Emacs Basics

These are only some very basic facts about emacs. To learn more, the on-line tutorial within emacs itself is very good. To get to it type (within emacs):

CTRL-h t

To execute **Control** commands, written as **CTRL-*letter***, hold down the Control key while typing the letter indicated.
To execute **Escape** commands, written as **ESC *letter***, press the Escape key (once) and then type the letter indicated.

To **start emacs**, type

emacs *filename*

and hit the **Return** key.

To **exit emacs,** type

CTRL-x CTRL-c

You will be asked whether you want to save the file.

You can also **save** your work without exiting, by typing

CTRL-x CTRL-s

It is a good idea to do this regularly while editing your files, although emacs is pretty good at making backup copies (the names of those files end in ~) and saving your files if the system crashes (the names of those files begin with #).

Emacs displays a certain number of characters per line until it gets to the right margin of the display. **Longer lines will wrap** to the next physical line; this is indicated by a \ at the end of each wrapped line. To avoid wrapping lines, either hit the Return key before you reach the right margin of the display, type **ESC q** to re-format a paragraph after finishing it (careful - if you don't have blank lines around the paragraph you might end up messing up your whole document - in that case do **CTRL-x u** (for 'undo')), or choose the **auto-fill** option (either as a default, in your .emacs file, or by typing **ESC x auto-fill-mode** (a toggle, i.e. doing the same thing again will turn it off.

Getting Help

You can invoke the general **emacs** help files by typing **CTRL-h CTRL-h** and following the instructions at the bottom of the screen.

CTRL-h a *FUNCTION*
    lists commands related to ***FUNCTION***
CTRL-h F *FUNCTION*

      get help about the specified function

CTRL-x 1

      get rid of Help window

ESC CTRL-v

      scroll Help window forward

CTRL-h i

      fancy emacs information browser

Error Recovery

CTRL-g

      cancel partially typed or accidental command

CTRL-l

      redraw garbaged screen

CTRL-x u

      undo the last change

ESC x recover-file

      recover lost file

ESC x revert-buffer

      restore buffer to original contents

Screen Motion

CTRL-v

      scroll to next screen

ESC v

      scroll to previous screen

Cursor Movement

CTRL-b

      go backwards one character without deleting (usually you can also use the cursor key for this)

CTRL-f

      go forward one character without deleting (usually you can also use the cursor key for this)

ESC b

      go backwards one word

ESC f

      go forward one word

CTRL-p

      go up one line (usually you can also use the cursor key for this)

CTRL-n

      go down one line (usually you can also use the cursor key for this)

CTRL-a

      go to beginning of the line

CTRL-e

      go to end of the line

ESC a

      go to end of the sentence

ESC e

go to end of the sentence

ESC <

go to the beginning of the file

ESC >

go to the end of the file

ESC x goto-line

go to the line specified

Buffers

The buffer is the basic editing unit. One buffer corresponds to one piece of text being edited. You can have several buffers open at once, but can edit only one at a time. Several buffers can be visible at the same time when you're splitting your window.

CTRL-x b

select another buffer

CTRL-x CTRL-b

list all buffers

CTRL-x 1

get rid of buffer list

CTRL-x k

kill a buffer

Multiple Windows

CTRL-x 0

dismiss this window

CTRL-x 1

dismiss all other windows

CTRL-x 2

split window in two horizontally

CTRL-x o

switch cursor to other window

Files

CTRL-x s

save the file

CTRL-x CTRL-w

save a file with a new name

CTRL-x CTRL-v

read a different file into emacs, replacing the current buffer

CTRL-x CTRL-f

read a different file into emacs and put it into another buffer

CTRL-x i

insert contents of another file into this buffer

Cutting and Pasting Text

In emacs, to "kill" means to delete something in a way that lets you recover it if you change your mind. DEL refers to the key on your keyboard that deletes backward one character--this may be the Backspace, Delete, or Rubout key depending on your particular keyboard.

DEL
      delete previous character
CTRL-d
      delete next character
ESC DEL
      kill previous word
ESC d
      kill next word
ESC 0 CTRL-k
      kill line to beginning
CTRL-k
      kill line to end
CTRL-x DEL
      kill sentence to beginning
ESC k
      kill sentence to end

To move or copy a region of text in emacs, you must first "mark" it, then kill or copy the marked text, move the cursor to the desired location, and restore the killed or copied text. A region of text is defined by marking one end of it, then moving the cursor to the other end.

CTRL-@
      set mark here
CTRL-x CTRL-x
      exchange cursor and mark
      (used to verify you have marked the desired region)
ESC h
      mark current paragraph
CTRL-x h
      mark entire buffer
CTRL-w
      kill the marked region
ESC w
      copy the marked region

All killed text (except single characters deleted with DEL or CTRL-d) can be restored to the buffer. The most recently killed text is the first to be restored, then previously killed text blocks are restored in reverse order.

CTRL-y
      re-insert ('yank') the last text that was killed
ESC y
      replace that restored text with previously killed text.
      Repeated use will cycle through killed text blocks in reverse order.

Searching and Replacing

In both forward and backward incremental searching, you can repeat the same command for next occurrences. The right cursor key exits the current search without moving the cursor. `CTRL-g` aborts the current search and moves the cursor back to the initial position.

`CTRL-s`
> incremental search forward.
> Note: On some terminals and serial connections, `CTRL-s` causes the display to freeze. If this happens, use `CTRL-q` to fix the problem, and use `ESC x search-forward` (`CTRL-x Esc` for further occurrences).

`CTRL-r`
> incremental search backward

`ESC %`
> (query-replace) - ask before replacing each OLD STRING with NEW STRING. Type `y` to replace this one and go to the next one, and type `n` to skip to next without replacing.

`ESC x replace-string`
> replace all occurrences of OLD STRING with NEW STRING.
> If you need to replace a `Return` character (^J), type CTRL-j instead (because typing `Return` is interpreted directly in the minibuffer).

`ESC-x list-matching-lines`
> lists all the lines matching your pattern in a separate buffer, along with their numbers. Use `ESC-x goto-line` to go to the occurrence you're interested in.

Keyboard Macros

Not everyone will agree these are 'basic', but they're extremely useful if you tend to do repetitive things and want to avoid getting Carpal Tunnel Syndrome...

`CTRL-x (`
> lets you start define a keyboard macro - basically records what you do, so do exactly what you want your macro to do (with some slight differences - e.g.\ in the new version of emacs on turing i-searches have to be terminated by `Return` in a macro, but not otherwise).

`CTRL-x )`
> end the definition of the keyboard macro

`CTRL-x e`
> executes the keyboard macro. You can also tell it how many times to execute the macro, for example if you want it executed 15 times, type `CTRL-u 15 CTRL-x e`. If you want your macro to be executed indefinitely, i.e until it runs out of things to do, use `CTRL-u 0 CTRL-x e`.

Here is a [Reference Card from DCG](), another [Reference Card](), and a comprehensive [Emacs Manual]()

[Back up to the Main Computing Page]()