

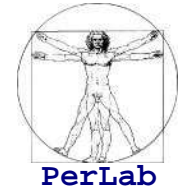
Editor di testo Emacs

Insegnamento di Sistemi Operativi di Rete
Master Universitario in Tecnologie Internet

Ing. Ilaria Giannetti

E-mail: ilaria.giannetti@iet.unipi.it Telefono: 050 2217 468
Pervasive Computing & Networking Lab (PerLab) <http://www.perlab.it>
Dipartimento di Ingegneria dell'Informazione, Università di Pisa

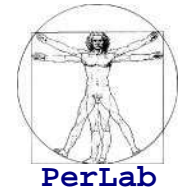
Sommario



- Caratteristiche e funzionalità
- Comandi
 - Base
 - File
 - Navigazione
 - Ricerca
 - Testo
 - Visualizzazione
- Programmazione
 - Scrittura di codice
 - Compilazione
- Composizione e invio mail

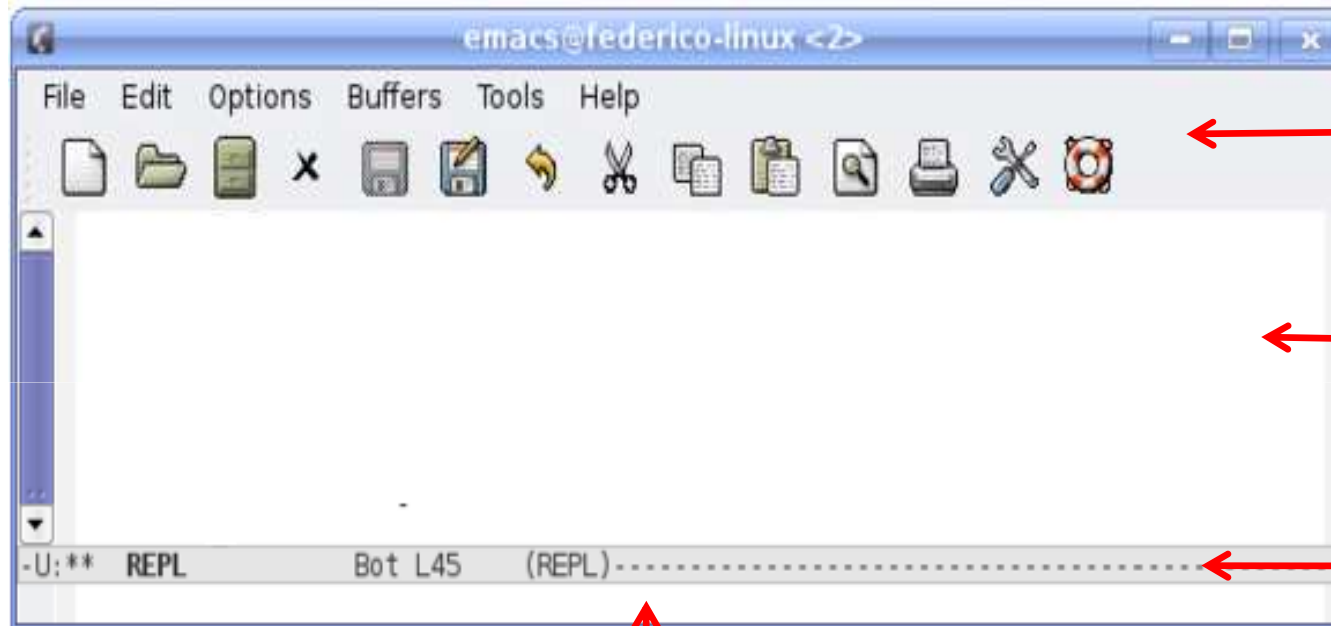
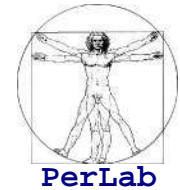
Caratteristiche e funzionalità

Editor Emacs



- Software libero, distribuito sotto i termini della GNU GPL
- Sofisticato, complesso e potente
- Simile a editor tradizionali, non modale (come vi)
 - Inserimento di testo e comandi in un unico ambiente (no doppia modalità)
- Interfaccia utente
 - Riga di comando: con terminale alfanumerico e inserimento di comandi da tastiera
 - Grafica: terminale grafico con menù a tendina selezionabili col mouse

