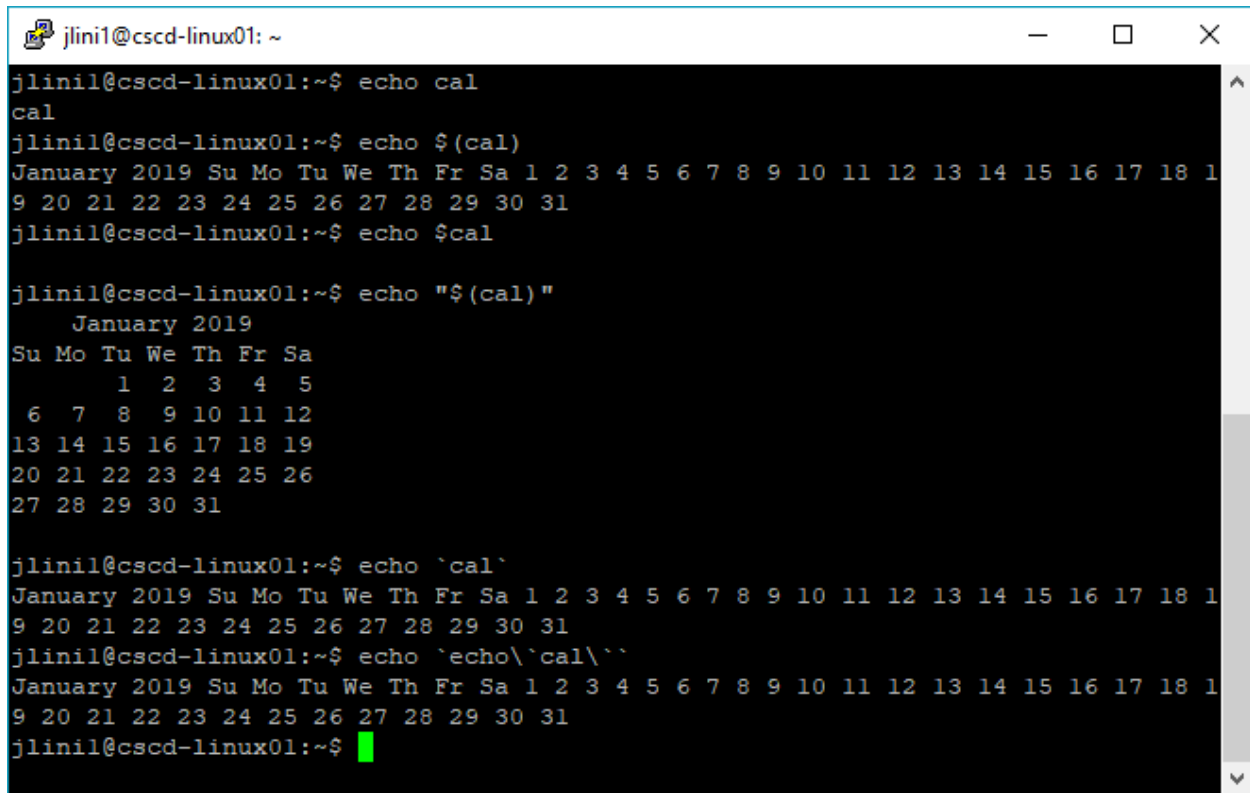


Lab 2: More Unix Commands

‘echo’

