

## Shell Commands (part 2)

### CS 3336

#### more

Display the contents of a file to the user one page at a time. Pressing space will let you go on to the next page. Usage: `more file`

#### less

Like more but better. Usage: `less file`

#### man

Provide online documentation for Unix commands, system calls, configuration files and other features. Usage: `man command_name`

- **-S *n*** Look for documentation in section *n* of the manual. Section 1 is for shell commands, section 2 is for system calls and section 3 is for other functions.
- **-k *words*** Look for man pages about the given keywords

#### head

Print the first ten lines of each file parameter (or from stdin if no parameter is given). Usage: `head file`

- **-n *num*** Print the first *num* lines of the file.

#### tail

Print the last ten lines of each file parameter (or from stdin if no parameter is given). Usage: `tail file`

- **-n *num*** Print the last *num* lines of the file.

#### touch

Bring the modification time of a file up to the current time. Also, create an empty file if it doesn't already exist. Usage: `touch file`

#### ps

Show currently running processes.

- **-e** Report on every process.
- **-H** Give hierarchical listing of parent/child processes.
- **-l** Give a long listing.

#### top

Like ps, but give a continuously updating report.

#### jobs

Show all processes that your shell is keeping up with along with their job numbers (shell builtin).

#### bg

Move a suspended process into the background (shell builtin). Usage: `bg job_number`

#### fg

Move a backgrounded or suspended job to the foreground (shell builtin). Usage: `fg job_number`

#### kill

Send a signal to a process asking it to terminate. Usage: `kill process_id`

#### killall

Kill all processes running a given command. Usage: `killall command_name`