Interfaccia grafica: composizione



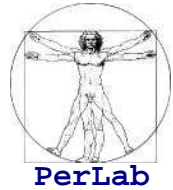
Barra degli
strumenti

Buffer

Barra di stato

Mini-buffer

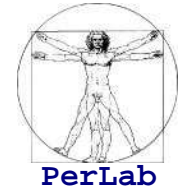
Caratteristiche



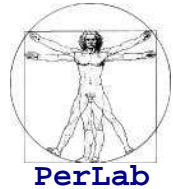
- I comandi sono espressi come **combinazioni di tasti**
- Lavora su un **buffer** e non sul disco e al salvataggio il file viene aggiornato sul disco
- **Point** = posizione del cursore
- **Region** = blocco di testo selezionato

- Scrittura di documenti
- Scrittura di programmi e compilazione del codice
- Lettura di messaggi di posta elettronica

Comandi



- I comandi sono **combinazioni di tasti** del tipo
 - CTRL - carattere
 - ALT - carattere
 - Abbreviazioni usate di seguito:
 - ⇒ **C** = CTRL
 - ⇒ **M** = Alt
- Può sembrare poco pratico ma una volta imparato consente di velocizzare il lavoro, in quanto consente di non alzare mai le mani dalla tastiera



- Le **combinazioni di tasti** sono scorciatoie a funzioni che fanno parte di Emacs
- E' possibile chiamare direttamente le funzioni invece che usare la combinazione di tasti con

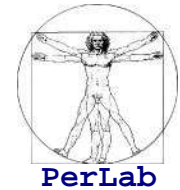
```
M-x nome_funzione <invio>
```

- Tutte le funzioni più comuni hanno le scorciatoie da tastiera, ma non tutte le funzioni
- Chiamare la funzione o usare la scorciatoia da tastiera produce il medesimo effetto
- Esempio, per salvare:

```
M-x save-buffer <invio>
```

```
C-x C-s
```

Installazione



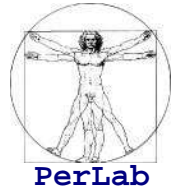
- Debian / Ubuntu

```
sudo apt-get install emacs22
```

- FreeBSD

```
sudo cd /usr/ports/editors/emacs  
make install clean  
rehash  
emacs &
```

Avvio e uscita



- Avvio

- senza specificare il file da creare o aprire

`emacs`

- creazione o apertura di un file già esistente

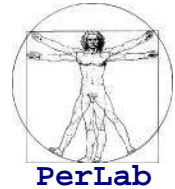
`emacs nome_file`

- in ambiente grafico apre emacs con interfaccia grafica

- opzione `-nw` per aprire l'interfaccia a riga di comando

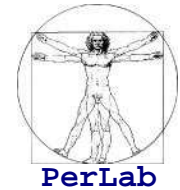
- Uscita

`C-x C-c`



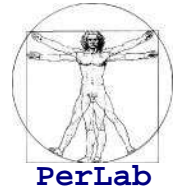
Comandi di base di Help

<code>C-h a espr</code>	<code>a</code> (apropos) : cerca nella documentazione in linea di Emacs l'espressione digitata.
<code>C-h c comando</code>	<code>c</code> : descrizione in breve del comando.
<code>C-h k comando</code>	<code>k</code> : descrizione estesa del comando.
<code>C-h t</code>	<code>t</code> : mostra il tutorial.
<code>C-x 1</code>	Elimina l'help e torna alla prima finestra.



Comandi di base

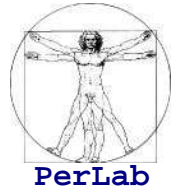
<code>C-x u</code>	(undo) Annulla operazione precedente.
<code>C-x C+g</code>	Esce dal comando corrente.
<code>C-g</code>	Sblocca Emacs bloccato su un comando, fermando il comando in corso Annulla un argomento numerico Annulla l'uso del tasto <ESC> premuto per errore



<ESC><ESC><ESC>

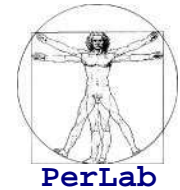
- Comando di uscita abbastanza generale
- Usato per eliminare finestre in più e per uscire dal "minibuffer"
- Ad esempio, usare M-x per entrare nel "minibuffer" ed usare <ESC><ESC><ESC> per uscirne

Esercizio



- Aprire il tutorial
- Cercare informazioni in forma breve e poi estesa sul comando `C-p`
- Tornare al tutorial
- Cercare nella documentazione in linea di Emacs tutto ciò che è relativo alla stringa `find`

Soluzione



C-h t

C-h c C-p

C-h k C-p

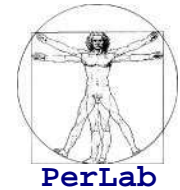
C-x 1 per tornare al tutorial

C-h a find

Mostra tutti i comandi contenenti la parola `find`.

