Blaine Mason

Lab-02

Dr.Park

Task 1:

1.) Write down the value of your PS1 environment variable as well as the file in which it was set.

```
bash-3.2$ echo $PS1
\s-\v\$
```

- File was in /etc/bashrc

2.)Write down the values of the above environment variables on your Linux machine. Indicate any that have no value.

```
bash-3.2$ echo $EDITOR
No Value.
bash-3.2$ echo $HOME
/Users/blaine-mason
bash-3.2$ echo $HOSTNAME
Blaines-MacBook-Pro.local
bash-3.2$ echo $LD_LIBRARY_PATH
No Value.
bash-3.2$ echo $LESS
-R
bash-3.2$ echo $MAIL
No Value.
bash-3.2$ echo $MANPATH
No Value.
bash-3.2$ echo $MORE
No Value.
bash-3.2$ echo $PAGER
less
bash-3.2$ echo $PATH
/usr/local/opt/tcl-
tk/bin:/usr/local/anaconda3/bin:/usr/local/anaconda3/condabin:/usr/local/bi
n:/usr/bin:/bin:/usr/sbin:/sbin
bash-3.2$ echo $PWD
/Users/blaine-mason
bash-3.2$ echo $SHELL
/bin/zsh
```

```
bash-3.2$ echo $TERM
xterm-256color
bash-3.2$ echo $USER
blaine-mason
```

Task 2:

1.)Save your prompt. Write down exactly how you did it.

```
bash-3.2$ echo $PS1 >> prompt.txt
```

2.) Change your prompt so it looks like [COSC350 basecwd]: where "basecwd" means the basename of the current working directory. Write down exactly how you did it.

```
bash-3.2$ PS1="[COSC350 \W]"
```

3.) Change your prompt to its previous value. Write down exactly how you did it.

```
[COSC350 ~]PS1=`cat prompt.txt`
bash-3.2$
```

Task 4:

1.) Invoke ls with a non-existent filename. You should see the error output on the screen. Do it again, but redirect the error output to a file named bar. Write down exactly how you did this.

```
bash-3.2$ls non
ls: non: No such file or directory
```

```
bash-3.2$ls non 2> bar
bash-3.2$cat bar
ls: non: No such file or directory
```

2.) Do it again, but redirect the error output to the "gone forever" file /dev/null. Write down exactly how you did this.

```
bash-3.2$ls non 2> /dev/null
```

3.) Create a file named foo by echo-ing the numbers 3,5,2,1 into it, one number per line. Write down exactly how you did this.

```
bash-3.2$echo 3 > foo
bash-3.2$echo 5 >> foo
bash-3.2$echo 2 >> foo
bash-3.2$echo 1 >> foo
bash-3.2$cat foo
3
5
2
1
```

4.) Create a file named bar by cat-ing foo into it. Write down exactly how you did this. (Yes, cp would also work, but this lab exercise is about redirection.)

```
bash-3.2$cat foo >> bar
bash-3.2$cat bar
3
5
2
1
```

5.) Redirect input from foo (it contains numbers, right?) to the sort function. You should see the sorted numbers on the screen. Write down exactly how you did this. Did the numbers turn out sorted numerically? If not, explain how the sort was done.

```
bash-3.2$sort < foo
1
2
3
5
```

- Yes they are sorted numerically

6.) Do it again, but redirect the output from the screen into the file bar. Write down exactly how you did this.

```
bash-3.2$sort < foo >> bar
bash-3.2$cat bar
1
2
3
5
```

Task 5:

1.)Create a file named numbs that contains the integers 1 through 100, one integer per line. The file will have 100 lines. Write down a short description of how you did this. (You can do it any way you want, including dumb brute force. You might also want to consider the bash for loop or a small C++ program.)

```
(base) ➜  ~ seq 1 100 > numbs
(base) ➜  ~ man seq
NAME
       seq - print a sequence of numbers
```

- I used the seq command which allows you to output a sequence of numbers, then redirected the output to the file numbs.

2.) Run wc on the file numbs. Write down the output and your explanation of what it means. Check the man page for wc if you're not sure.

```
(base) ➜  ~ wc numbs
100 100 292 numbs
```

- Number of new lines, words, bytes for file

3.) Use pipes and redirection to produce a second file named somenumbs that contains lines 25 through 38 of numbs. Write down exactly what you did.

```
(base) ➜  ~ tail -n +25 numbs | head -n 14 > somenumbs
(base) ➜  ~ cat somenumbs
25
26
27
28
29
30
31
32
33
34
35
36
37
38
```

4.) Run wc on the file somenumbs. Write down the output and your explanation of what it means.

```
(base) ➜  ~ wc somenumbs
14 14 42 somenumbs
```

- Number of new lines, words, bytes for file