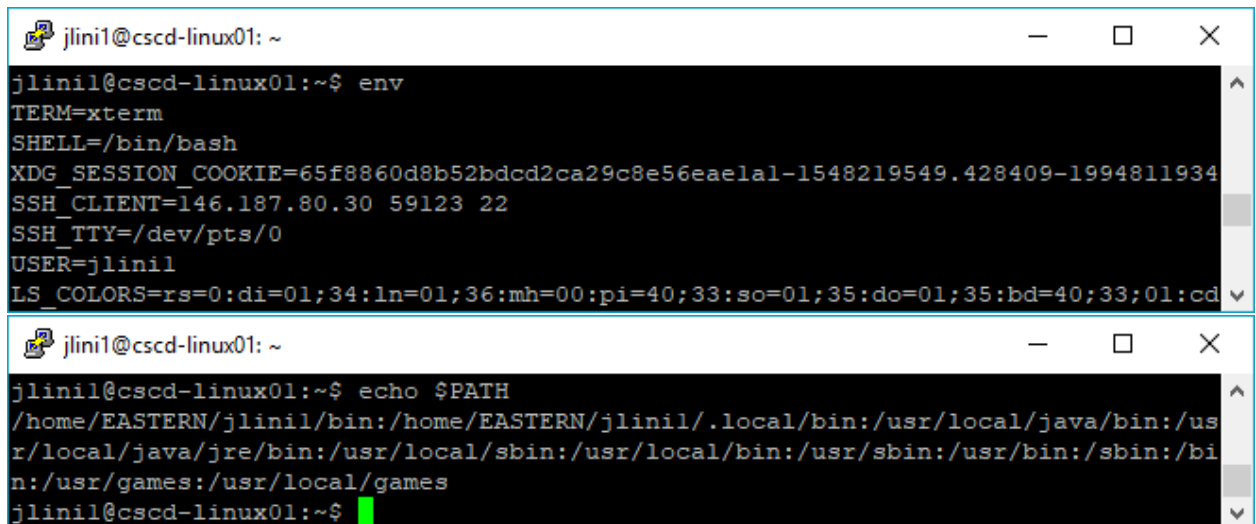
1. What are the differences among the following commands? Explain with screenshots.
(6 pts , 1 point each)
 - a) `echo cal` – **echo sees cal as a literal message to output**
 - b) `echo $(cal)` – **the parenthesis makes it so it runs the cal command**
 - c) `echo $cal` – **without parenthesis, it's printing out a variable called cal, which is not initialized**
 - d) `echo "$(cal)"` – **the quotes makes echo output the newline statements it seems**
 - e) `echo `cal`` ← This is back quote – **the back quotes run the command**
 - f) `echo `echo `cal``` ← This is back quote – **same as above but also echo's an empty echo command**



```
jlinil@cscd-linux01: ~  
jlinil@cscd-linux01:~$ echo cal  
cal  
jlinil@cscd-linux01:~$ echo $(cal)  
January 2019 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 1  
9 20 21 22 23 24 25 26 27 28 29 30 31  
jlinil@cscd-linux01:~$ echo $cal  
  
jlinil@cscd-linux01:~$ echo "$(cal)"  
January 2019  
Su Mo Tu We Th Fr Sa  
1 2 3 4 5  
6 7 8 9 10 11 12  
13 14 15 16 17 18 19  
20 21 22 23 24 25 26  
27 28 29 30 31  
  
jlinil@cscd-linux01:~$ echo `cal`  
January 2019 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 1  
9 20 21 22 23 24 25 26 27 28 29 30 31  
jlinil@cscd-linux01:~$ echo `echo `cal``  
January 2019 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 1  
9 20 21 22 23 24 25 26 27 28 29 30 31  
jlinil@cscd-linux01:~$
```

Environment variables

2. What command will show you all the environment variables? What command will display the environment variable named **PATH**? Show both with screenshot.
(2 points, 1 point each)



```
jlinil@cscd-linux01: ~  
jlinil@cscd-linux01:~$ env  
TERM=xterm  
SHELL=/bin/bash  
XDG_SESSION_COOKIE=65f8860d8b52bdcd2ca29c8e56eae1a1-1548219549.428409-1994811934  
SSH_CLIENT=146.187.80.30 59123 22  
SSH_TTY=/dev/pts/0  
USER=jlinil  
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd
```

```
jlinil@cscd-linux01: ~  
jlinil@cscd-linux01:~$ echo $PATH  
/home/EASTERN/jlinil/bin:/home/EASTERN/jlinil/.local/bin:/usr/local/java/bin:/usr/local/java/jre/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games  
jlinil@cscd-linux01:~$
```

