

## Chapter 3:

### Editing Files With vi

In this course, we are going to be writing a lot of computer programs, which are just files on your computer. To create and edit these files, we are going to be using a computer program called vi (pronounced “vee eye”). Think of vi as a very, very, very low-tech word processor. However, vi is *not* a word processor, as you will soon see for yourself.

To start vi, you need to know the name of the file that you are editing or creating. For example to work with a file called test.c, you would enter the following from the Unix command line:

```
vi test.c
```

(This is an excellent example of a difference between Unix and Windows. In Windows, you would start Microsoft Word, then click on File, then Open, and then select the file that you want. In Unix, you use a “verb-noun” paradigm— “vi test.c”, where vi is the verb and test.c is the noun.)

#### One BIG Thing to Know about Using vi

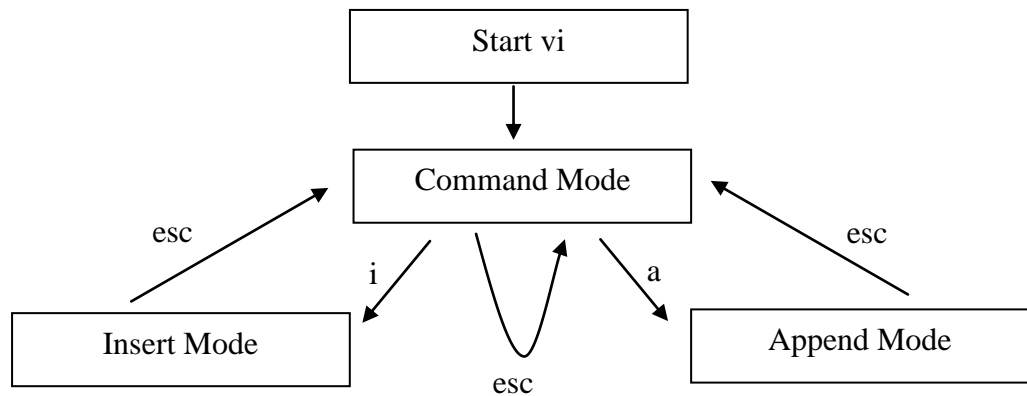
The program that we will use for editing files, vi, is *not* a word processor. Just typing the words that you want to see will not necessarily add those words to the file. Rather, vi needs to be in the right *mode*. However, it is not entirely obvious which mode vi is in at any given time.

In its “Command Mode”, you can manipulate text by deleting characters or lines, searching for text, saving the file, exiting, etc. However, you *cannot* “enter text” in the command mode. When you start vi, by default you are running vi in the Command Mode.

In its “Insert Mode”, you can type text and it will appear *at the location of the cursor on the screen*. You enter Insert Mode from Command Mode by pressing “i”. You exit Insert Mode and return to Command Mode by pressing “escape”.

In its “Append Mode”, you can type text and it will appear *after the location of the cursor on the screen*. (If it sounds like Insert and Append Modes are very similar, you are correct. Most of the time, just use the Insert Mode and you will be fine.) You enter Append Mode from Command Mode by pressing “a”. You exit Append Mode and return to Command Mode by pressing “escape”.

So here is a flowchart that shows how to move between the different modes of vi:



Therefore, when in doubt, it is usually best to hit the escape key and throw vi into "command" mode. If you previously were in the "append" or "insert" modes, vi will smoothly switch over to the "command" mode. If vi was already in the "command" mode, vi will beep at you, but there is no harm. Then, to go into "insert" mode, hit "i". Or, to go into "append" mode, hit "a".

### Ten Small Things to Know About Using vi

#### 1. saving

To save what you are working on, go into "command" mode by hitting escape. Then type:

```
:w
```

which stands for "write".

#### 2. saving and exiting vi

To save what you are working on and exit vi, go into "command" mode by hitting escape. Then type:

```
:wq
```

which stands for "write and quit".