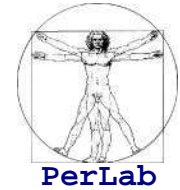
Tali comandi sono eseguibili con M-x comando

Emacs: comandi su file



File	
<code>C-x C-f</code>	Apri un file nel buffer corrente.
<code>C-x C-c</code>	Esce e chiede di salvare.
<code>C-x C-s</code>	Salva il file.
<code>C-x C-w</code>	Salva con nome.
<code>C-x C-v</code>	Cerca e legge un file alternativo.
<code>C-x i</code>	Inserisce il contenuto di un file nella posizione corrente.
<code>C-x C-b</code>	Elenca i buffer attivi.

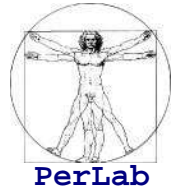
Esercizio



- Creare un file chiamato `uno.txt`, scrivere del contenuto a piacere e salvare il file
- Creare un file chiamato `due.txt`, scrivere del contenuto a piacere e salvare il file
- Spostarsi nel file `due.txt`, inserire in un punto di tale file il contenuto del file `uno.txt` e salvare

- Guardare l'elenco dei buffer attivi

Soluzione



C-x C-f e uno.txt, <scrivere>, C-x C-s

C-x C-f e due.txt, <scrivere>, C-x C-s
<spostare il cursore>

C-x i e uno.txt, C-x C-s

C-x C-b

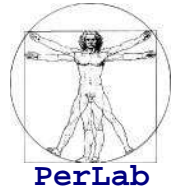
Emacs: comandi la navigazione1 / 2



Movimenti del cursore

C-p	Linea precedente.
C-n	Linea successiva.
C-f	Avanti di un carattere.
C-b	Indietro di un carattere
C-a	Inizio linea.
C-e	Fine linea.
C-v	Sfoggia avanti.
M-v	Sfoggia indietro.

Emacs: comandi la navigazione2 / 2



Movimenti del cursore

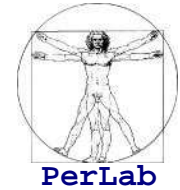
M-f	Avanti di una parola.
M-b	Indietro di una parola.
M-a	All'inizio della frase.
M-e	Alla fine della frase.
M-<	Inizio buffer.
M->	Fine buffer.
M+ -x goto-line <invio> n	Muove il cursore alla linea n.

- Nota: è possibile indicare un argomento numerico che indica il numero di ripetizioni dell'operazione

C-u num comando

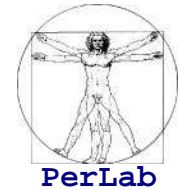
Esempio: 4 righe avanti C-u 4 C-n

Esercizio



- Aprire il tutorial
- Eseguire i seguenti spostamenti:
 - Due pagine avanti
 - Alla linea precedente
 - Tre parole avanti
 - All'inizio del buffer
 - Alla fine della frase
 - Alla riga 47

Soluzione



C-h t

C-v C-v (eccezione per cui C-u non funziona)

C-p

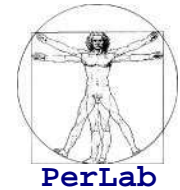
C-u 3 M-f oppure M-f M-f M-f

M+<

M+e

M-x goto-line <invio> 47

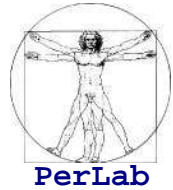
Emacs: comandi per la ricerca



Ricerca	
<code>C-s stringa</code>	Ricerca in avanti.
<code>C-r stringa</code>	Ricerca al rovescio.
<code>M-x replace-string <invio> stringa <invio> nuovastringa <invio></code>	Cerca e rimpiazza una stringa.

