

# Princeton University

## COS 217: Introduction to Programming Systems

### The Emacs Editor

This reference sheet assumes that Emacs is configured using the .emacs file provided to COS 217 students.

To type "Ctrl-*somechar*" (for any character *somechar*), type the *somechar* key while holding down the Ctrl key. To type "Esc *somechar*" (for any character *somechar*), type the Esc key followed by the *somechar* key. Typing "Alt- *somechar*" has the same effect as typing "Esc *somechar*".

In Emacs all work is accomplished by calling functions. To call a function, type "Esc x *function*".

Many functions are bound to keystrokes.

Commonly used functions are in **boldface**.

#### Moving the Point

Binding	Function	Description
→	<b>forward-char</b>	<b>Move the point forward one character</b>
←	<b>backward-char</b>	<b>Move the point backward one character</b>
↓	<b>next-line</b>	<b>Move the point to the next line</b>
↑	<b>previous-line</b>	<b>Move the point to the previous line</b>
Ctrl-f	forward-char	Move the point forward one character
Ctrl-b	backward-char	Move the point backward one character
Ctrl-n	next-line	Move the point to next line
Ctrl-p	previous-line	Move the point to previous line
Esc f	forward-word	Move the point to next word
Esc b	backward-word	Move the point to previous word
<b>Home</b>	<b>beginning-of-line</b>	<b>Move the point to beginning of line (but not with some terminal apps)</b>
<b>End</b>	<b>end-of-line</b>	<b>Move the point to end of line (but not with some terminal apps)</b>
<b>Ctrl-a</b>	<b>beginning-of-line</b>	<b>Move the point to beginning of line</b>
<b>Ctrl-e</b>	<b>end-of-line</b>	<b>Move the point to end of line</b>
Esc a	c-beginning-of-statement	Move the point to the beginning of C statement
Esc e	c-end-of-statement	Move the point to the end of C statement
<b>PageDn</b>	<b>scroll-up</b>	<b>Move the point to next page (but not with some terminal apps)</b>
<b>PageUp</b>	<b>scroll-down</b>	<b>Move the point to previous page (but not with some terminal apps)</b>
<b>Ctrl-v</b>	<b>scroll-up</b>	<b>Move the point to next page</b>
<b>Esc v</b>	<b>scroll-down</b>	<b>Move the point to previous page</b>
<b>Esc &lt;</b>	<b>beginning-of-buffer</b>	<b>Move the point to beginning of the buffer</b>
<b>Esc &gt;</b>	<b>end-of-buffer</b>	<b>Move the point to end of the buffer</b>
Esc Ctrl-a	beginning-of-defun	Move the point to beginning of the C function
Esc Ctrl-e	end-of-defun	Move the point to end of the C function
<b>Ctrl-x   <i>line</i></b>	<b>goto-line</b>	<b>Move the point to line whose number is <i>line</i></b>

#### Inserting and Deleting

Binding	Function	Description
<b>Bsp</b>	<b>c-electric-backspace</b>	<b>Delete the character before the point</b>
Esc Bsp	backward-kill-word	Delete the characters from the point to the beginning of the word
<b>Ctrl-d</b>	<b>c-electric-delete-forward</b>	<b>Delete the character at the point</b>
<b>Ctrl-k</b>	<b>kill-line</b>	<b>Cut the current line</b>
<b>Ctrl-Sp</b>	<b>set-mark-command</b>	<b>Set the mark at the point</b>
Ctrl-x Ctrl-x	exchange-point-and-mark	Exchange the mark and the point
Ctrl-x h	mark-whole-buffer	Set the point at the beginning and the mark at the end of the buffer
<b>Ctrl-w</b>	<b>kill-region</b>	<b>Cut the region denoted by the mark and the point</b>
Esc w	kill-ring-save	Copy the region denoted by the mark and the point
<b>Ctrl-y</b>	<b>yank</b>	<b>Paste the previously cut/copied region at the point</b>

## Saving and Exiting

Binding	Function	Description
<b>Ctrl-x Ctrl-s</b>	<b>save-buffer</b>	<b>Save the current buffer to its file</b>
Ctrl-x Ctrl-w <i>file</i>	write-file	Write the current buffer to <i>file</i>
Ctrl-x Ctrl-q	vc-toggle-read-only	Toggle the current buffer between read-only and read/write
<b>Ctrl-x Ctrl-c</b>	<b>save-buffers-kill-emacs</b>	<b>Save all buffers and exit Emacs</b>

## Indenting

Binding	Function	Description
Ctrl-c .	c-set-style	Set the C indentation style to the specified one
<b>TAB</b>	<b>c-indent-command</b>	<b>Indent the current line of the C program</b>
Esc Ctrl-\	indent-region	Indent the region of the C program denoted by the mark and the point
<b>Ctrl-x p</b>	<b>indent-all</b>	<b>Indent all lines of the C program (i.e. indent the program perfectly)</b>

## Searching and Replacing

Binding	Function	Description
<b>Ctrl-s <i>string</i></b>	<b>isearch-forward</b>	<b>Search forward for <i>string</i></b>
Ctrl-r <i>string</i>	isearch-backward	Search backward for <i>string</i>
<b>Esc % <i>old new</i></b>	<b>query-replace</b>	<b>Replace the <i>old</i> string with the <i>new</i> one</b> y => replace n => skip ! => replace all q => quit

## Managing Windows and Buffers

Binding	Function	Description
<b>Ctrl-x Ctrl-f <i>file</i></b>	<b>find-file</b>	<b>Load <i>file</i> into a buffer</b>
Ctrl-x Ctrl-r <i>file</i>	find-file-read-only	Load <i>file</i> into a buffer for read only
<b>Ctrl-x 2</b>	<b>split-window-vertically</b>	<b>Split the current window into two windows arranged vertically</b>
<b>Ctrl-x o</b>	<b>other-window</b>	<b>Move the point to the other window</b>
Ctrl-x 3	split-window-horizontally	Split the current window into two windows arranged horizontally
Ctrl-x 0	delete-window	“Undisplay” the current window
<b>Ctrl-x 1</b>	<b>delete-other-windows</b>	<b>“Undisplay” all windows except the current one</b>
Ctrl-x Ctrl-b	list-buffers	Display a new window listing all buffers
Ctrl-x b <i>file</i>	switch-to-buffer	Load <i>file</i> into a buffer if necessary, and then display that buffer in the current window

## Building and Debugging

Binding	Function	Description
	<b>compile <i>command</i></b>	<b>Build the program using <i>command</i></b>
	<b>gdb <i>executablefile</i></b>	<b>Launch the GDB debugger to debug <i>executablefile</i></b>

## Miscellaneous

Binding	Function	Description
Ctrl-x u	undo	Undo the previous change
<b>Ctrl-_</b>	<b>undo</b>	<b>Undo the previous change</b>
<b>Ctrl-g</b>	<b>keyboard-quit</b>	<b>Abort the multi-keystroke command</b>
Ctrl-h	help-command	Access the Emacs help system
Esc `	tmm-menubar	Access the Emacs menu
<b>Ctrl-x n</b>	<b>linum</b>	<b>Display/undisplay a line number before each line</b>