#### 3. moving the cursor to some line in the file

Firstly, let me point out--in case you haven't already noticed--that there aren't any "scroll bars" in vi. Also, the mouse doesn't work in vi. Welcome to the 1960s, kids. However, the arrows keys do work, which is

nice. In some versions of vi, the page up and page down keys work, too-- if they don't, the f key (in command mode, obviously) is like page down and the b key is page up (for "forward" and "back").

Suppose you know that you want to go to line 300 in the file (maybe you know that there is an error on that line). To jump to that line in the file, first put vi into "command" mode by pressing escape. Then enter the number of the line that you want to move to, and follow that with a capital G (for "go", presumably).

Moving to the end of the file (a particularly common need, since that's generally where you add text) is easy. In vi, the code for "the end of the file" is the dollar sign. So, to move to the end of the file, type:

```
$G
```

#### 4. finding text

Suppose that you are looking for the word "test" in a file. Go into the "command" mode by pressing escape. Then, type a forward slash like this:

```
/
```

and enter the text that you are looking for, and press the enter key. At this point, vi will look for the text, starting at the location of the cursor.

To find more instances of the desired text, press the n key repeatedly.

#### 5. erasing a character

In command mode, erasing a single character in vi is accomplished using the x key. Position the cursor on the character that you want to erase, and press x.

#### 6. erasing a line

To erase a line, be sure that you are in the "command" mode by pressing the escape key. Then type:

```
dd
```

#### 7. repeating a command many times

To repeat a command, enter the number of times that you want to repeat a command before entering the command itself. For example to erase 5 lines, type:

```
5dd
```

#### 8. undoing the last change

If the last command that you entered was a mistake, you can undo it (inside "command" mode) using the u key.

#### 9. quitting, but not saving the file

Sometimes you just need to quit working on a file and don't want to save it as-is; in other words, you want to lose the changes. To do this, be sure that you are in the "command" mode by pressing the escape key. Then type:

```
:q!
```

which stands for "quit, damn it".

#### 10. matching parenthesis and braces

In programming, a common error is to have an opening parenthesis or brace, but not the closing one. You can use vi to help you solve this problem. If you aren't sure which closing parenthesis matches which opening parenthesis, position the cursor on one parenthesis and, in command mode, hit the percent key. The cursor will jump to the parenthesis that matches up. If it isn't the one you thought it would be, this helps you find the errors.

Additional resources about vi commands:

<http://www.eng.hawaii.edu/Tutor/vi.html>

<https://engineering.purdue.edu/ECN/Resources/KnowledgeBase/Docs/20020202121609>

<http://www.thomer.com/vi/vi.html#manuals>

[http://math.la.asu.edu/vi\\_tutorial/vicontents.html](http://math.la.asu.edu/vi_tutorial/vicontents.html)

## Assignment 1: Unix commands and editing with vi

### Part 1: Unix commands

Using the Unix commands learned in this course, complete the following tasks:

- Create a subdirectory called assignment1.
- Make a copy of the file called assignment1.txt from the ~schragej directory and put it in your home directory.
- Copy assignment1.txt into the assignment1 subdirectory.
- In your assignment1 subdirectory:
  - Rename the assignment1.txt file my.assignment1.txt.
  - Copy my.assignment1.txt to a file called test.data.
  - Remove the assignment1.txt file.

### Part 2: Editing text files in vi

Using vi commands learned in this course, open the test.data file, and complete the following tasks:

- Change the low temperature “tonight” to 25 degrees.
- On “Monday night”, change “Mostly cloudy and windy” to “Mostly cloudy with a chance of flurries”.
- Delete the blank line between each time period of the forecast.

### Part 3: Writing text files in vi

Write a brief description of what coursework or classroom experiences you have had that involved computers. These could be classes in high school or in college. Tell me what kinds of things you learned, the types of computers that you used, etc.

How to “Impress Me”: The first two parts of the assignment are very straightforward—don’t try to “impress me”, just get them done exactly as shown. For the third part, you can impress me by making your text “tidy”—white space between paragraphs, nice indentation, maybe an outline, etc.

This assignment is worth 5 points.