

MUOVERSI IN EMACS

LEONARDO TAMIANO

Created: 2023-12-31 Sun 11:50

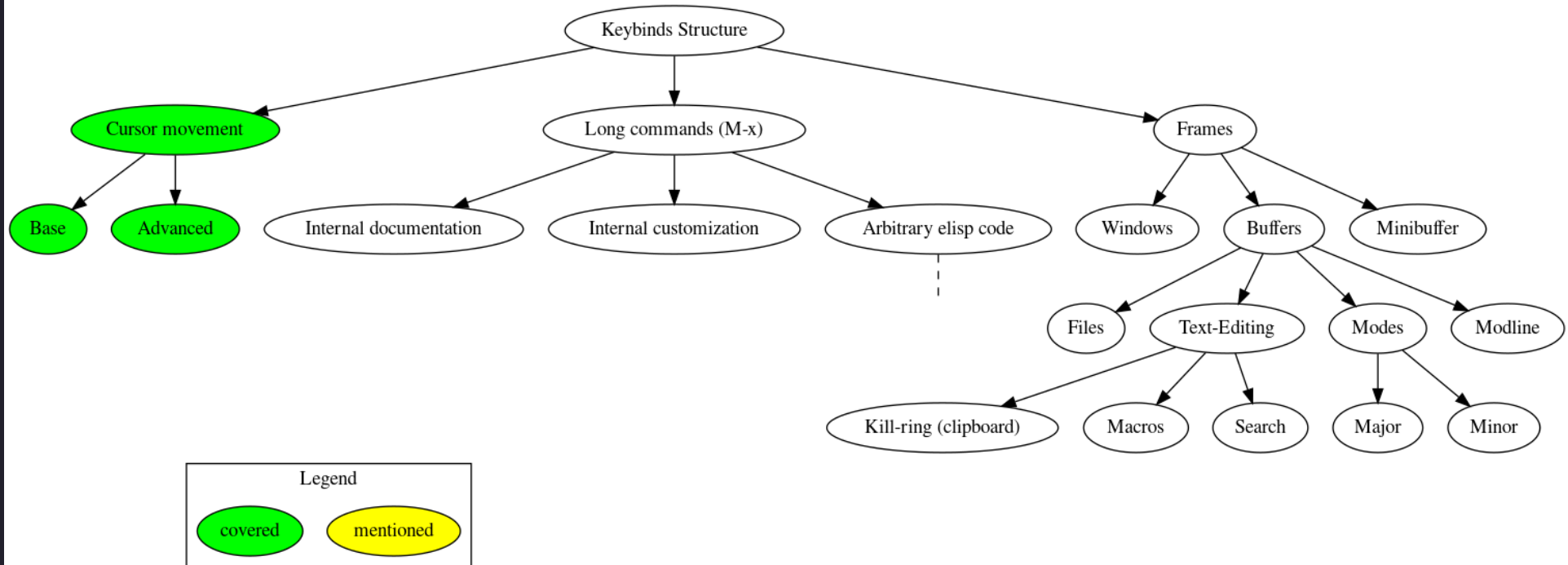
TABLE OF CONTENTS

- Overview
- Definizioni
- Movimenti Cursore Base
- Perché?
- Movimenti Cursore Avanzati
- Movimento Schermo
- Esercizi

OVERVIEW

Utilizzando lo schema introdotto nel video precedente possiamo inquadrare i concetti che affronteremo in questo video.

Emacs Baseline Concepts



DEFINIZIONI

Prima di procedere qualche definizione utile...

PUNTO

Il **punto** (point) è il luogo in cui emacs inserisce il prossimo carattere che scriviamo.

Ogni **buffer** ha associato il proprio punto.

In **elisp** il punto è rappresentato come un intero.
La funzione (`point`) ritorna l'intero corrispondente al
punto nel buffer corrente.

Per testare il valore del point possiamo eseguire il seguente codice elisp, andandolo a copiare in un buffer vuoto e chiamando il comando `eval-buffer`.

```
(defun print-point()  
  (interactive)  
  (message "%d" (point))  
  )  
  
(global-set-key (kbd "C-c C-y") 'print-point)
```

Senza entrare nel dettaglio, tale codice definisce una funzione che stampa il valore del punto nel buffer in cui ci troviamo. Tale funzione viene bindata al keybind

`C-c C-y`.

