OSU Center for Genome Research and Biocomputing (CGRB), MCB 599: "Introduction to Unix/Linux" (INX_U18)
"Cheat Sheet": Day 8, **Pages:** 2, **Revision:** 2018-08-06, **Instructor:** Matthew Peterson (matthew@cgrb.oregonstate.edu)

Command/Concept	Usage/Syntax	Description	Example
Standard Out	stdout	Most commands by default send their output to a data stream called "Standard Out" or "stdout" By default stdout displays its output on your terminal.	# Displays to stdout `echo hello`
`>`	`<cmd> > <filename>`	You can redirect a program's stdout to a file using >	# Stores stdout to a file `echo hello > hi.txt`
Standard Error	stderr	Commands can also send diagnostics or error messages to a secondary data stream (independent of stdout) called "Standard Error" or "stderr" By default stderr displays its output on your terminal.	See Chapter 8 "Standard Streams" in the textbook for examples and diagrams.
Capturing stderr to a file independent of stdout	`(<cmd> > stdout.txt) >& stderr.txt`	The workaround in tcsh is to first capture stdout via `(<cmd> > stdout.txt)` followed by capturing standard error via `>&`	# `/fake` does not exist and # generates a stderr message `(ls /fake /etc > stdout.txt) >& stderr.txt`
Capturing both stdout and stderr to the same file	`<cmd> >& stdout-and-stderr.txt`	In tcsh you can capture the output of both stdout and stderr to the same file.	# Both data streams `ls /fake /etc >& stdout-and-stderr.txt`
`grep`	`grep '<pattern>'`	Extract a pattern from a file (or data stream).	`grep 'needle' haystack.txt`
`grep –v`	`grep -v '<pattern>'`	Inverse grep, extract all lines that do not contain the `<pattern>`	# Everything but the # needle is extracted `grep -v 'needle' haystack.txt`

`wc`	`wc <filename>`	Counts lines, words, and characters from a text file (or data stream). Use the `-l` parameter to just output the number of lines.	# Outputs the number of
			# lines, words, and characters
			`wc /etc/motd`
			# Outputs number of lines
			`wc -l /etc/motd`

Standard In	stdin	Secondary input mechanism for programs (other than reading from a file). By default programs do not use stdin.	See Chapter 8 "Standard Streams" in the textbook for examples and diagrams.

Pipes	`|`	You can use pipes (the | character) to redirect the stdout of one program into the stdin of another.	# Count the number of lines
			# in the "Message of the day"
			# that contain: `jobs`
			`grep jobs /etc/motd`
			`| wc -l`

`cat`	`cat <filename>`	Concatenate files together to stdout. Can operate on just one file. Like `less` but it sends the entire contents of the file to stdout (your terminal) at once.	# Show "Message of the Day"
			`cat /etc/motd`
			# Merge 3 files into a single
			# stdout data stream
			`cat 1.txt 2.txt 3.txt`

Commands over multiple lines on the terminal	`\`	Instead of typing a set of commands on a single line you can separate them over multiple lines on the terminal using `\`	`cat /etc/motd \`
			`| grep jobs \`
			`| grep terminated \`
			`| wc -c`