WIIT 7780 Command Line and Files Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Step #1**

Do a ***whatis*** on the following commands and write down the definition of the command:

**mkdir -** make directories

**rmdir-** remove empty directories

**touch -** change file timestamps

*Can also create an empty file*

**cp -** copy files and directories

**mv-** move (rename) files

**rm -** remove files or directories

**pwd -** print name of current/working directory

**ls -** list directory contents

**Step # 2**

• Do an **info** on **ls**. Skim through the data. [Ctrl C to escape]

• Do a **man** on **ls** and review the options

• Issue a **ls** command

• Issue **ls –a** command. What is different from the previous ls command? What type of file is now showing? Shows all files and folders, shows all the files including hidden ones starting with .

• Issue **ls** command with at least 3 options, what did you choose as the options? -ald

• What happened? Listed only one directory

• Issue a **history** command. What is listed at line #6? whatis mv

• At the prompt, type **hi** then press the tab key. What happened? Listed hibagent history

• At the prompt, press the up arrow 3 times. What happened? displayed all the commands I issued

**Step # 3** - Navigation and commands: Provide the definition or describe what the command does.

• Absolute path = the path to a file or directory starting from the root

• Relative path = Path relative to where you are

**• cd** = change directory

**• ..** = go back one directory

**• /**  = root directory

**• cd /** = change directory to the root directory

**• mkdir** = create a directory

**Step # 4** - In MS-DOS we used HELP, in Linux we have the ‘**man**’ command for manual pages.

Use the **man** commands discover what the following utilities and their switches do:

**Command** **Switch** **Description**

Bash -r If the **-r** option is present, the shell becomes restricted

chmod -v **--verbose** output a diagnostic for every file processed

cp -l **--link** hard link files instead of copying

grep -c **--count**  Suppress normal output; instead print a count of matching lines

for each input file. With the **-v**, **--invert-match** option (see

below), count non-matching lines.

ls -d **--directory** list directories themselves, not their contents

man -k **-k** printf Search the short descriptions and manual page names for the keyword

printf as regular expression. Print out any matches. Equivalent

to **apropos** printf**.**

mkdir -m **--mode**=MODE set file mode (as in chmod), not a=rwx - umask

ps -r Restrict the selection to only running processes.

sort -r **--reverse**  reverse the result of comparisons

wc -L **--max-line-length** print the maximum display width

Now that you learn the man command, let’s take a few minutes and find resources on the Internet. List five Internet sites that provides help with Linux commands. Along with each site, tell what you like or do not like about it. Sort the sites from most helpful to least helpful.

**Site URL**

**What you liked or disliked**

**https://www.google.com/search?q=linux+commands&oq=linux+commands&aqs=chrome..69i57.416748j0j7&sourceid=chrome&ie=UTF-8**

**Google search give you at the command right awayΩ**

**https://maker.pro/education/basic-linux-commands-for-beginners**

**Great for beginners**

**https://searchdatacenter.techtarget.com/tutorial/77-Linux-commands-and-utilities-youll-actually-use**

**Great for the seasoned user, not very easy to use**

**http://www.linuxdevcenter.com/cmd/**

**Same as above .. more user friendly. But slow**

**http://www.dummies.com/computers/operating-systems/linux/common-linux-commands/**

**Great for beginners**

• Type **whoami** and press Enter to display your current username. What is it? Gimeia1

• Here's an interesting little trick: Type **!!** and press Enter. You should see that the Bash shell "re-issued" the command from the step above. The !! command "pulls out" the last command you did from your command history, displays the command, and runs it again.

• To see all your command history (commands you have previously entered in the bash shell), type **history** and press Enter. You should see several commands listed along with a number by each one.

• Pick one of the commands from the history list and note its number. Type   
 **!*#*** (where *#* is the number of the command in your history) and press Enter. The command should display and execute!

• What command did you pick? ps

• History is another method, besides the up-arrow key, you can use to recall previous commands.

**Step # 5** - Create the following structure in the home directory:

Refer to previous Steps 1 + 3 for reference, or **man, info** or **whatis**. Do a **tree** to check as you go

***Note: To enable it on other Debian distros: sudo apt-get install tree***

• Start by creating a directory in the home directory in which to experiment.

• Do a **pwd** and make sure you are in **/home/pi**

• Make a directory called ***YourInitials\_fun***Example: dkd\_fun

• In that directory, create a series of files and directories outlined below

• Make a directory called **Ying**

• Make a directory called **Yang**

• Navigate to the **Ying** Directory

• Create an empty file called **Mind**

• Create an empty file called **Full**

• Navigate to the **Yang** Directory

• Create a Directory called **Zen**

• From the Yang Directory create a file in **Zen** called **Pooh**

• Navigate to your ***YourInitials\_fun*** Directory and copy Pooh from the **Ying** Directory to ***YourInitials\_fun***

• Write out the command you used cp Yang/Zen/Pooh .

• In the **Yang** Directory, create 2 empty files, **Robin** and **Rabbit**

• Copy **Rabbit** to the **Zen** Directory.

• Write out the command you used cp Rabbit Zen

• Remove the **Rabbit** file in the **Yang** Directory.

• Write out the command you used rm Rabbit

• Do a **tree** on ***YourInitials\_fun*** for Instructor Sign off.

Instructor Sign Off:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_