Esempio di utilizzo (vedere il minibuffer)

```

Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua. Ut enim ad minim veniam, quis
nostrud exercitation ullamco laboris nisi ut aliquip ex
ea commodo consequat. Duis aute irure dolor in
reprehenderit in voluptate velit esse cillum dolore eu
fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia
deserunt mollit anim id est laborum.

```

scratch 1:0 All LF UTF-8 Text (+3)
1

Inizio buffer

```

Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua. Ut enim ad minim veniam, quis
nostrud exercitation ullamco laboris nisi ut aliquip ex
ea commodo consequat. Duis aute irure dolor in
reprehenderit in voluptate velit esse cillum dolore eu
fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia
deserunt mollit anim id est laborum.

```

scratch 1:444 12:02 LF UTF-8 Text (+3)
445

Fine buffer

CURSORE

Il **cursore** (cursor) invece rappresenta il "punto grafico", ovvero ciò che possiamo vedere visivamente mentre scriviamo.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

scratch 1:167 All

LF UTF-8 Text (+3)

Il cursore è il riquadro sopra la lettera i

Possiamo cambiare il tipo di cursore utilizzato tramite la variabile `cursor-type`. I valori possibili sono


<code>t</code>	use the cursor specified for the frame
<code>nil</code>	don't display a cursor
<code>box</code>	display a filled box cursor
<code>hollow</code>	display a hollow box cursor
<code>bar</code>	display a vertical bar cursor with default width
<code>(bar . WIDTH)</code>	display a vertical bar cursor with width WIDTH
<code>hbar</code>	display a horizontal bar cursor with default height
<code>(hbar . HEIGHT)</code>	display a horizontal bar cursor with height HEIGHT
<code>ANYTHING ELSE</code>	display a hollow box cursor

Tra i particolari valori i più interessanti sono box e bar.

```

Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua. Ut enim ad minim veniam, quis
nostrud exercitation ullamco laboris nisi ut aliquip ex
ea commodo consequat. Duis aute irure dolor in
reprehenderit in voluptate velit esse cillum dolore eu
fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia
deserunt mollit anim id est laborum.

```

 *scratch* 1:167 All


LF UTF-8 Text (+3)

Cursore box

```

Lorem ipsum dolor sit amet, consectetur adipiscing
elit, sed do eiusmod tempor incididunt ut labore et
dolore magna aliqua. Ut enim ad minim veniam, quis
nostrud exercitation ullamco laboris nisi ut aliquip ex
ea commodo consequat. Duis aute irure dolor in
reprehenderit in voluptate velit esse cillum dolore eu
fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia
deserunt mollit anim id est laborum.

```

 *scratch* 1:168 All

LF UTF-8 Text (+3)

Cursore bar

Quando utilizziamo un cursore tipo box, ovvero quando il cursore appare sopra un carattere, il punto è immediatamente prima del carattere. Questo significa che la successiva lettera sarà scritta alla sinistra di quella coperta dal cursore.

