Stat 290: Lab 1

Introduction to Unix and Emacs

Lab Objectives

1. To introduce basic emacs commands
2. To introduce basic unix commands

Assignment

There is a trivial assignment at the end of the lab, to be handed in at class on Thursday, August 30. The main point is to make sure you can print, so that you don't have printing problems later on.

Notation

Recall that the notation `C-g` means to hold down the "Control" key and hit the "g" key, while `M-a` means to invoke the Meta key (either hold down the "Alt" key, or hit and release the "Esc" key, depending on your keyboard) and hit the "a" key.

Reminders

If something starts to go horribly wrong, use `C-g` to cancel the current command (infrequently, you may need to use it several times). You can also undo commands successively with either `C-x u` or `C-_` (that's Control-underscore). Recall that `C-x C-c` will quit emacs, and `C-z` will suspend it, shrinking the window into an icon. The help menus in emacs are triggered with `C-h` and are quite extensive.

Intro to Emacs

Start up Emacs by typing at the unix command prompt:

```
emacs &
```

Note that the ampersand runs the process in the background, so that you can continue to use the unix shell window while emacs is running. Now type the following into your emacs window:

```
The Mongolian steppe, where the Mongolian gerbil lives, is a biotope
with harsh, extreme circumstances.  Due to the fact that not many
animals can live under such circumstances, the gerbil has not many
natural enemies.
```

(Depending on how your defaults are set up, emacs may or may not automatically insert line breaks when you type too close to the right hand margin. If it does not, you can get it to "fix" your paragraph By typing `M-q`. The technical term for this is *auto-fill*.)

Cursor Movement

Now we'll walk through a variety of cursor movement commands. Try the following to get accustomed to moving the cursor without using the arrows or the mouse. Hit `M-<` to move to the beginning of the buffer. `C-f` moves you forward one character, `C-b` moves you back. `M-f` moves you forward one word, `M-b` moves you back one word. `C-n` moves you down to the next line, `C-p` moves you up to the previous line. `C-e` moves you to the end of the line, `C-a` moves you to the beginning of the line. If your document scrolled over more than one screen, you can use `C-v` to scroll down and `M-v` to scroll up.

Files and Buffers

Suppose you now want to save this unnamed file (or if it already had a name, supposed you wanted to rename it). To write a file to disk, use `C-x C-w` and then type in the name of the file and hit enter, as in:

```
C-x C-w gerbils.txt
```

Now we'll update the file. Add the following line to your file:

```
Gerbils are talented diggers.  They make very complicated subterranean
networks, including tunnels, storage rooms, and nestrooms.
```

Now save your updated file by typing `C-x C-s`. This is the standard way of saving your file. Emacs does have an auto-save feature, but you should get into the habit of saving your work very frequently as you work, just in case something bad should happen.

This next section will be part of the assignment to be turned in. Without closing emacs, open a new file (the previous file, `gerbils.txt,` will still be open, but in a different buffer which will be behind the current one) by typing

```
C-x C-f lab1hw.txt
```

Type in your name on the first line and your department on the second line. Leave the third line blank. On the fourth and following lines, tell me your experience with S-Plus, LaTeX, and C. Save this file with `C-x C-s`.

Since gerbils.txt is still in another buffer, we can still access it. To switch to another buffer, type in

```
C-x b
```

and it will ask you which buffer you wish to switch to. The default will be `gerbils.txt`, so you can just hit enter (otherwise you can type in the name of the buffer you want). In this way, you can have many files open in the same emacs process, and just switch between them as need. You can also have more than one displayed at the same time. Split the screen by typing

`C-x 2`

You will now have two buffers, each half the size of the emacs window, each containing the same file. Switch one of them to `lab1hw.txt` using `C-x b`. Now you have easy access to both files. To move the cursor between the buffers, use `C-x o`. To go back to a single window (just the window containing the cursor) use `C-x 1`. You can close a file with `C-x k` (this kills the buffer), but leave the gerbil file open, as we'll continue to use it in the next section.

Cut, Copy, and Paste

Bring `gerbils.txt` to the front of your emacs window (if it isn't already, use `C-x b` to switch buffers, or `C-x C-f` to open it from the disk if you've closed it). Suppose we want to move the two sentences about digging to the beginning of the file. Move your cursor to the first letter of the first sentence on digging and then type Control-space. This sets the "mark" at that character. Now move past the end of the second sentence and type `C-w`. This cuts the marked section. (If you miss, you can use `C-_` to undo.) If we just wanted to delete this part, we have done so and can just move on, but we want to move it, so we need to paste it back in. Move to the beginning of the document (`M-<` is the shortcut) and type in `C-y` (yank is the mnemonic) to paste in the text. Move to the end of the file (`M->`) and paste another copy with `C-y`. This is just to practice a different deletion technique. Move your cursor back to the beginning of the section that was just pasted in and use `C-k` to kill all the text on the same line to the right of the cursor. Use `C-k` repeatedly until all the freshly pasted text has been removed. Finally, you should know that `M-w` will copy text to the kill buffer (like `C-w`) but without deleting it, so that you can paste it elsewhere without removing the original. You can either save your work with `C-x C-s`, or kill all the recent changes by killing the buffer `C-x k` and entering `yes` when it asks for confirmation of closing without saving. We're done with emacs for this lab, so you can close it with `C-x C-c`.

Basic Unix Commands

Back in your unix shell window, type

`ls`

After hitting enter, you will get a list of all the files in the current directory. You may have started in your root directory (~), which probably isn't where you want these files. Make a new directory for this class by typing `mkdir 290`

Now move the two new files into this directory, first use

`mv gerbils.txt 290/.`

Note that the period at the end says to keep the name of the file the same. For the second file, use tab-completion so that you don't have to type in the whole file name, i.e., instead of typing in `mv lab1hw.txt 290/.`, type in (without hitting enter) `mv la` and hit the tab key so that unix will fill in the rest of the name (or as much as it can if there are multiple files that all start the same way), and it should fill in the rest of that file name, and then you can complete the line with `290/.` and then finally hit enter. Now that you've moved the files, change to this subdirectory with

`cd 290`

Typing `ls` should show that the only two files in this directory are the two that you just moved in. You can copy files with the syntax `cp originalname copyname`, for example

`cp gerbils.txt newgerbils.txt`

If you have an old, incorrect, or otherwise unnecessary file, delete it with `rm`, as in

`rm newgerbils.txt`

To go back to the next higher directory in the hierarchy, use

`cd ..`

If you do an `ls` here, you will likely find files like `gerbils.txt∼` or `gerbils.txt∼1`. These are old versions that emacs automatically keeps for you in case you later want to revert to an older version (e.g., if you accidentally erase a large section of your work and then accidentally save it, this way you can recover an earlier version). Most of the time, these files just take up a lot of space unnecessarily. In ISDS, we have a shortcut `pu` that will automatically clean up all of these old files in the current directory (you need to run it separately in each subdirectory). If you do not have this shortcut, you can delete them manually with

`rm gerbils.txt~*`

The asterisk is a wild card character.

Assignment

Print out the file `lab1hw.txt`. The print command in unix is `lpr`, but you typically need to specify which printer you want to use to print, which is done with the `-P` switch. For example, to print this file on the ISDS printer in room 222, make sure you are in the directory with `lab1hw.txt` and type

`lpr -P222 lab1hw.txt`

Turn in this printout in class.