

Linux Command Summary

Directory Navigation and Listing	
cd	change to home directory
cd ..	go up to parent directory
cd subdir	change to subdirectory <code>subdir</code>
ls	list content of current directory
ls -l	list content with details
ls -a	list content including hidden files

File Commands	
cp src dest	copy <code>src</code> file to <code>dest</code> file
cp -r sDir dDir	copy “recursively” <code>sDir</code> directory to <code>dDir</code> directory (copies subdirectories too)
mv src dest	move - renames <code>src</code> as <code>dest</code>
rm fileName	removes file <code>fileName</code> (like <code>core</code>)
rm -r dirName	removes directory recursively
rmdir dirName	removes empty <code>dirName</code>
mkdir dirName	makes directory called <code>dirName</code>
chmod 750 file1	change permission of <code>file1</code> by specifying a three digit octal # where digits are owner, group, world each octal digit in binary are: read (4), write (2), execute (1)
cat file1	display <code>file1</code> to screen
less file1	display <code>file1</code> with pagination (space - next page, q-exit, ↑, ↓- keys)
passwd	change password (8 char. min.: 1 upper, 1 lower, 1 digit, and 1 special)

Process Management	
ps	List processes with pid
top	Shows the real-time processes
kill -9 pid	Kills the process with <code>pid</code> #

Keyboard Shortcuts	
<tab>	Auto-complete partial file name
<Ctrl>+c	Kill current command/program
<Ctrl>+z	Sleep current program
<↑>	Recall previous command(s)
<Ctrl>+d exit	log-off and close terminal

“Programming” Tools	
nano file.cpp	Simple text-editor
emacs file.cpp	Better C/C++ editor
gcc file.cpp g++ file.cpp -o exeFile	C compiler: compile to <code>a.out</code> C++ compiler: compile to <code>a.out</code> Options: compile to <code>exeFile</code> instead
./a.out	execute program in current directory (“.”) called <code>a.out</code>
time exeFile	run <code>exeFile</code> and print timing when done
script out.txt	capture output to file <code>out.txt</code> <Ctrl>+d to end

1) Log-on to student.cs.uni.edu using a Telnet/ssh client (e.g., PuTTY: <http://www.chiark.greenend.org.uk/~sgtatham/putty/>)
(On a MAC you can probably use: `ssh userName@student.cs.uni.edu` in a terminal to log-on)

2) Your initial log-in is the same as your UNI CatID with initial password shown on the handout (On first log in, you’ll be forced to change the password)

3) For this activity I want you to:

- create and then move into a directory called `hw1` to store today’s files
- use an editor (emacs commands on back) to write a simple C++ program that prompts the user for their age, allows them to enter it, and outputs (*echoes*) it back for them. Use the file name `age.cpp`: `emacs age.cpp`
- compile the C++ to an executable file called `age` using: `g++ -o age age.cpp`
- when its working capture the interactive running of the program using: `script out.txt` to start the capture and `<Ctrl>+d` to end the capture, then run the program using `./age`
- display the contents of the `out.txt` to the screen using the `less` command: `less out.txt`

4) Use a secure ftp client (e.g., FileZilla: <https://filezilla-project.org>) to copy `hw1` to your local computer
(On a MAC you can probably use: `scp -r userName@student.cs.uni.edu:/hw1 localDir`)

5) On your local computer zip the `hw1` directory and submit as Homework #1 at:

<http://www.cs.uni.edu/~fienu/cs1160s14/homework/index.htm>

Emacs Reference

Starting Emacs

emacs filename

Command Keys Meaning

C-key Hold <Ctrl>-key down while hitting specified “key”

A-key Hold <Alt>-key down while hitting specified “key”

Exit Emacs and File Commands

C-x C-c Exit emacs-- answer ‘y’ to save work back to file

C-x C-s Save work back to disk without exiting

C-x C-w Write work to a file that you specify

Cursor Motion - arrow keys: ↑ ↓ ← →

C-a Move to beginning of the current line

C-e Move to end of the current line

C-v Move down a “page”

A-v Move up a “page”

A-< Move to beginning of the file

A-> Move to end of the file

A-g g # Go to specified line #

Editing - Cutting, pasting, search, replace

C-k Kill/cuts from cursor up to end of line. Puts line in kill buffer

C-y Yank/paste copy of kill buffer at cursor location

C-spacebar Begin marking at cursor location

C-w Wipe/delete from beginning mark to cursor location. Puts deleted text into kill buffer

A-w Copy without deleting from beginning mark to cursor location. Puts copied text into kill buffer

C-c C-c Comment out from beginning mark to cursor location

C-u C-c C-c Uncomment out marked region

C-s Interactive Search as you type search string at bottom of screen. Hit C-s again to repeat last search (wraps to top)

C-r Interactive Reverse Search (“upward” from cursor)

A-% Query replace. At bottom of screen type search-string<enter> followed by replacement-string<enter>. Highlights first occurs after cursor. Hit:

<spacebar> to replace and move to next match
! to replace all remaining matches without questions
<backspace> to skip to next occurrence
<Enter> to quit

Windows, Buffers, and Regions

C-x 2 Split window into two buffers vertically

C-x 3 Split window into two buffers horizontally

C-x o Move cursor to other buffer

C-x 1 Collapse window into one containing cursor

C-x c-f Open or create file in window containing cursor