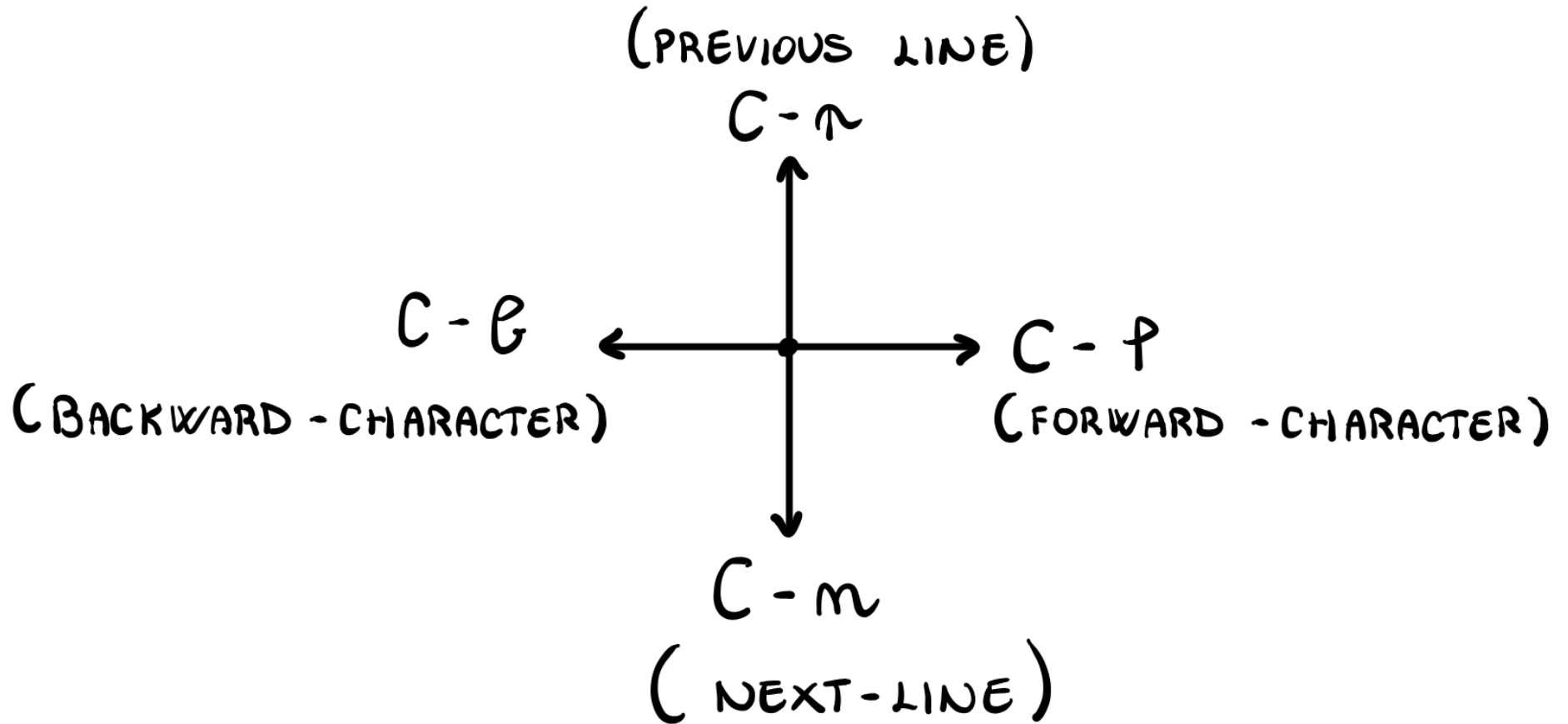
Se invece utilizziamo un cursore di tipo bar, allora il punto è nella stessa posizione del cursore.

MOVIMENTI CURSORE BASE

I movimenti base del cursore sono i seguenti

Keybind	Comando
C-p	previous-line
C-n	next-line
C-b	backward-char
C-f	forward-char

Graficamente



PERCHÉ?

