Final Exam Review

Question 1

awk

- Description: awk is a scripting language used for processing and displaying text
- Formula:
  - awk + options + {awk command} + file + file to save (optional)
- Examples:
  - convert the first field to upper/lower case
  - awk -F: '{print toupper($1)}' /etc/passwd
  - print the first and the last field of the /etc/passwd
  - awk -F: '{print $1,"=",$NF}' /etc/passwd
  - print the first and 3 field with line numbers
  - awk -F: '{print NR, $1,$3}' /etc/passwd

- cat

- Description: cat command is used for displaying the content of a file
- Formula:
  - cat + option + file(s) to display
- Examples:
  - display the content of a file with line numbers:
  - cat -n ~/Documents/todo.md
  - display the content of a file with line numbers excluding empty lines:
  - cat -b ~/Documents/todo.md
  - display the content of a file suppressing repeating empty lines to single empty line:
  - cat -s ~/Documents/todo.md

- cp

- Description: cp copies files/directories from a source to a destination
- Formula:
  - cp + files to copy + destination
  - cp -r + directory to copy + destination
- Examples:
  - copy the content of a directory to another directory:
  - cp Downloads/wallpapers/* ~/Pictures/
  - copy multiple files in a single command:
  - sudo cp-r script.sh program.py home.html assets/ /var/www/html/
  - copy a directory with absolute path:
  - cp -r ~/Downloads/wallpapers ~/Pictures/

- cut

- Description: cut is used to extract a specific section of a line of a file and display it to the screen

- Formula:
    - cut + option + file(s)
- Examples:
    - display a list of all the user in your system:
    - cut -d ':' -f1 /etc/passwd
    - cut a range a bytes per line:
    - cut -b 1-5 usernames.txt
    - cut a file using a delimiter but changing the delimiter in the output:
    - cut -d ":" -f1,7 --output-delimiter=" => " /etc/passwd

- grep

- Description: grep is used to search text in given file, works line by line basis:
- Formula:
    - grep + option + search criteria + files(s)
- Examples:
    - search any line that contains the word "dracula" regardless of the case:
    - grep -i "dracula" ~/Documents/Books/dracula.txt
    - search for all lines that did not contain the word "war":
    - grep -v "war" ~/Documents/Books/war-and-peace.txt
    - search for all lines that start with a capitol letter:
    - grep -n '^[A-Z]' ~/Documents/Books/war-and-peace.txt

  head

  Description: displays the top N number of lines of a given file, by default it prints the first 10 lines
    - Formula: head + option + file(s)
- Examples:
    - display the first 10 lines of a file:
    - head ~/Documents/Books/dracula.txt
    - display the first 5 lines of a file
    - head -5 ~/Documents/Books/dracula.txt

  ls

  Description: ls is used for displaying all the files inside a given directory
- Formula:
    - ls + option + direcectory to list
- Examples:
    - long list all the files inside given directory recursively
    - ls -1R ~/Pictures
    - list all the files in a given directory sorted by extension
    - ls -X ~/Documents
    - list all the files in a given directory sorted by file size
    - ls -T ~/Documents

man

Description: manual pages are documentation files that describe linux shell commands, executable programs and system calls

- Formula:
  - man + command
- Examples:
  - open the man page of the passwd command: man passwd
  - show all the available pages of a command: man -a passwd
  - searches for man page given word expression or phrase: man -k file

mkdir

Description: mkdir is used for creating a single directory or multiple directories

- Formula:
  - mkdir + the name of the directory
- Examples:
  - create a directory with a parent directory at the same time: mkdir -p wallpapers_others/movies
  - create a directory with a space in the name: mkdir wallpapers/new\ car
  - create multiple directories: mkdir wallpapers/cars wallpapers/cities wallpapers/forest

mv

Description: mv moves and renames directories

- Formula: *mv + source + destination
  - mv + file/directory to name + new name (rename)
- Examples:
  - move multiple directories + files to a different directory: mv games/ wallpapers/ rockmusic/ /media/student/flashdrive (move 3 files to flashdrive)
  - rename a file using absolute path: mv ~/Downloads/homework.docx ~/Downloads/cis106homework.docx
  - move and rename a file in one single command: mv Downloads/cis106homework.docx Documents/new_cis106homework.docx

tac

Description: tac is used for displaying the content of a file in reverse

- Formula:
  - tac + option + file(s) to display
- Examples:
  - display the content of a file located in the pwd: tac todo.md
  - display the content of a file using absolute value: tac ~/Documents/todo.md

tail

Description: tail displays the last N number of lines of a given file, by default it prints the last 10 lines

- Formula:

    - tail + option + file
- Examples:
    - display the last 10 lines of a file: tail ~/Documents/Books/dracula.txt
    - display the last 5 lines of a file: tail -5 ~/Documents/Books/dracula.txt

touch

Description: touch is used for creating files * Formula: touch + name of the directory

- Examples:
    - create a file with a space in its name: touch "'list of foods.txt'"
    - create several files: touch list_of_cars.txt script.py names.csv
    - create a file using absolute path: touch ~/Downloads/games.txt

tr

Description: tr is used for translating or deleting characters from a standard output

- Formula:
    - standard output | tr + option + set + set
- Examples:
    - translate one character to another: cat file.txt | tr '.' ','
    - translate white space into tabs: cat program.py | tr "[..space..]" '\t'
    - translate tabs into space: cat file.py | tr -s "[..space..]" ' '

tree

Description: tree list all the files and directories in a given directory in a nice tree like format * Formula: tree + the name of the directory

- Examples:
    - long list of all the files inside a given directory recursively: tree -1R ~/Pictures
    - list all the files in a given directory: tree -X ~/Documents
    - list all the files given a directory sorted by file size: tree -t ~/Documents

vim/nano

Description: vim stands for "Vi improved" which is a command-line text editor

- Formula:
    - vim (nano) + option
- Examples:
- How to install vim: sudo atp install vim -y
    - To quit vim: q!
    - to start it: vim

Question 2

Answer each question:

- How to work with multiple terminals open?

    - open one terminal and then another and set them side by side or use Tillix and split the terminal as needed *How to work with manual pages?
    - to view the manual of a command type man + command
    - to navigate the man page, you can use the arrow key or the man command internal shortcuts
    - to exit the manpage press the letter "q"
- How to parse (search) for specific words in the manual page?
    - searches for a man page for given word or regular expression or phrase: man -k file
- How to redirect output (> and |):
    - the pipe allows you to redirect the standard output of a command to the standard input of a file
- Example:
    - use grep to use for a string in a particular man page: man ls | grep "human-readable"
    - display only the options of any command from its man page: man ls | grep "^[[:space:]]* [[:punct:]]"
- How to append the output of a command to a file: *append means to add more to a file instead of overwriting its content: ls -la > allmyfiles.lst
- How to use wildcards: for copying and moving multiple files at the same time: use ls to find out what files are needed use mv to move or cp to copy multiple files at the same time
- Formula:
    - mv(cp) + what you are moving plus destination
    - Example: ls lab6/*.log mkdir lab6/log-files mvlab6/*.log lab6/log-files/
- How to use brace expansions
    - for creating entire directory structures in a single command
    - mkdir -p music/{jazz,rock}/{mp3files,video,oggfiles}/new