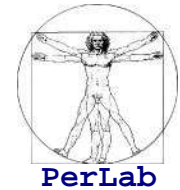
- Ricerca incrementale
- Per trovare il punto in cui stringa compare di nuovo digitare ancora `C-s` o `C-r`

Esercizio



- Aprire il tutorial
- Cercare in avanti la parola `scorrimento`
- Scorrere alle successive
- Cercare all'indietro la parola `cursore`
- Scorrere alle precedenti
- Spostarsi all'inizio del documento e sostituire la parola `scorrimento` con `ciao`

Soluzione



C-h t

C-s scorrimento

C-s C-s ..

C-r cursore

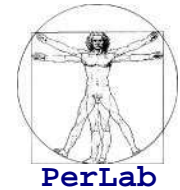
C-r C-r ..

M-<

M-x replace-string <invio>

scorrimento <invio>

Ciao <invio>



Cancellazione testo

C-d	Cancella un carattere.
M-d	Cancella dal cursore fino a fine parola.
C-k	Cancella dal cursore fino a fine linea.
M-k	Cancella dal cursore fino alla fine della frase.

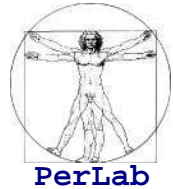
Trasformazione testo

M-u	Maiuscolo dal carattere corrente fino a fine parola.
M-l	Minuscolo dal carattere corrente fino a fine parola.

- Nota: è possibile indicare un argomento numerico che indica il numero di ripetizioni dell'operazione

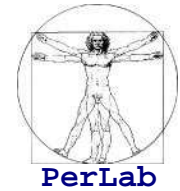
C-u num comando

Esercizio



- Aprire il tutorial
- Cancellare i 6 caratteri successivi al cursore
- Cancellare le 3 parole successive al cursore
- Cancellare le 2 righe successive al cursore
- Cancellare la frase successiva al cursore
- Trasformare in maiuscolo le 10 parole successive al cursore

Soluzione



C-h t

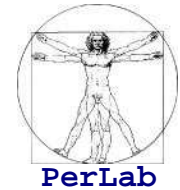
C-u 6 C-d

C-u 3 M-d

C-u 2 C-k

M-k

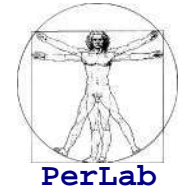
C-u 10 M-u



Copia, Taglia, Incolla

C- <code><space></code>	Per marcare una posizione.
M-w	Copia la parte evidenziata o dalla marcatura al cursore.
C-w	Taglia/cancella la parte evidenziata o dalla marcatura al cursore.
C-y	Incolla.

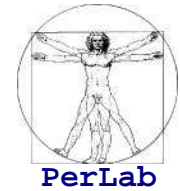
Esercizio



- Porre una marcatura sul testo, spostarsi in un altro punto e copiare dalla marcatura al cursore
- Spostarsi in un altro punto del testo e incollare

- Porre una marcatura sul testo, spostarsi in un altro punto e copiare dalla marcatura al cursore
- Cancellare il testo
- Annullare l'operazione precedente

Soluzione



C-<spazio>, <spostarsi nel testo>, M-w

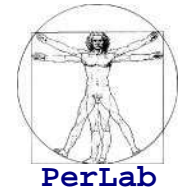
C-y

C-<spazio>, <spostarsi nel testo>, M-w

C-w

C-x u

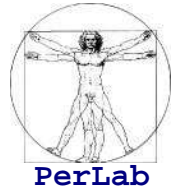
Emacs: comandi di visualizzazione



Dividere l'ambiente in più finestre

C+x 2	Divide l'ambiente in 2 finestre.
C+x 1	Cancella tutte le finestre ad eccezione di quella corrente.
C+x o	Si sposta sull'altra finestra.

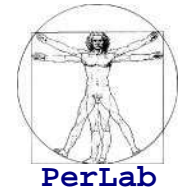
Esercizio



- Dividere l'ambiente in 2 finestre
- Aprire in una il file `uno.txt`, spostarsi nell'altra ed aprire il file `due.txt`
- Chiudere tutte le finestre ad eccezione di quella corrente

- Riaprire il tutorial

Soluzione



C-x 2

C-x C-f uno.txt

C-o

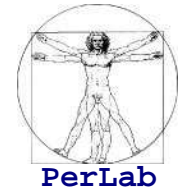
C-x C-f due.txt

C-x 1

C-h t

Programmazione

Emacs: comandi per scrittura codice

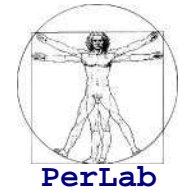


- supporta le modalità di un'ampia gamma di linguaggi di programmazione come **C** e **C++**.

