How do I do _____ in CAEN?

The use of CAEN Linux in EECS281 is all-but-required, as we strongly recommend you test and debug (if not write) your code on CAEN before submission. Here are a few of our favorite Linux command-line tools that may be useful to you in EECS281 and beyond:

- `cat [filename1] [filename2]...`
  - Concatenates its arguments and prints the result to standard output
- `head [filename]`
  - Prints the first 10 lines of a file to standard output
  - `tail` does the same, but with the last 10 lines
- `man [command]`
  - Read the manual pages for a command to learn about it
- `file [filename]`
  - Tells you the type of a file
- `touch [filename]`
  - Creates a file, or updates the time it was last accessed if it already exists.
  - `mkdir [dirname]` will make a new directory
- `rm [filename]`
  - Remove (delete) files. The -r flag makes rm act recursively, so you can use it to delete a directory.
  - The -f flag forces rm to ignore warnings.
  - Only use -f with extreme caution - this holds doubly for using -f with -r, and triply for using it with `sudo` or `--no-preserve-root`. Being careless can lead to permanent data loss or damage to your system!
- `whereis [command]`
  - Locates binaries, headers, and manual pages for a command
  - Example: find if your computer has the getopt.h header by running `whereis getopt` and seeing if any results are the include directory
- `ps -e -f`
  - Lists currently running processes on your computer
  - Better version of `ps -A`
- `locate [filename]`
  - Search for a file on your computer
- `< filename`
  - Redirect standard input from filename
- `> filename`
  - Redirect standard output to filename
  - If the file does not exist, it will be created. If it does exist, it will be overwritten.
- `>> filename`
  - Redirect standard output to filename, appending instead of overwriting

    - Example: `echo "Make sure to pass ZombieClass by reference!" >> notes.txt` will add a line to the end of your file
- `2> filename`
    - Redirect standard error to filename
- `pwd`
    - Print current working directory
- `cd [directory]`
    - Change current working directory to the one specified
    - Addresses can be relative to the current directory or absolute (beginning with / or ~/)
    - All directories have the self (.) and parent (..) directories
    - / is the root directory, ~/ is your home directory
- `ls [directory]`
    - List files in a directory - if no argument is given, list files in current directory
    - `ls -l`
        - List in a more verbose format, showing file owners, permissions, time last modified, etc.
    - `ls -A`
        - List hidden files along with regular files
- `reset`
    - Clear your terminal - saves you from having to find which compiler errors are from your most recent attempt at compiling
- `dos2unix [filename]`
    - Convert a file with DOS (Windows) line endings to Unix line endings
    - `unix2dos` does the opposite
- `grep [expression]`
    - Searches standard input for lines containing the given regular expression
    - Often used to search for a keyword in the output of another program - see pipes
- `|`
    - Pipe - redirect standard output from one command to the standard input of another
    - `ls | grep cpp`
        - List all files with "cpp" in their names
- `*`
    - Globbing - select all files matching a certain pattern
    - Example: `g++ *.cpp -o lazy` will compile all .cpp files in your directory
    - Beware - * has a very different meaning in regular expressions, so be careful when using it with grep!
- `cppcheck *.cpp *.h`
    - Run the cppcheck static analysis tool on all your source files
    - This is the same tool the autograder uses to give you feedback on your style
    - Ok, this one isn't on CAEN - it can be easily downloaded through apt though!
- `tar -t -f fullsubmit.tar.gz`

    - This command will list the files in your submission tarball
    - Use this to make sure that your submission has the files you need
- `diff file1 file2`
    - Checks for differences between files. Use this to check that your output matches our sample output!
    - Use Kompare (available on CAEN Linux, can easily be downloaded through apt) instead of diff for a more graphical view of differences between files.
- `chmod +x myscript.sh`
    - Use this to make your own scripts executable
- Keyboard tools
    - Pressing tab will autocomplete or give a list of possible autocompletions for files or directories
    - `![expression]` will run the last command beginning with [expression]. Use with caution - the command will run without asking for confirmation!
    - Ctrl-C will interrupt your program. This will usually cause it to end. (GDB can catch a SIGINT.)
    - Ctrl-D gives the end-of-file (EOF) delimiter.
    - In the terminal, to copy/paste use Ctrl-Shift-C/V