CS 532 EC 0.2 - ODU CS Linux

Kimberley Cossey

2024-01-15

Part #1 - Setting up directory, file, and permissions

Creating a course directory & text file

I named my class directory `websci_202401` to match the names I use on my regular computer. I forgot to include the course number in the title like you suggested, so I will need to go back and edit the directory name when I figure out how to do that.

Using the `nano` command took me to a new screen to make the `test.txt` file. I made a mistake when I created the file - I hit "enter" after typing the last class. I didn't know how to fix the file after I had saved it, so my line counts and some of my sorting commands had an extra blank line.

```
cs_kcoss002@ganymede:~$ pwd
/home/cs_kcoss002
cs_kcoss002@ganymede:~$ cd websci_202401/
cs_kcoss002@ganymede:~/websci_202401$ pwd
/home/cs_kcoss002/websci_202401
cs_kcoss002@ganymede:~/websci_202401$ nano test.txt
```

Figure 1: `mkdir` & `nano` commands

Permissions

I forgot to do this when I created the file. I was able to fix it later in the process.

```
cs_kcoss002@ganymede:~/websci_202401$ -l
-l: command not found
cs_kcoss002@ganymede:~/websci_202401$  ls -l
total 0
-rw-r--r-- 1 cs_kcoss002 grad 46 Jan 15 13:21 test.txt
cs_kcoss002@ganymede:~/websci_202401$ chmod g-rwx test.txt
cs_kcoss002@ganymede:~/websci_202401$ chmod o-rwx test.txt
cs_kcoss002@ganymede:~/websci_202401$ ls -l
total 0
-rw------- 1 cs_kcoss002 grad 46 Jan 15 13:21 test.txt
cs_kcoss002@ganymede:~/websci_202401$
```

Figure 2: `chmod` view and edit

Part #2 - Command Descriptions

Question #1

```
cs_kcoss002@ganymede:~/websci_202401$ wc -l text.txt
wc: invalid option -- 'l'
Try 'wc --help' for more information.
cs_kcoss002@ganymede:~/websci_202401$ wc -l test.txt
wc: invalid option -- 'l'
Try 'wc --help' for more information.
cs_kcoss002@ganymede:~/websci_202401$ wc -l test.txt
6 test.txt
```

Figure 3: `wc -l`

This command printed the number of lines that were in the test file. It took a couple of tries because I have trouble telling the difference between 1 and the letter "l" when I have to use that font. In my case I had 6 lines since it counted the blank line.

Question #2

```
cs_kcoss002@ganymede:~/websci_202401$ echo "CS 800" >> test.txt; cat test.txt
CS 800
CS 432
CS 725
MATH 212
MATH 32

CS 800
```

Figure 4: `echo "CS 800" >> test.txt; cat test.txt`

This command added "CS 800" to a new line at the end of the test file, so now the test file has 7 lines.

Question #3

```
cs_kcoss002@ganymede:~/websci_202401$ grep CS test.txt
CS 800
CS 432
CS 725
CS 800
```

Figure 5: `grep CS test.txt`

`grep` printed the full text of all the lines in the test file that included the string "CS".

Question #4

This command printed the number of lines in the test file that included the string "CS".

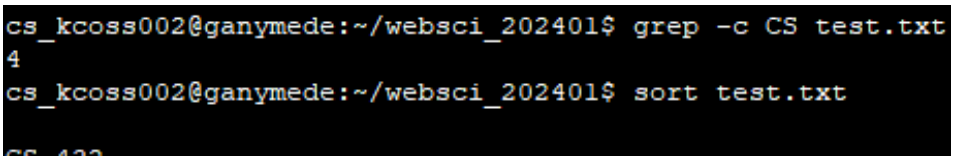

Figure 6: `grep -c CS test.txt`

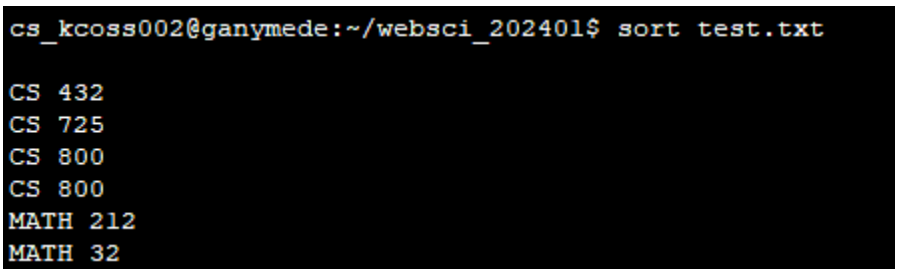

Figure 7: `sort test.txt`

Question #5

The basic sort command sorted the lines in the text file by alphabetical order of the first letter, then by ascending order of the first digit of the class number, and finally then showed the full text of the file. It looks like the blank line was sorted to the top.

Question #6

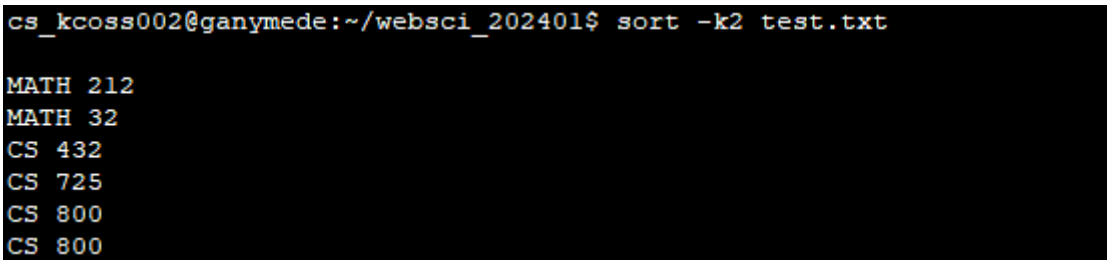

Figure 8: `sort -k2 test.txt`

Adding the `-k2` sorted the lines in the file by reverse alphabetical order, but still sorted the first digit of the class number as ascending (as in, it didn't reverse sort the first digit from 9 to 1). The blank line is still sorted to the top.

Question #7

Adding the `-n` sorted the lines in the file by reverse alphabetical order, then sorted the class numbers ascending by their full numeric value (it finally recognized that 32 was less than 212).

Question #8

Adding the `uniq -c` command sorted the lines in alphabetical order, then by full numerical value of the classes. It also returned the number of lines that contained (or maybe exactly matched) each string. It

```
cs_kcoss002@ganymede:~/websci_202401$ sort -k2 -n test.txt

MATH 32
MATH 212
CS 432
CS 725
CS 800
CS 800
```

Figure 9: `sort -k2 -n test.txt`

```
cs_kcoss002@ganymede:~/websci_202401$ sort test.txt | uniq -c
      1
      1 CS 432
      1 CS 725
      2 CS 800
      1 MATH 212
      1 MATH 32
```

Figure 10: `sort test.txt | uniq -c`

shows that there are 2 lines that have the string CS 800, so this would be a good way to check for duplicates in longer files.

References

Basic Linux commands

- https://www.digitalocean.com/community/tutorials/linux-commands

Creating directories & files

- https://www.redhat.com/sysadmin/create-delete-files-directories-linux
- https://www.freecodecamp.org/news/how-to-save-and-exit-nano-in-terminal-nano-quit-command/

How to do permissions

- https://www.cs.odu.edu/~zeil/cs252/latest/Public/protection/index.html