CSCI 3308 Software Development Methods and Tools
Spring 2018

Homework # 1

Bash Shell Scripts & Regular Expressions

Due Date: 11:55pm on Sunday, February 11, 2018

Objectives

- Write a bash shell script with bash shell commands to loop through a data file
- Write a bash shell script using UNIX commands like “awk”
- Practice using Regex commands to parse text
- Gain more experience in pair-programming collaboration [optional]

Part 1 Bash Shell Scripts

Step1:
Place the below content in a file named StudentGrades.txt :

```
123456789 Lee Johnson 72 85 90
999999999 Jaime Smith 90 92 91
888111818 JC Forney 100 81 97
290010111 Terry Lee 100 99 100
199144454 Tracey Camp 77 84 84
299226663 Laney Camp 70 74 71
434401929 Skyler Camp 78 81 82
928441032 Jess Forester 85 80 82
928441032 Chris Forester 97 94 89
```

Step 2A:
Create a bash script file with the name Grades.sh

Step 2B:
Create another AWK script file with the name GradesAwk.sh

Step 3:
The above 2 files must contain scripts to:

a) Read the contents of StudentGrades.txt
b) Calculate the average of the scores for each record
c) Sort the output by last name and then the first name
d) Format the output as shown in the “Report” below

Report:

```
71 [299226663] Camp, Laney
80 [434401929] Camp, Skyler
81 [199144454] Camp, Tracey
93 [928441032] Forester, Chris
82 [928441032] Forester, Jess
92 [888111818] Forney, JC
82 [123456789] Johnson, Lee
99 [290010111] Lee, Terry
91 [999999999] Smith, Jaime
```

First Column should be the average(Rounded or truncated averages are accepted). Second Column is the Student ID within square brackets. Third column is the Student Last name. This is followed by a comma, space, and the First name.

Output should be sorted, first based on the last name. If the last name is the same, sort then on the first name. If the person has the same last name and first name, then sort based on the ID. All IDs are unique in the file.

Your scripts will be tested against different test datafiles(Not just the content in StudentGrades.txt). However, the test datafiles used for evaluation will be in the same format as in StudentGrades.txt, though it may contain more or less number of lines in the file. All students in the test datafiles will have 3 grades.

The objective of writing two scripts is to see that there are multiple correct solutions to such problems. One solution should use the awk tool, and the other should use bash commands (bash scripting).

Part 2 Regex

Download the Regex Practice Data from Moodle. Create RegexAnswers.sh and for each of the questions listed below, write the regex expression necessary to calculate the answer.

Hints :

- The command grep and egrep are your friends (egrep treats { } differently than grep)
- Be sure to check for word boundaries in your answers ‘\b’ where appropriate
- Pipe answers to “wc –l” to get the count

1. How many lines end with a number?
2. How many lines do not start with a vowel?
3. How many 12 letter (alphabet only) lines?
4. How many phone numbers are in the dataset (format: ‘___-___-____’)?
5. How many city of Boulder phone numbers (Eg: starting with 303-___-____)?
6. How many begin with a vowel and end with a number?

7. How many email addresses are from geocities? (Eg: end with'geocities.com')?
8. How many email addresses are in ‘first.last’ name format and involve someone who’s first name starts with a letter in the first half of the alphabet?

Running RegexAnswers.sh script file should output 8 lines which is the result of ‘wc –l’ for each regex command. If unsure of any one of the answers, use echo “0” so that the rest of your answers align in the output.

Requirements:

1. Scripts must be bash files named
   - Grades.sh
   - GradesAwk.sh
   - RegexAnswers.sh

2. At the top of your scripts, include comment with your name (and your partner’s name if you pair program)

3. For all scripts, read in the name of the data file from command-line arguments. (The file names should not be hard coded in the scripts) We will test all the three scripts with additional data files that have different names

4. If the program is run without the filename as the command-line argument, print out the usage statement:
   Usage: Grades.sh filename
   Or
   Usage: GradesAwk.sh filename
   Or
   Usage: RegexAnswers.sh filename

6. A single zipped file containing all three scripts should be uploaded on Homework1 Submission Link. Only **One** submission is expected if you pair-program

   NOTE:
   If you are working alone, name the zip file using the following template:
   Lastname_HW1.zip
   If you are pair programming, then name the file using this template:
   Lastname1_Lastname2_HW1.zip

All 3 scripts should be runnable from command line where filename is given as an argument. If a script doesn’t execute or doesn’t provide the right output, then points will be deducted.