I/O

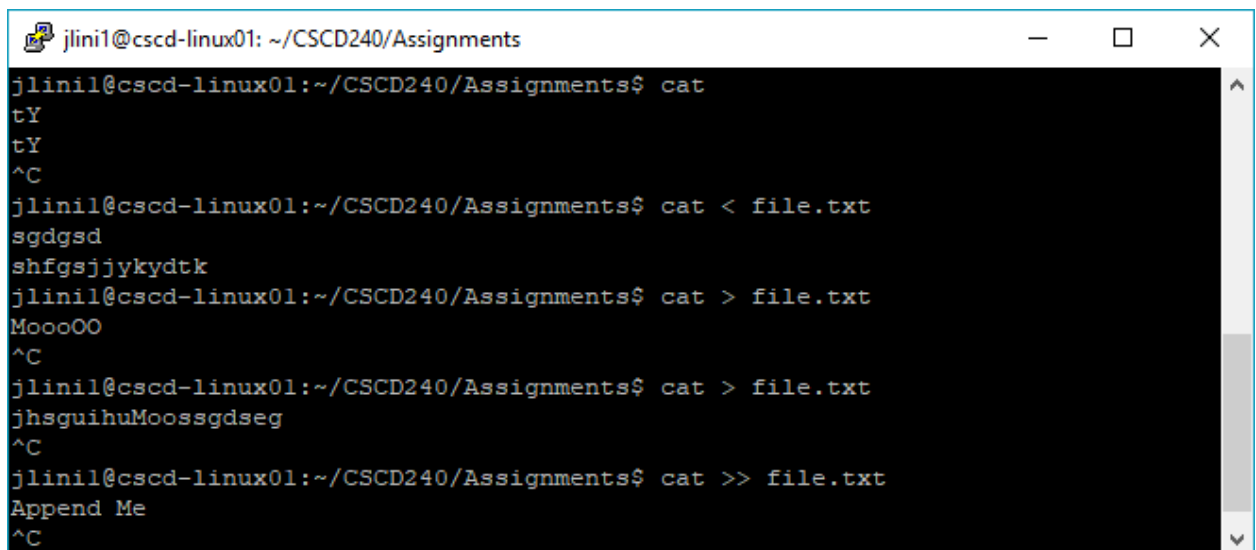
3. What are the differences among the following commands. Explain with examples and screenshot. (4 points, 1 point each)

cat – This will prompt for input and then repeat it after hitting ‘Enter’

cat < filename – cat outputs the contents of the file “filename”

cat > filename – cat outputs what you type in, into “filename”, overwriting it

cat >> filename – same as above, but appends instead of overwrites



```
jlinil@cscd-linux01: ~/CSCD240/Assignments  
jlinil@cscd-linux01:~/CSCD240/Assignments$ cat  
tY  
tY  
^C  
jlinil@cscd-linux01:~/CSCD240/Assignments$ cat < file.txt  
sgdgsd  
shfgsjjykydtk  
jlinil@cscd-linux01:~/CSCD240/Assignments$ cat > file.txt  
MoooOO  
^C  
jlinil@cscd-linux01:~/CSCD240/Assignments$ cat > file.txt  
jhsguihuMoossdseg  
^C  
jlinil@cscd-linux01:~/CSCD240/Assignments$ cat >> file.txt  
Append Me  
^C
```

4. Write a command that counts the total number of lines the string “**bird**” exists in a file named “**The Rhyme of Ancient Mariner**” in your current directory. (1 point)
grep -c 'bird' 'The Rhyme of Ancient Mariner'
5. Write a command that searches the string “**line**” in all .c and .txt files starting from your current directory and all sub directories. (1 point)
grep --include=*.c,*.txt 'line' -R .

Metacharacters in Regular Expression

6. What will the following patterns match? Explain. (4 points, 1 point each)
 - a) **^bags\$** lines with only “bags” will match
 - b) **^...\$** lines with only 3 characters
 - c) **l.g** match any 3-letter word that starts with l and ends with g
 - d) **^\.** matches any line that starts with period

‘grep’, ‘find’, pipe

7. Consider the following file named “**FruitsList.txt**”. Try the following commands and explain each output with screenshot. (6 points, 1 point each)

- a) **grep “[A-Z]e” FruitsList.txt**
matches lines that contain an e after a capital letter
- b) **grep -i “[A-Z]e” FruitsList.txt**
matches any line containing a letter (case-insensitive) followed by e
- c) **grep “[^A-Z]e” FruitsList.txt**
matches lines with an e, with not capital A-Z in front
- d) **grep -i “^[A-Z]e” FruitsList.txt**
lines starting with any letter (case-insensitive) and an e
- e) **grep “^le” FruitsList.txt**
lines starting with le
- f) **grep “le\$” FruitsList.txt**
lines ending with le

FruitsList.txt

```
apple
Orange
Pineapple
Banana
lemon
Kiwi
berry
```

```
jlini1@cscd-linux01: ~/CSCD240/Assignments
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep "[A-Z]e" FruitsList.txt
apple
Orange
Pineapple
lemon
berry
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep -i "[A-Z]e" FruitsList.txt
apple
Orange
Pineapple
lemon
berry
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep "[^A-Z]e" FruitsList.txt
apple
Orange
Pineapple
lemon
berry
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep -i "[^A-Z]e" FruitsList.txt
apple
Orange
Pineapple
lemon
berry
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep "^le" FruitsList.txt
lemon
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep "le$" FruitsList.txt
apple
Pineapple
jlini1@cscd-linux01:~/CSCD240/Assignments$ grep -i "^ [A-Z]e" FruitsList.txt
lemon
berry
jlini1@cscd-linux01:~/CSCD240/Assignments$
```

