

Physics 305
Introduction to Computational Physics
Fall 2009

The vi text editor

There are many reasonable choices for a text editor. Two that are powerful and universally available are “**vi**” and “**emacs**”. These notes will help you get started using the **vi** editor; if you prefer **emacs** go right ahead and use that instead.

The first thing you have to know about text editors is that the keyboard has to serve two purposes. Sometimes when you type something it is text that is to be entered into the file you are editing, and sometimes it is a command to do something to the file. For example, typing “**delete line 15**” might mean to enter the text “delete line 15” into the file (I just did that in making these notes), or it might mean that whatever is on line 15 of your file is no longer wanted. It isn’t really practical to have two keyboards, so a text editor must have some way to distinguish the two kinds of input. In the **vi** editor this is done by having two modes, the “command mode” and “insert mode”. They are just what they sound like. In the command mode, you are telling the computer to do something to the file, and in the “insert mode” you are entering text into the file. When you start the editor, you are in the command mode. One (of many) ways to get into the insert mode is to hit “i”. Then, what you type will be placed in the file, beginning at the location of the cursor. To get out of the insert mode and back into the command mode, hit the escape key, usually abbreviated *esc*.

Every frustrated beginner sooner or later ends up trying to type commands, and finding that everything just appears on the screen as you type it. This is probably because you are in the insert mode. The escape key is your friend. When you don’t understand what is happening, try hitting the escape key!

You will spend a lot of time editing files, so it is a good thing that the common commands have very short abbreviations. For example, you don’t really type “delete three lines”, instead you type “3dd”. This is annoying at first, since the abbreviations are not obvious and you will always be referring to your references. But after a day or so you will be doing things quickly, if not accurately.

Here is a really minimal set of commands, just to get you started. This is mostly just to convince you that this really isn’t that complicated. But you will soon want to know more commands so that you can do things more efficiently.

There are really only five things you need to know how to do: start the editor, stop the editor, move around in the file, add something and remove something.

1. To start the editor: “`vi filename`”. Here *filename* is the name of the file you want to edit. It can be a file that already exists that you want to change, or a new file that you want to create.
2. To stop the editor: “`:x`”. “x” stands for “exit”. Watch out for that colon! This writes the file you are editing to disk, and stops the editor.
3. To move around: Lots of ways, but start with the arrow keys: ↓, ↑, → and ←.
4. To add something: “i”, for “insert”, then type whatever you want to put in. Then hit *esc* to stop inserting.
5. To remove something: Lots of ways. start with “x”, which deletes the character at the cursor.

Now go try it. After you have edited a couple of files, continue to the next page where there are more commands that will let you do your work faster. They are all variations on the same five functions.

Actually, before proceeding, you may find that in your drunken stupor you have irretrievably mangled the file you were trying to fix. If this happens, and you want to get out of the editor without saving all the changes you have made, type “`:q!`”. Again, note the colon. Here “q” stands for “quit”, and the exclamation point tells the computer that you really do mean to quit without saving your changes.

Now here is a bigger table of commands. It still isn't all of them, but it is probably good enough.

- **Starting the editor**

Edit one file `vi filename`
Edit several files `vi file1 file 2 ...` (“:n” moves to next file)

- **Stopping the editor**

Save changes and quit `:x`
Quit if you haven't made changes `:q`
Discard changes and quit `:q!`

- **Moving around**

Up	<code>“↑”</code> or <code>“k”</code>	
Down	<code>“↓”</code> or <code>“j”</code>	
Right	<code>“→”</code> or <code>“l”</code>	
Left	<code>“←”</code> or <code>“h”</code>	(because “h,j,k,l” are right under your fingers.)
End of line	<code>“\$”</code>	
Beginning of next line	<code>return</code>	
Beginning of previous line	<code>“_”</code>	
Down 1/2 page	<code>ctrl-d</code>	(hold ctrl key, hit “d”)
Up 1/2 page	<code>ctrl-u</code>	(“up”)
Down full page	<code>ctrl-f</code>	(“forward”)
Up full page	<code>ctrl-b</code>	(“backward”)
Line number 77	<code>“:77”</code>	etc.
Last line	<code>“:\$”</code>	(“\$” means “end”)
Next occurrence of “pattern”	<code>“/pattern”</code>	<code>ret</code>
Previous occurrence of “pattern”	<code>“?pattern”</code>	<code>ret</code>
Repeat last search	<code>“n”</code>	(stands for “next”)

- **Adding stuff** (“esc” to end additions)

Add before cursor `i` (“insert”)
Add after cursor `a` (“append”)
Start new line below cursor `o` (“open”)

- **Removing stuff**

Delete one character `x`
Delete one word `dw`
Delete one Word `dW` includes punctuation
Delete one line `dd`
Delete four lines `4dd` etc.
Delete three words `3dw` etc.

- **Copy and paste**

Grab one line	Y	(“yank”)
Grab five lines	5Y	etc.
Paste last thing grabbed	p	(“put”)

- **Miscellaneous**

Undo the last command	u	USEFUL!
Repeat last modification	.	USEFUL
Built-in help	:help	(might help)

There is an online manual for the “improved” vi editor at <http://vimdoc.sourceforge.net>, and probably other places. There are several books on vi. One good one is the O’Reilly Book *Learning the vi editor*, by Lamb and Robbins. (The “O’Reilly books” are the ones with cute pictures of animals on the cover, and are generally very good.)