Command line review

Glenn Bruns
CSUMB

Practicing command line

Use the flash cards on cram.com

www.cram.com/flashcards/bash-practice-6518378

www.cram.com/flashcards/awk-334-6529711

Review past labs and homework

Write your own flash cards

Use the command line in your everyday life

shell and environment variables

Which variables are inherited by child processes?

1. shell
2. environment
3. both

environment

shell and environment variables

Set new local variable x to 'okay'

$ _____

```
$ x=okay
```

shell and environment variables

Display the value of your PATH variable

$ _____

```
echo $PATH
```

shell and environment variables

Add directory /home/classes/maria/bin to your path

$ _____

```
PATH=$PATH:/home/classes/maria/bin
```

'export' only needed when an environment variable is created

Question

delete all files in the current directory ending with .c

```
rm *.c
```

Question

create an alias 'm' for the command 'less'

```
alias m=less
```

Question

create a file 'myfiles.txt' that contains the names of all files in the current working directory

(include files having names that start with .)

```
ls –a > myfiles.txt
```

Question

Use a bash loop to create files temp1, temp2, temp3

```
for i in 1 2 3; do touch temp$i; done

for i in {1..3}; do touch temp$i; done
```

Question

list all files in the current directory with 'assign' in the name, and ending with '.tar'

```
ls *assign*.tar
```

Question

display the value of bash variable x

```
echo $x
```

Question

create a single new file both.txt by concatenating files foo.txt and bar.txt

```
cp foo.txt both.txt
cat bar.txt >> both.txt
```

simpler:

```
cat foo.txt bar.txt > both.txt
```

Question

create a single new file all.txt by concatenating all .txt files in the current working directory

```
cat *.txt > all.txt
```

Question

create a tar file all-txt.tar from all the .txt files in the current working directory

```
tar cf all-txt.tar *.txt
```

Question

Write a makefile that will create a tar file all-txt.tar from all the .txt files in the current working directory

```
all-txt.tar:   *.txt
	tar cf all-txt.tar *.txt
```

Question

Add to your makefile to create a compressed tar file all-txt.tar.gz from all the .txt files in the current working directory

```
all-txt.tar.gz: all-txt.tar
	gzip all-txt.tar

all-txt.tar:   *.txt
	tar cf all-txt.tar *.txt
```

Question

Copy directory /home/classes/brun1994/private to your home directory. Don't spell out your home directory.

```
cp -r /home/classes/brun1994/private ~
```

Question

display all lines from file foo.txt in the current working directory that contain '334'

```
grep 334 foo.txt
```

Question

display all lines from all .txt files in the current working directory that contain '334'

```
grep 334 *.txt
```

Question

Suppose there is an absences file abs.txt that looks like this:

```
1023   9/2/2017
4432   9/15/2017
1023   9/27/2017
3329   9/31/2017
2225   10/8/2017
```

Write an awk script that will have output like this:

```
1023   2
4432   1
3329   1
2225   1
```

Don't worry about the order of the output

```
{ abs[$1]++ }
END {
  for (id in abs)
    print id " " abs[id]
}
```

Question

Run your awk script absences.awk on the input file abs.txt

```
$ awk -f absences.awk abs.txt
```

Question

Don't modify your awk script, but run it from the command line in a way that the output is sorted by number of absences, greatest first.

```
$ awk -f absences.awk abs.txt | sort -k2 -rn
```

Question

Don't modify your awk script, but take its output and produce a list of all students with at least one absence, printed in order of student id.

```
$ awk –f absences.awk abs.txt | awk '{print $1}' |
sort -n
```

Question

Write bash code, that, for each <name>.awk file in the current working directory, will create a directory <name>.awk.dir, and will move the awk file into the corresponding directory. Use a bash 'for' loop.

```
for f in *.awk; do
  mkdir $f.dir
  mv $f $f.dir
done
```

Question

Last night I was looking at available disk space on /home/CLASSES on mlc104. What command would you use to do this?

```
$ df /home/CLASSES

or just

$ df
```

Question

I was looking for files of size 1GB or larger in /home/CLASSES and below. What command your write to do this? Show the size of each file.

```
$ find . –size +1G -ls
```

Question

The previous command will give lots of 'Permission denied' errors. Change your code to hide this output

```
$ find . –size +1G -ls 2> /dev/null
```

Question

Write a bash script that will take a file name as parameter, and then use awk to count the number of lines in that file. (you can assume the file name is provided as a parameter)

```
#!/bin/bash
# print the number of lines in the specified file
awk '{cnt++}END{print cnt}' $1
```

Question

You have just written your bash script. Its name is 'cnt-lines.sh'. Now run it on file absences.txt

```
chmod +x cnt-lines.sh
./cnt-lines.sh absences.txt
```

Question

Assume the only .c files in the current working directory are msh1.c and msh2.c. They have 50 and 80 lines, respectively.

What happens when you run the following line?

$ ./cnt-lines.sh *.c

```
./cnt-lines.sh *.c
50
```

Question

Write a bash for loop that will print the numbers 1 to 10, in that order.

```
for i in {1..10}; do echo $i; done
```

Question

move file README (in the current working directory) to your home directory

```
mv README ~
```

Question

extract the files in tar file 'code.tar' that is in your current working directory

```
tar xf code.tar
```

Question

print the values of all environment variables

```
printenv
```

Question

append the number of lines of file 'foo.txt' to the end of file 'counts.txt'

```
wc -l foo.txt >> counts.txt
```

Question

uncompress file 'nov-24.tar.gz'

```
gunzip nov-24.tar.gz
```

Question

Show all the unique lines in file 'users.txt'

```
sort users.txt | uniq

cat users.txt | sort | uniq      # but why?
```

Question

show lines of file /etc/passwd that end with 'nologin'

```
grep 'nologin$' /etc/passwd
```

Question

print the length (number of characters) of the environment variable HOME

```
echo ${#HOME}
24

# checking the answer
echo $HOME
/home/CLASSES/brunsglenn
```

Question

print fields 1 and 4 of the output of command 'ps –Af'. Separate the field values with a comma

```
$ ps –Af | awk '{print $1","$4}'
```