Supporto alla programmazione: scrittura del codice

<code>M-x [language]-mode</code>	Passa ad una determinata modalità di linguaggio (per il C: <code>M-x c-mode</code>). <code>M-x fundamental-mode</code> per tornare alla visualizzazione normale.
<code>M-x indent-region</code>	Per indentare con il tasto TAB.
<code>M-x font-lock-mode</code>	Attiva la colorazione della sintassi.

Emacs: compilazione



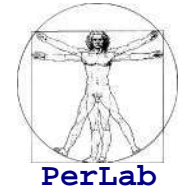
- Chiama compilatori esterni
- Mostra il risultato in un buffer
 - **Compilation finished:** se la compilazione va a buon fine
 - **Compilation exited abnormally:** se la compilazione non va a buon fine. Nel buffer dedicato ai messaggi del compilatore viene visualizzato l'elenco dei messaggi di errore con indicazioni sulla linea del sorgente

Supporto alla programmazione: compilazione

`M+x compile`

Chiama il **compilatore** al quale bisogna passare i comandi per compilare, es.: `gcc -o main main.c`

Esercizio

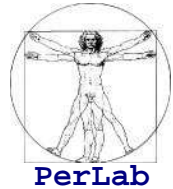


- Creare un file chiamato `prova.c`
- Passare alla modalità C
- Scrivere nel file il programma Hello World!

```
#include <stdio.h>
int main(){
    printf("Hello World!\n");
    return 0;
}
```

- Compilare con il comando
`gcc -o prova prova.c`
- Lanciare da terminale l'eseguibile `prova` col comando `./prova`

Soluzione



emacs prova.c oppure C-x C-f e prova.c

- Scrivere nel file

```
#include <stdio.h>
int main(){
    printf("Hello World!\n");
    return 0;
}
```

- Salvare C-x C-s

- Digitare M-x compile e

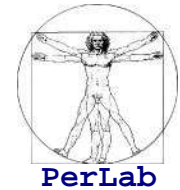
```
gcc -o prova prova.c <invio>
```

- Aprire un terminale e digitare

```
./prova
```

Composizione e invio mail

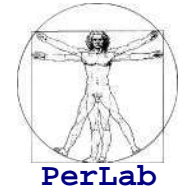
Emacs: comandi per scrittura codice



Supporto alla programmazione: scrittura del codice

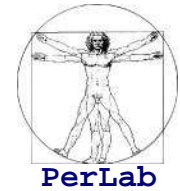
<code>C-x m</code>	Aprire la finestra in cui comporre il messaggio da inviare via mail.
<code>C-x 4 m</code>	Aprire una seconda finestra in cui comporre il messaggio da inviare via mail.
<code>C-x 5 m</code>	Aprire un nuovo frame in cui comporre il messaggio da inviare via mail.
<code>C-c C-s</code>	In mail-mode inviare il messaggio.
<code>C-c C-c</code>	In mail-mode inviare il messaggio e chiudere il mail buffer.

Esercizio



- Comporre una mail in una seconda finestra
- Inviarla e chiudere con un unico comando

Soluzione

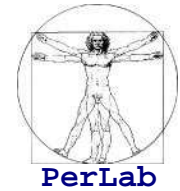


C-x 4 m

<scrivere mail>

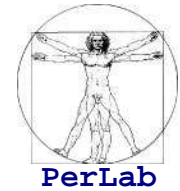
C-c C-c

Esercizio



- Comporre una mail in una seconda finestra
- Inviarla e chiudere con un unico comando

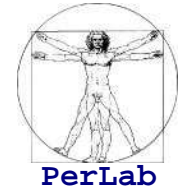
Soluzione



C-x 4 m

<scrivere mail>

C-c C-c



- Riferimenti:

- Dispensa 'Amministrazione di un Sistema UNIX in Rete'
Cap. 3 Paragr. 2

http://www2.ing.unipi.it/~a008149/corsi/so/materiale/Amministrazione_UNIX.pdf

- Tutorial Emacs:

<http://lists.linux.it/pipermail/tp/2002-August/002942.html>