GNU Emacs
Quick Reference Guide
for CS61A

Bullets (•) mark a suggested starting set of commands. Daggers (†) denote key bindings that are not standard in GNU Emacs. *ARG* denotes the prefix numeric argument (entered with `C-u` or `M-`*digit*). The notation '`C-`*x*' means "control-*x*", the result of holding down the control key while typing *x*. '`M-`*x*' means "meta-*x*", the result of holding down the '`Meta`', `Alt`, or ◇ key (depending on keyboard) while typing *x*. If you are not using a window system or have a keyboard without these keys, the sequence of two characters `ESC` *x* is equivalent. The notation '`C-M-`*x*' is equivalent to holding down both control and meta keys while typing *x*, or of typing the two characters `ESC C-`*x*.

Cursor motion.

`C-f`	Forward character.•
`C-b`	Backward character.•
`M-f`	Forward word.•
`M-b`	Backward word.•
`C-e`	Forward to end of line.•
`C-a`	Backward to start of line.•
`C-M-f`	Forward S-expression.•
`C-M-b`	Backward S-expression.•
`M-e`	Forward sentence.
`M-[`	Forward paragraph.
`M-a`	Backward sentence.
`M-]`	Backward paragraph.

`C-n`	Next line.•
`C-p`	Previous line.•
`M-<`	Beginning of buffer.•
`M->`	End of buffer.•
`C-v`	Scroll text up one screen (or *ARG* lines).•
`M-v`	Scroll text down (or *ARG* lines).•
`M-g`	Go to line number *ARG*.†
`M-x what-line`	Display line number.
`C-M-v`	Scroll other window up one screen (or *ARG* lines).

Marking regions of text

`C-@`	Set mark at point.•
`C-x C-x`	Exchange mark and point.•
`C-SPC`	Same as `C-@`.
`M-@`	Set mark after end of next word.
`M-h`	Set mark and point around current paragraph.
`C-x h`	Set mark and point around current buffer.

Deletion and yanking

`DEL`	Delete character before cursor.•
`M-DEL`	Delete word before cursor and add to kill buffer.•
`C-d`	Delete character at cursor.•
`M-d`	Delete word at and after cursor and add to kill buffer.•
`C-k`	Delete to end of line and add to kill buffer.•
`C-w`	Delete current region, and add to kill buffer.•
`M-w`	Copy current region to kill buffer without deleting.•
`M-\`	Delete surrounding blanks and tabs.
`M-SPC`	Delete all but one surrounding blank.
`C-x C-o`	Delete all but one surrounding blank line.
`C-M-w`	Cause next command, if a kill, to append to previous kill buffer, instead of new one.

`C-y`	Insert text from kill buffer at point.•
`C-u C-y`	Insert text from kill buffer at point without moving point.
`M-y`	Replace preceding `C-y` text with next most recent kill buffer.
`M-w`	Copy region to kill buffer, no deletion.

Indentation

`TAB`	Indent according to mode.•
`LFD`	Same as RET TAB.
`M-;`	Indent and start comment.
`M-LFD`	Continue comment on next line.
`C-x TAB`	Indent region rigidly by *ARG*.
`C-M-\`	Indent region according to mode.•

Search

`C-s`	Search forward.•
`C-s C-s`	Same as `C-s` with last string.•
`C-r`	Search backward.•

`C-r C-r`	Same as `C-r` with last string.•
`C-u C-s`	Search forward for regular expression.
`M-x occur`	Display lines matching a regular expression.
`M-x grep`	Display results of UNIX `grep` utility.
`M-x count-matches`	

The following subcommands are valid during a search.

`RET`	End search.•
`DEL`	Undo effect of last search character typed.•
`C-g`	Abort search.•
`C-s`	Search for next match forward.•
`C-r`	Search for next match backward.•
`C-q`	Quote next character.
`C-w`	Extend search string with next word.
`LFD`	Extend search string with rest of line.

Replacement

`M-%`	Query replace.•
`M-x delete-matching-lines`	
`M-x delete-non-matching-lines`	

During a query-replacement, the following are valid responses to prompts.

`SPC`	Make replacement and go to next.•
`DEL`	Skip replacement and go to next.•
`RET`	End replacement.•
`!`	Replace all remaining instances without asking.•
`C-r`	Enter recursive edit; return with `C-M-c`.

