HW2

Hyunki Lee

Part 1

1. grep: Filter out all lines that do not contain a specified pattern. Using basic regular expression

   egrep: The same as grep but egrep uses extended regular expression.

   fgrep: fgrep is also same as grep, but it searches for fixed – character strings.

2. ‘tar’ can be used to compress or decompress files.

   For compressing multiple files into a single file: If I want to make an archive file(test) and add multiple files (file_name1, file_name2) in the file(test), the command will be

   tar -cvf test.tar file_name1 file_name2

3. awk

   Default field separator: spaces and tabs

   Define a separator manually in the command line:

   Ex) $awk -F# ‘program’ /home/file

   In this case, to use ‘#’ as the field separator

Part 2

1. c. Hello World!!!
2. e. none of above. Answer will be awk 'NR<=5{print $1}' filename
3. b. good
4. d. /\+$/{print $0}
5. e. sed '1,5d' file

Part 3

1. Output: 3:When everything seemed so clear.

   Function: NR means current line numbers. In this case, the line number is more than 2 and less than 4. Therefore, it is 3rd line. The function is printing the 3rd line from beginning to end because of $0.

2. Output: 1:Wish

   2:When

   4:Now

   Function: The function means start with any single character with many times and end with 'ing'. Output prints the first words of the matching lines.

3. Output: Start to scan file

   Wish,strong,

   And,days

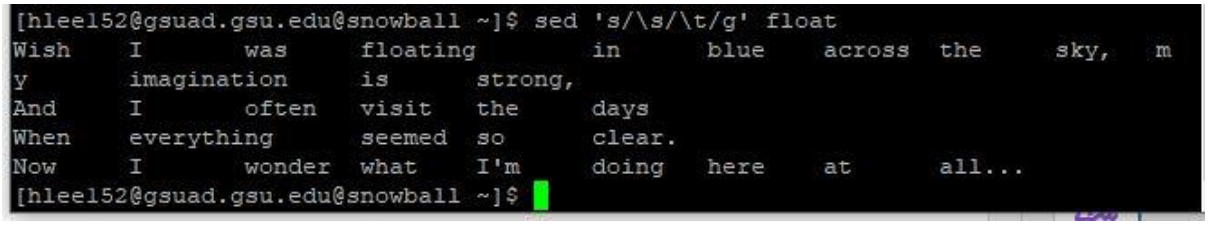
   When,clear.

   Now,all…

   END- float

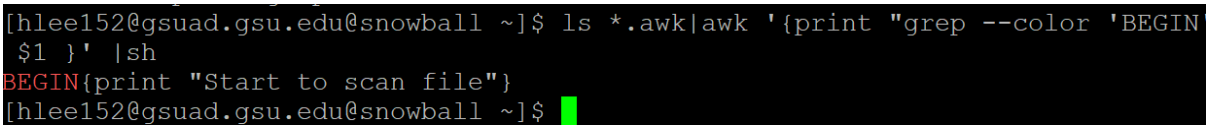
Function: NF means number of fields. It begins to print ‘start to scan file’, and It prints the first item of each line. ‘,’ used as separator. Also, last item of each line prints because of $NF. At the end, the function prints ‘END- float’.

4. Output:



Function: ‘s/’ means substitute following conditions and ‘/g’ stands for entire contents of a file. The condition ‘\s’ and ‘\t’ means that substitute space to tab character.

5. Output



Function:

ls *.awk – using awk extension, list all the files. awk '{print "grep --color 'BEGIN' " $1 }' means awk files are treated as input and search ‘BEGIN’. After searching ‘BEGIN’, applying color on the matching characters. Print it on the current line and output is printed on shell.

6. Output

```
[hlee152@gsuad.gsu.edu@snowball ~]$ mkdir test test/test1 test/test2
[hlee152@gsuad.gsu.edu@snowball ~]$ cat>test/test.txt
[hlee152@gsuad.gsu.edu@snowball ~]$ cd test
[hlee152@gsuad.gsu.edu@snowball test]$ ls -l .|grep '^d'|awk '{print "cp -r " $
F " " $NF".bak"}' |sh
```

There is no visible output however, I can check that the function works.

```
[hlee152@gsuad.gsu.edu@snowball ~]$ mkdir test test/test1 test/test2
[hlee152@gsuad.gsu.edu@snowball ~]$ cat>test/test.txt
[hlee152@gsuad.gsu.edu@snowball ~]$ cd test
[hlee152@gsuad.gsu.edu@snowball test]$ ls -l .|grep '^d'|awk '{print "cp -r " $N
F " " $NF".bak"}' |sh
[hlee152@gsuad.gsu.edu@snowball test]$ ls
test1  test1.bak  test2  test2.bak  test.txt
[hlee152@gsuad.gsu.edu@snowball test]$ ls -l
total 16
drwxrwxr-x. 2 hlee152@gsuad.gsu.edu hlee152@gsuad.gsu.edu 4096 Sep 24 11:29 test
1
drwxrwxr-x. 2 hlee152@gsuad.gsu.edu hlee152@gsuad.gsu.edu 4096 Sep 24 11:30 test
1.bak
drwxrwxr-x. 2 hlee152@gsuad.gsu.edu hlee152@gsuad.gsu.edu 4096 Sep 24 11:29 test
2
drwxrwxr-x. 2 hlee152@gsuad.gsu.edu hlee152@gsuad.gsu.edu 4096 Sep 24 11:30 test
2.bak
-rw-rw-r--. 1 hlee152@gsuad.gsu.edu hlee152@gsuad.gsu.edu    0 Sep 24 11:30 test
.txt
[hlee152@gsuad.gsu.edu@snowball test]$
```

Function: ‘ls -l’ means to list files in the directory. ‘grep ‘^d’’ means that the command search beginning of line and get ‘d’. Therefore, test1 and test2 finds from the command. Then, test1 directory is copied as test1.bak, and test2 is also copied as test2.bak. ‘NF’ is the number of fields.