prossima
 - **^**
Salta all'occorrenza precedente
 - **.**
Sostituisce l'occorrenza ed esce