Duane's Incredibly Brief Intro to Unix

How to get help on unix:

man <command-name>	Get full description of command
man -k <keyword>	List commands mentioning keyword in title

Logging in and out:

logout	Terminate session
exit	Terminate current "shell"
ssh <remote host>	Login securely to a remote host

File manipulation:

emacs <file>	Edit a text file (see "cheat sheet")
mv <old> <new>	Rename/move <old> file to a <new> name
rm <file(s)>	Delete file(s) from system
cp <orig> <duplicate>	Copy <orig> to file named <duplicate>
sftp <remote host>	Secure batch file transfers between mach'ns
scp host:<orig> host:<dup>	Securely transfer files between machines
cat <file>	Display/catenate file contents to screen
more <file>	Display file, page by page (but: use less)
less <file>	Display file, page by page (avoid more)
head <file>	Display the first few lines of a file
tail <file>	Display the last few lines of a file
grep <pattern> <file(s)>	Search for/display pattern within file(s)
source <file>	Read commands from <file> (also: . <file>)
turnin -c <course#> <file>	Turn in a copy of file under CS course # (on upstairs unix boxes) e.g. turnin -c 237 x.c

Directory manipulation:

cd <directory>	Change focus of session to files in directory
ls	List files in current directory
mkdir <name>	Make a new subdirectory, called <name>
rmdir <name>	Remove an empty subdirectory

Printing & Mail:

enscript <file>	Print a pretty copy of file in unix lab
enscript -d lw-cs-217a <file>	Print a pretty copy of file to max lab
firefox <url>	Open browser (under X11) at url
acroread <acrobat file>	Read a ".pdf" file with Acrobat reader
gs <postscript file>	Read a ".ps" file with "ghostscript" reader
lynx	Text-only (read: fast) browser

C:

gcc <file.c>	Compile C program into "a.out" executable
gcc -o <executable> <file.c>	Compile C program into executable
gcc -g -c <file.c>	Compile C into debuggable object file <file.o>
gcc -o <exec> <f.o> <g.o>	Link several object files together into exec
gdb <executable>	Run executable under debugger control
gdb <executable> <core>	Resurrect core file with executable as model

Information about users and systems:

w	Who's on the system
top	What are top cpu processes
ps	List processes on this system
whoami	Who is logged in at this window
finger <user>	Get details on user (or user@host)
last <user>	List last time(s) user used this machine
uptime	Print stats on machine, also time since boot
sign	Change message on 312 sign

Web:

http://www.cs.williams.edu	CS home page
http://www.cs.williams.edu/~<yourname>	Your home page (put stuff in ~/www)
http://www.cs.williams.edu/cgi-bin/showtop10	Unix disk hog list
https://www.cs.williams.edu	Secure web-based email

CS Machines (9/07, see: http://www.cs.williams.edu/systems/comps/comps.html):
amerifax angus ayrshire barzona beefalo brahman brownswiss corriente
devon dexter droughtmaster durham galloway guernsey guzerat hereford
holstein jersey longhorn montbeliard pinzgauer remus senepol shaver

Duane's Ten Ways To Make Your Unix Life More Reasonable

0. Walk away from the machine. Don't waste your time in front of a machine if you're not making any progress. Print a listing and walk away. Make and take a friend with you. Life will be better if you reconsider the situation without the pressure of a computer.

1. Read the man pages.
Realize, if you haven't already, that you don't know everything. Learn. The world travels about 66,600 miles an hour about our Sun, and the Sun glides gracefully along its own path dragging us along. Hackers have no impact. None.

2. Learn the emacs keystrokes. It will save you when you have to use a system whose mouse is not working. Avoid the "arrow keys". Why?...

3. Use emacs keystrokes in the shell. Many cursor manipulation keystrokes from emacs recall history in the "bash" shell:
    ^P = previous command, ^N = next command,
    ^R = search for command from the past by typing a few letters
    ^A = go to beginning of command line
    ^E = go to end of command line
    ^B = go back one character
    ^F = go forward one character
    ^D = delete this character
    <del> = delete previous character

4. Learn about your environment. Shells like "bash" have survived evolution by helping their users do complex things. Type:
    man bash
Good things to keep an eye out for are "aliases" and "shell scripts". Other things to read about: find, tar, gawk, perl.

5. Forward your mail to one address. On unix machines you want mail forwarded from, create a file ".forward". This file should contain a single line with your ideal address. WARNING: don't forward mail back-and-forth between two systems.
Your email address with us is of the form 82dab@cs.williams.edu

6. Stay organized.
Create directories to organize your belongings. Delete temporary files that you no longer need. Besides taking up space, they add friction to your life.

7. Use the facilities we provide.
Using our labs allows us to help you if you have problems. They're also a good place to meet others that are suffering the same project. Leaving your room to do your work makes it a nicer place to return.

8. Practice. Yes, even more.

9. Write. Good writing is hard, and computer scientists write far too little real prose. A good, small place to start: comments on your code. Another place: write your name on everything you do. If it's really yours, copyright it (it's free)!

See?