Regular expressions

`.`	Match any character.•
`^`	Match at start of line.•
`$`	Match at end of line.•
`[...]`	Match any character in the '...'.•
`[^...]`	Match any character except those in '...'.•
`*`	Match 0 or more of pattern to left.•
`+`	Match 1 or more of pattern to left.
`?`	Match 0 or 1 of pattern to left.
`\`*c*	Quotes *c*, except for the following.
`\b`	Match at beginning or end of word.
`\B`	Match except at beginning or end of word.
`|`	Match either pattern to left or right.
`\(...\)`	Grouping.
`\`*n*	Match copy of whatever matched n^{th} group.

Miscellaneous editing

`C-o`	Insert newline after cursor.
`C-t`	Transpose characters.
`M-t`	Transpose words.
`C-x C-t`	Transpose lines.
`M-u`	Convert whole word to upper case.
`M-l`	Convert whole word to lower case.
`M-c`	Capitalize word.

Files

`C-x C-f`	Find file; load if needed.•
`C-x 4 C-f`	Find file in other window.•
`C-x C-s`	Save file.•
`C-x C-w`	Write to explicitly-named file.
`C-x i`	Insert file at cursor.
`M-x recover-file`	Recover file after disaster from auto-save file.
`M-x revert-buffer`	Throw away changes to buffer and restore from file.

Buffers and windows

`C-x o`	Put cursor in other window.•
`C-x 1`	Grow current window to full screen.•
`C-x 2`	Split current window vertically.•
`C-x b`	Put named buffer in window.•
`C-x 0`	Remove current window.
`C-x 3`	Split current window horizontally.
`C-x C-b`	List all buffers.
`C-x k`	Delete buffer.

Shells

`M-x shell`	Run UNIX shell in a buffer.•
`M-|`	Execute single shell command on region. With *ARG*, replaces region.

Commands active in shell buffers:

`RET`	Send current line to shell.•
`TAB`	Complete preceding file name.•
`C-c C-c`	Send interrupt to shell.•

C-c C-u	Erase current input line.•
C-c C-z	Send stop signal to shell.•
C-c C-d	Send EOF to shell.

Scheme

M-x run-scheme	Run Scheme interpreter in buffer *scheme*.•
C-c C-z	Put the cursor in buffer *scheme*.•
C-c C-e	Send definition to *scheme*.
C-c M-e	Same as C-c C-e C-c C-z.
C-c C-l	Load file into *scheme*.
C-c M-l	Same as C-c C-l C-c C-z.•†
C-c C-r	Send current region to *scheme*.
C-c M-r	Same as C-c C-r C-c C-z.
M-C-q	Indent Scheme expression.•
M-C-\	Indent current region.
TAB	Indent the current line.•
LFD	Same as RET TAB.

Compilation, debugging, and tags

M-x compile	Execute command (by default, make) asynchronously.
C-x ‘	Position to next error or next line found by M-x grep command.
M-x kill-compiler	Stop active compile.

M-x visit-tags-table	Specify file containing tags produced by etags.
M-.	Display source for given tag.
C-u M-.	Find next alternate definition for last tag.
C-x 4 .	Display source for tag in other window.
M-x tags-search	
M-x tags-query-replace	Look for pattern in all files named in tags table.
M-x tags-apropos	Display matching tags.

M-x gdb	Run GNU debugger on file.

Commands valid in gdb mode.

C-c C-s	Step.
C-c C-n	Next.
C-c <	Up stack.
C-c >	Down stack.
C-c C-r	Finish.
C-x SPC	In any source file, sets a break point.
C-c C-i	Stepi.

Help and documentation

M-x manual-entry	UNIX man page for given topic.
C-h a	Look up names of matching Emacs commands.•
C-h b	Display key bindings.•
C-h f	Help for M-x function.•
C-h C-h	Help for C-h.•
C-h i	Run info browser.•
C-h k	Help for key.•
C-h m	Help for current mode.
C-h t	Tutorial.
C-h w	Key containing function.

Inside an *info* buffer (result of C-h i), the following are defined.

m	Select menu item.•
l	Go to last-visited node.•
?	Get help for browser.•
u	Go to node's parent.•
n	Go to next node in sequence.•
q	Leave browser.•
.	Go to top of node.
d	Go to top-level node.

Mouse commands

Left, middle, and right buttons are LB, MB, and RB.

LB	Put cursor at mouse. Dragging marks region.•
MB	Paste text from window-system cut buffer at mouse.•
RB	Extend region to pointer and copy into cut and kill buffers. Clicking twice deletes region.•
C-LB	Select a buffer.