CSc 3320: Systems Programming

Spring 2021

Homework

# 2: Total points 100

Submission instructions:

1. Create a Google doc for each homework assignment submission.

2. Start your responses from page 2 of the document and copy these instructions

on page 1.

3. Fill in your name, campus ID and panther # in the fields provided. If this

information is missing in your document TWO POINTS WILL BE DEDUCTED per

submission.

4. Keep this page 1 intact on all your submissions. If this submissions instructions

page is missing in your submission TWO POINTS WILL BE DEDUCTED per

submission.

5. Each homework will typically have 2-3 PARTS, where each PART focuses on

specific topic(s).

6. Start your responses to each PART on a new page.

7. If you are being asked to write code copy the code into a separate txt file and

submit that as well.

8. If you are being asked to test code or run specific commands or scripts, provide

the evidence of your outputs through a screenshot and copy the same into the

document.

9. Upon completion, download a .PDF version of the document and submit the

same.

Full Name: Dang Truong

Campus ID: 002420309

Panther #: 002420309

PART1:

1. Grep is used for finding a particular work from the file wherever it occurs. For example, $ grep ‘word’ filename, this command will search for any line with ‘word’ and highlight.

* egrep: this command used for pattern searching. For example, egrep “word” filename, it will display the entire line containing the searched string ‘word’.
* fgrep: this command used to search for the fixed-character strings in a file. For example, $ fgrep -h “usin.g” filename, this will display the matched lines.

1. - tar

- $ tar -czvf archive.tar.gz filename

* $ tar -cxvf archive.tar.gz filename
* $ tar -cvf file1 file1 | compress > my\_file.tar.z

1. awk

* the default separator is tab or \t
* awk -F “\t” ‘{print $1 | $2}’ filename

1. sort command sorts the contents of a text file line by line, it prints the lines of its input or concatenation of all files listed in its argument list in sorted order.

PART2a:

1. Hello World!!!
2. - prints the fifth item of each line in the file

* prints the first item of each line in the file where row number >=5 instead of printing first five lines.
* print $0 prints the current line instead of first field of each line in the file
* print $1 prints all the first fields of each line in the file instead of printing first field of first five lines

1. good
2. /\+$/{prints $0}
3. to delete first 5 lines, 1,5d, and to delete the last 5 lines, head -n-5

PART2b:

1. Function: the file h1.awk contains the data: NR>2 && NR<4{print NR”:”$0}, so the command awk ‘./\*ing/{print NR “:” $1}’ float searched for the rows where there is a string with a word that contains ‘ing’ in it.

* output: 1: wish, 3: when, 4: Now

1. Function: awk -f h1.awk and float files as progfile which takes input data from the two files, by h1.awk it specifies to print the data between records 2 and 4 so it prints 3.

* Output: 3: When everything seemed so clear.

1. Function: print ‘start to scan file’ in the beginning, then print first and last field of each line by separating with ‘,’ from $NF, and last print END with the filename float.

* Output: Start to scan file, Wish, Strong. And, days When, Clear. Now, all… END- float

1. Function: sed patterns filename, sed ‘s/\s/\t/g’ float where s stands for substitute, and /g apply to entire file, sed substitutes spaces encountered with \t of the entire file.

* output: Wish I was floating in blue across the sky,my imagination is strong. And I often visit the days. when everything seemed so clear. Now I wonder what I’m doing here at all…

1. Function: a|b|c where the output from a is fed as the input to b and output from b is fed as input to c . ls\*.awk-list all th files with awk extension. awk ‘{print “grep –color ‘BEGIN’” $1}’ this command takes awk files as input and searches for matching ‘BEGIN’ and applies color to it and prints the current line and output is printed on the shell.

* output: h1.awk h2.awk
* BEGIN {print “Start to scan file”}

1. Function: mkdir test makes test directory and subdirectory test1 and test2 under the directory. cat>test/test.txt this is a test file^D will create the test.txt file under test directory. cd test to go inside test directory. ls -l .|grep ‘^d’|awk ‘{print :cp-r” $NF “ “ $NF “.bak”}’ | sh will create the backup file with the .bak extension all the directory and the subdirectory.

PART3:







