

ESCI 324 Unix Commands

Below are some Basic commands and information about using Unix that you may find useful throughout the semester. Please note that I have simplified the Usage for a number of the commands to limit the information provided below to the scope of the course.

Shell command

tcsh – Usage: tcsh

The tcsh command will change the shell type of the terminal. The advantages of using the tcsh shell include that you will be able to use the arrow keys to move the cursor within a command line, use the up key to access recently used commands, use the tab key to complete long file or directory names without typing the entire name if unique.

Exploring Directories and Basic Commands

cd – Usage : cd *directory_name*

The cd command changes the current working directory.

cd : go to your home directory

cd .. : go up one directory

ls – Usage: ls (*directory_name or filename*)

The ls command will list the files in the current directory or the specified directory or filename. When specifying a directory or filename you may use wildcards such as the asterisk, *. An explanation of some wildcards is included in its own section below.

mkdir – Usage: mkdir *directory_name*

The mkdir creates a new directory with the user specified name.

cp – Usage: cp *file_to_copy new_file_name*

The cp command copies files. Note that if you use a directory as the new file name the command will copy the file to the directory using the original file name.

mv – Usage: mv *file1 file2*

The mv command moves file1 to file2. If file2 is a directory than file1 is moved to the directory named file2. If file2 is a filename, then Unix essential renames file1 to file2.

rm – Usage : rm *file1 file2 ...*

The rm command removes the files listed after the command. Unix will ask if you want to remove each file before deletion. If you want to suppress the safeguard, type \rm *file1 file2*. I would **not** recommend using this option as you can accidentally delete files that you need to use.

pwd – Usage: pwd

The pwd command simply returns the current directory.

man – Usage: man *Unix_command*

This command displays the manual page for any Unix command or GMT command.

chmod – Usage: chmod +rwx *filename*

The chmod changes permissions on a file. In the case of this lab, the chmod command will likely only be used as shown above to change shell scripts edited in a text editor to an executable shell.

more – Usage: more *filename*

The more command displays a text or ASCII file on the screen.

Some Scrolling commands:

Spacebar: Displays next screen of text

Ctrl B: Displays previous screen of text

Return: Displays next line of text

Ctrl D: Scrolls Down half a screen

cat – cat *file1 file2...*

Concatenates and displays text and ASCII files.

Note that cat *file1 file2* > *file3* will add *file2* to the end of *file1* and save the results to *file3*.

paste - paste *file1 file2...*

Merges corresponding lines of the listed files; this is analogous to adding columns.

tail – Usage: tail -line_number *filename*

Displays the end of a text or ASCII file. The command tail -15 quake.proj displays the last 15 lines of the file quake.proj. The default number of lines displayed is 10; the command tail quake.proj will display the last 10 lines of the file.

head – Usage: head -line_number *filename*

Displays the beginning of a text or ASCII file. The command behaves in the same manner as the tail command.

gv – Usage: gv *postscript_file*

This command displays a postscript file using ghostview.

kill – Usage: kill process_number

This command will stop a process running on Unix. In the case of this lab, the process number is the number displayed when the & is used to run a script in the background. For example:

```
squiggle.rice.edu% plot_qandv &
```

```
[1] 17049
```

```
squiggle.rice.edu%
```

The number 17049 is the process number.

Other

command line & - Run a process in the background so that you can input commands using the same terminal. Also see the kill command description.

Ctrl + C – kills a currently running process in a terminal. This will kill a process that is not running in the background of your terminal.

Wildcards – Wildcards are generally used when searching for files but do not know or do not wish to type the entire file name.

- * matches any number of characters in a filename including none: *.ps would find all postscript files in a directory

- ? matches a single character in a filename

- [] Usually used to find a range of files by name the brackets represent a single character: [A-B]*.BHZ would find all files that have A or B as the first letter and end with .BHZ.

- ~ expands to the name of your home directory

Languages and programs

sac – Usage : sac

This will start the SAC program on the terminal. SAC, Seismic Analysis Code, is a program used to analyze and display seismic data, usually earthquake waveforms, that is stored in sac format. The program is similar to Matlab in that it has its own set of pre-made functions that will read, write, process, and display data. There will be further explanation of these functions later in the semester. If you want to look ahead the SAC homepage is <http://www.iris.edu/manuals/sac/manual.html> .

awk – Awk is a very useful programming language that can reformat, find items with a specified property, check validity, perform basic math, etc on text or ASCII data that is organized in columns. This will also be explained later in the course as necessary. One good resource for those of you that are interested is the text, The AWK Programming Language by Alfred V Aho, Brian W Kernighan, and Peter J Weinberger. There is one copy of the book in Fondren Library.

matlab – Usage : matlab

When the command is written in a Unix terminal, Matlab will open on the Unix machine. Matlab is a tool to perform an abundance of mathematical and statistical analyses on data sets. Note that the manual page for any Matlab command can be reached by typing help followed by the command name