CSC 2903: DevOps

Lab 01 – CLI I

General Instructions

Use your knowledge of Unix, previous course material, and reference material to fill out each question on this lab. Type the answer to your questions in **blue**.

Submission Instructions

To submit, **change the name in the header** and save this document as a PDF. Attach your PDF document to the iLearn dropbox.

Lab Questions

CLI Basics

1. (1) Briefly, what information does the `--help` option display for the tar utility? How would you display this information one screen at a time?

It shows a man page for the ‘tar’ command, to show one screen at a time $tar –help | less

2. (1) How would you display a list of utilities that compress files? (*hint*: look up the `apropos` command.)

```
$apropos compress
```

3. (1) How would you find out which Linux utilities create and work with archive files?

```
$apropos archive
```

4. (2) What happens when you give the following commands if the file named done already exists?
   a. `$ cp to_do done`

Replaces ‘done’ contents to ‘to_do’ contents.

   b. `$ mv to_do done`

Changes file ‘to_do’ to ‘done’ leaving only ‘done’ left.

5. (5) Is each of the following an absolute pathname, a relative pathname, or a simple filename?
   a. `Milk_co`

Simple filename

   b. `correspond/business/milk_co`

Relative Pathname

   c. `/home/max`

Absolute pathname

   d. `/home/max/literature/promo`

Absolute Pathname

   e. `..`

Relative Pathname

6. (1) The `echo` builtin copies its arguments to standard output which, by default, bash directs to the screen. Write the command to redirect standard output (Sobell, page 138) of `echo` to write a short message (e.g., “Hi there”) to a file and then use `cat` to display the contents of the file.

```
$echo hi there > test
$cat test
```

Basic File Manipulation

7. (4) Do the following:
   a. Write the command to redirect standard output of `cat` to create a file named `days` that holds the names of the days of the week in chronological order, one per line. Do not redirect standard input to `cat`; it will come from the keyboard. Remember to press CTRL-D on a line by itself to exit from `cat`.

```
$cat > days
Monday
…
Sunday
```

   b. Use `cat` to read the days file and send it to standard output, through a pipeline, to standard input of the sort (Sobell, pages 58 and 145) utility. The result will be a list of days in alphabetical order.

```
$cat days | sort
```

c. Replace sort in the preceding command with `grep` (Sobell, page 56) with an argument of (uppercase) T. The result will be a list of days that have an uppercase T in their names in chronological order.

$cat days | grep T

d. Create a filter (Sobell, page 146) by repeating the preceding command but sending standard output of `grep` through a pipeline to standard input of sort. The result will be a list of days that have an uppercase T in their names in alphabetical order.

$cat days | grep T | sort

Basic Vim Commands and Scripting

- Install Vim
  - `$ sudo apt update`
    - this goes and checks for updates for your system.
  - `$ sudo apt upgrade`
    - this will download and install the updates found in step 1.a.
  - `$ sudo apt install neovim`
    - ‘neovim’ is NeoVim, a newer slimmer package of ‘vim’. These two packages are essentially the same and can be used interchangeably.
- Create your first file
  - Type ‘`nvim showShells.sh`’
  - Tap ‘`i`’ on your keyboard to enter *input* mode.
  - Type the following into the file:

```
#!/bin/bash
echo "You are running $(uname -s) version $(uname -r)"
for x in ash bash bsh csh pdksh ksh sh tcsh zsh; do
    test -x /bin/$x && shells="$shells $x"
done
echo "You have at least the following shells installed:$shells"
```

- Press the escape key to enter command mode.
- Move your cursor to the word "at" in the second echo statement. In order to do this, you should use the arrow keys, or the following: "`l`" moves your cursor right, "`h`" moves your cursor left, "`k`" moves your cursor up, and "`j`" moves down. Practice moving around until you get to the first "at" in this script.
- Is your cursor placed before the word? Place it after the word by hitting the "`w`" key. Now, hit the "`b`" key. Pretty neat, isn’t it? Play around for a minute and then bring yourself back to the beginning of the word "at."
- Now, press "`dd`". We erased the whole line! Type "`u`". It is fixed!
- Make sure you are located right at the word "at" and press "`dw`". This should delete the word. Press the period key on your keyboard to repeat the previous command.

- Now, let’s save the document. Press the ESC key to bring yourself back to command mode. Then type “`:wq`" and hit enter to save and quit.
- Enter the following command: `chmod u+x showShells.sh`
- Now run your new shell script by entering the following command: `./showShells.sh`

8. (1) What does the script do?

It shows which Linux version I have installed and what shells I have installed.

9. (3) What do the following do?
   - `chmod`

It sets permissions of files and directories.

   - `echo`

It prints out the argument to the screen

   - `test`

It checks file types and compares values

10. (2) What are the primary 2 modes of Vim?

Command and Insert

11. (1) Why did we use `sudo` to install neovim?

We need the “Super User Do” to install and make changes to the system for neovim.

12. (8) Write a shell script called `aboutme.sh` that prints your name, rank (freshman, sophomore, junior, senior), TN Tech email address, and a short bio of you on the terminal screen. Make sure to move the file to your host machine when you are done so that you can include it in iLearn.