Ignore highlighted command; correction at bottom

8. Suppose you are in your home directory. What are the differences between the following commands? Explain with screenshot. 1 point

`find . -name "*.txt"`

`find ~ -name "*.txt"`

command with ‘.’ will search current directory, and the command with ‘~’ will search the home directory. They will act the same because the current directory is the home directory

```
jlinil@cscd-linux01: ~  
jlinil@cscd-linux01:~$ find . -name "*.txt"  
.  
./bash_history  
./bash_logout  
./moo.swp  
./netstorage  
./CSCD240  
./CSCD240/Assignments  
./CSCD240/Assignments/The Rhyme of Ancient Mariner  
./CSCD240/Assignments/file.txt  
./profile  
./nano  
./cache  
./cache/motd.legal-displayed  
./bashrc  
find: '-name': No such file or directory  
find: '*.txt': No such file or directory  
jlinil@cscd-linux01:~$ find ~ -name "*.txt"  
/home/EASTERN/jlinil  
/home/EASTERN/jlinil/.bash_history  
/home/EASTERN/jlinil/.bash_logout  
/home/EASTERN/jlinil/.moo.swp  
/home/EASTERN/jlinil/netstorage  
/home/EASTERN/jlinil/CSCD240  
/home/EASTERN/jlinil/CSCD240/Assignments  
/home/EASTERN/jlinil/CSCD240/Assignments/The Rhyme of Ancient Mariner  
/home/EASTERN/jlinil/CSCD240/Assignments/file.txt  
/home/EASTERN/jlinil/.profile  
/home/EASTERN/jlinil/.nano  
/home/EASTERN/jlinil/.cache  
/home/EASTERN/jlinil/.cache/motd.legal-displayed  
/home/EASTERN/jlinil/.bashrc  
find: '-name': No such file or directory  
find: '*.txt': No such file or directory  
jlinil@cscd-linux01:~$
```

9. Write a command that finds all text files in your home directory and subdirectory and shows the long listing. 1 point

find . -name "*.txt" -ls

10. What will the following commands do? Explain with screenshots. (2 points, 1 point each)

a) `ls -l | grep '^.....rw'`

lists all files/dirs/links with read and write permissions on other

```
jlinil@cscd-linux01: ~/CSCD240/Assignments
jlinil@cscd-linux01:~/CSCD240/Assignments$ chmod o+rw permCheckOther
jlinil@cscd-linux01:~/CSCD240/Assignments$ ls -l | grep '^.....rw'
-rw-r--rw- 1 jlinil IT-GenericLinuxGroup 0 Jan 23 21:21 permCheckOther
jlinil@cscd-linux01:~/CSCD240/Assignments$ ls -l
total 16
-rw-r--r-- 1 jlinil IT-GenericLinuxGroup 39 Jan 23 11:29 file.txt
-rw-r--r-- 1 jlinil IT-GenericLinuxGroup 48 Jan 23 21:08 FruitsList.txt
-rw-r--rw- 1 jlinil IT-GenericLinuxGroup 0 Jan 23 21:21 permCheckOther
drwxr-xr-x 2 jlinil IT-GenericLinuxGroup 4096 Jan 23 21:03 subdir
-rw-r--r-- 1 jlinil IT-GenericLinuxGroup 15 Jan 23 11:18 The Rhyme of Ancient
Mariner
jlinil@cscd-linux01:~/CSCD240/Assignments$
```

- b) `grep -n variable *.*[ch]`
lists .c and .h files that contain the pattern 'variable' and lists the line number of where the pattern match was found

```
jlinil@cscd-linux01: ~/CSCD240/Assignments
jlinil@cscd-linux01:~/CSCD240/Assignments$ grep -n variable *.*[ch]
moo.c:1:variable
moo.h:4:variable
jlinil@cscd-linux01:~/CSCD240/Assignments$
```

Processes and Jobs

11. What is process? How will you differentiate processes from jobs? 1 point
Processes are programs and their wrappers that run in the system, jobs are also programs and their wrappers (so also processes) but they are tied to the active shell
12. What are the difference between the following commands: Explain with screenshot.
ps and **ps -aux**. 1 point
ps prints the active processes and jobs for the user, ps -aux shows a list of all processes and jobs currently running on the system for all users

Submission:

- A PDF file - Name this file as follows: your last name, first letter of your first name, Lab2.pdf (i.e., YasminSLab2.pdf). This file will contain all your answers. Each question should be copied first and then answered.
- You should turn in through the EWU Canvas system.
- Submission deadline is **Wednesday, January 23. No late submission will be accepted.**