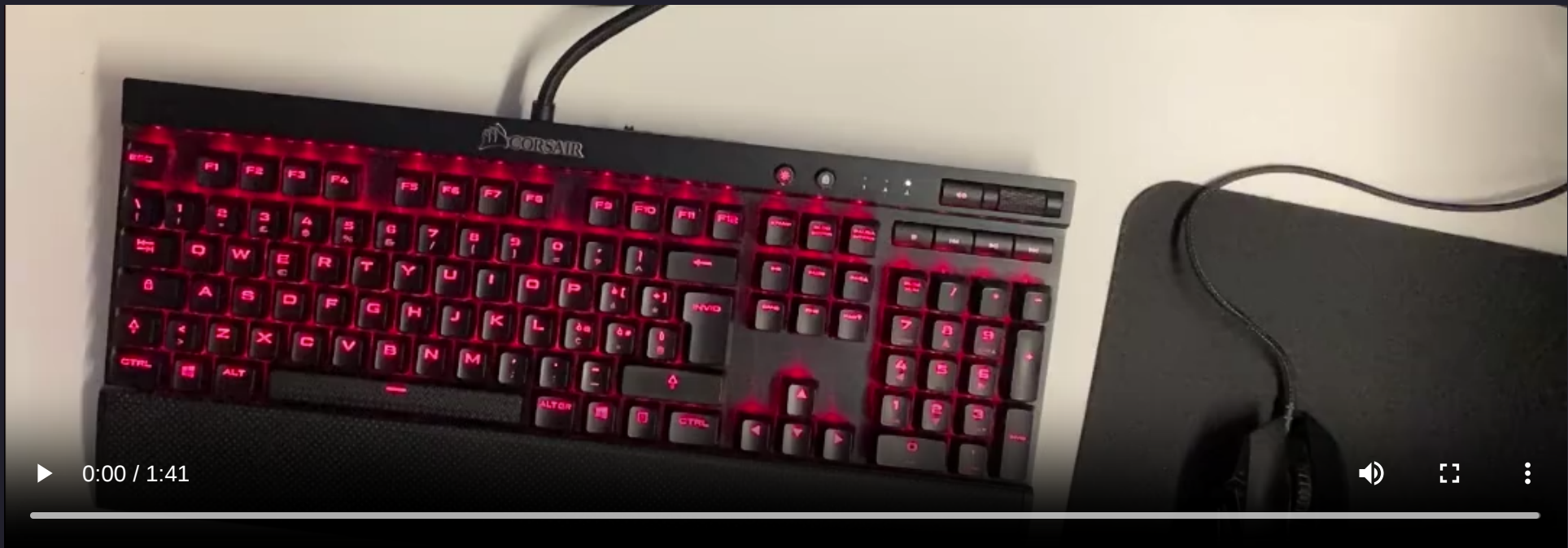
Inizialmente i keybinds per muoversi in Emacs potrebbero confondere, specialmente a chi non è abituato a questa tipologia di software (Emacs/Vim/Unix).

Detto questo, c'è una chiara e semplice motivazione dietro a questi keybinds: **non allontanare la posizione delle mani dal centro della tastiera**

Infatti, il tempo impiegato per muoversi dal centro della tastiera al mouse, per quanto piccolo possa essere, diventa significativo quando consideriamo il numero di volte in cui dobbiamo farlo.

Utilizzando i keybinds di Emacs siamo in grado di restare sempre al centro della tastiera, in modo da avere accesso alla maggior parte dei tasti senza dover spostare la mano.

Consideriamo la mia tastiera...



MOVIMENTI CURSORE AVANZATI

Oltre ai movimenti base, che ci spostano di un carattere alla volta, ci sono altri keybinds che ci permettono di effettuare degli spostamenti più ampi.

PER PAROLA

Keybind	Comando
---------	---------

M-f	forward-word
-----	--------------

M-b	backward-word
-----	---------------

INIZIO/FINE LINEA

Keybind	Comando
C-a	(beginning-of-visual-line)
C-e	(end-of-visual-line)

PER FRASE

Keybind	Comando
---------	---------

M-a	(backward-sentence)
-----	---------------------

M-e	(forward-sentence)
-----	--------------------

PER PARAGRAFO

Keybind	Comando
M-{	(backward-paragraph)
M-}	(forward-paragraph)

MOVIMENTO SCHERMO

Oltre a muovere il cursore, è possibile anche muovere lo schermo, ovvero la sezione del buffer visibile. Questa tipologia di movimenti sono molto utili quando abbiamo a che fare con file molto grandi.

Il movimento dello schermo può essere effettuato sia con il mouse che automaticamente muovendo il cursore.

Abbiamo poi i seguenti keybinds

Keybind	Comando
C-v	scroll-down-command
M-v	scroll-up-command
M-<	beginning-of-buffer
M->	end-of-buffer
C-l	recenter-top-bottom

Infine, troviamo i seguenti comandi

Keybind	Comando
-	goto-line
-	goto-char

che sono molto utili quando dobbiamo, ad esempio, capire un errore di compilazione.

ESERCIZI

TASK #1

Assegnare il comando `goto -line` ad un keybind a scelta, e utilizzarlo.

TASK #2

Aprire un file e muoversi nel file utilizzando solamente i keybinds per il movimento base, senza arrows né mouse.

TASK #3

Cambiare lo stile del cursore al valore \hbar .

