**CSC3320 System Level Programming**

**Program Challenge 4**

**(27 points in total)**

**Part 1: File processing.**

The following table is from Wikipedia. It shows the eleven highest mountains in Georgia.

|  |
| --- |
| [Brasstown Bald](http://en.wikipedia.org/wiki/Brasstown_Bald) ,(summit),4784,feet,[Union County](http://en.wikipedia.org/wiki/Union_County,_Georgia)  [Rabun Bald](http://en.wikipedia.org/wiki/Rabun_Bald), (summit),4696,feet,[Rabun County](http://en.wikipedia.org/wiki/Rabun_County,_Georgia)  [Dick's Knob](http://en.wikipedia.org/wiki/Dick%27s_Knob), (summit),4620,feet,Rabun County  [Hightower Bald](http://en.wikipedia.org/wiki/Hightower_Bald), (summit),4568,feet,[Towns County](http://en.wikipedia.org/wiki/Towns_County,_Georgia)  [Wolfpen Ridge](http://en.wikipedia.org/wiki/Wolfpen_Ridge), (ridge high point),4561,feet,Towns and Union Counties  [Blood Mountain](http://en.wikipedia.org/wiki/Blood_Mountain), (summit),4458,feet,Union County  [Tray Mountain](http://en.wikipedia.org/wiki/Tray_Mountain), (summit), 4430,feet,Towns County  [Grassy Ridge](http://en.wikipedia.org/wiki/Grassy_Ridge), (ridge high point),4420,feet,Rabun County  [Slaughter Mountain](http://en.wikipedia.org/wiki/Slaughter_Mountain), (summit),4338,feet,Union County  [Double Spring Knob](http://en.wikipedia.org/wiki/Double_Spring_Knob), (summit),4280,feet,Rabun County  [Coosa Bald](http://en.wikipedia.org/wiki/Coosa_Bald), (summit),4280,feet,Union County |

In above table, each line contains 5 fields separated by comma. Create a file named “mountainList.txt” in your computer, and copy the contents in above table to it.

Or use the following commands to get the file:

**wget cs.gsu.edu/~ylong4/2016fall/mountainList.txt**

Or if in snowball by the following command

**cp /home/local/GSUAD/ylong4/public/ mountainList.txt mountainList.txt**

Be sure it succeeds using “ls” to see the file name “mountainList.txt” listed.

**Please complete the following tasks and answering corresponding questions step by step.**

1. Type command **cat –n mountainList.txt** and execute it. Then attach a screenshot of the output. **1 point**

1 point

1. Type command **grep –-color -n ‘Union’ mountainList.txt** and execute it. Then attach a screenshot of the output.

1 point

Describe what does the above grep command do?

Output all lines containing Union 1 point

1. Type command **grep –-color -n ‘Union|Rabun’ mountainList.txt** and execute it. Then attach a screenshot of the output.

No output. 1 point

Describe what does the above grep command do?

Searching for lines containing “Union|Rabun”. 1 point

1. Type command **egrep –-color -n ‘Union|Rabun’ mountainList.txt** and execute it. Then attach a screenshot of the output.

1 point.

Describe what does the above egrep command do?

Searching for lines containing **Union or Rabun 1 point**

1. Write the command to count the number of mountains with “ridge high point”. (Hint: use Pipe ‘|’ and **wc**) **2 point**

$grep 'ridge high point' mountainList.txt | wc –l

grep 0.5 point

‘ridge high point’ 0.5 point

Mountainlist.txt 0.5 point

wc -l 0.5 point

1. Type command **sed ‘/summit/p’ mountainList.txt** and execute it. Then attach a screenshot of the output.

1 point

Type command **sed –n ‘/summit/p’ mountainList.txt** and execute it. Then attach a screenshot of the output.

1 point

Open the manual page of **sed** and describe what does **–n** do in **sed**?

Suppress automatic printing of pattern space. 1 point

Describe what does the above sed command do?

( Above sed command means **sed –n ‘/summit/p’ mountainList.txt** )

Print out lines containing summit. 1 point

1. Use **sed** to list the lines beginning with white spaces in mountainList.txt . (2 points)

sed –n ‘/^ \*/p’ mountainList.txt

-n 0.5 point

/^ \*/ 0.5 point p 0.5 point

MountainList. Txt 0.5 point

If students use a single space between ^and \*, -0.5

sed –n ‘/^ +/p’ mountainList.txt

-n 0.5 point

/^ +/ 0.5 point p 0.5 point

MountainList. Txt 0.5 point

1. Use **sed** to remove the leading spaces in each line of mountainList.txt and save the output to file mountainList\_v1.txt **2 point**

sed 's/^ \*//g' mountainList.txt > mountainList\_v1.txt

's/^ \*//g' 1 point

MountainList. Txt 0.5 point

> mountainList\_v1.txt 0.5 point

If students use a single space between ^and \*, -0.5

1. Use **sed** to remove the leading spaces in the lines only for the mountains with “ridge high point” in mountainList.txt . **2 point**

sed '/ridge high point/s/^ \*//g' mountainList.txt

/ridge high point/ 1 point

s/^ \*//g' 0.5 point

mountainList.txt 0.5 point

1. Use **sed** to remove the second field and a comma after it in each line of *mountainList.txt*, and save the output to file mountainList\_v2.txt . **2 point**

E.g. after removing, the first line should be changed to

[Brasstown Bald](http://en.wikipedia.org/wiki/Brasstown_Bald) ,4784,feet,[Union County](http://en.wikipedia.org/wiki/Union_County,_Georgia)

Note: **sed** can only support basic regular expression unless the option –E is used for extended regular expression.

Option 1: sed 's/(.\*),//' mountainList.txt > mountainList\_v2.txt

mountainList.txt 0.5 point

s/..// 0.5 point

(.\*), 1 point

Option 2: sed –E 's/\(.\*\),//' mountainList.txt > mountainList\_v2.txt

mountainList.txt 0.5 point

s/..// 0.5 point

\(.\*\), 1 point

**If no -E , -0.5 point**

Option 3: sed –E 's/\(summit\),|\(ridge high point\),//' mountainList.txt > mountainList\_v2.txt

mountainList.txt 0.5 point

s/..// 0.5 point

\(summit\),|\(ridge high point\), 1 point

**If no -E , -0.5 point**

1. Use **sed** to insert a new line “Table: Eleven highest mountains in Georgia” at the beginning of mountainList\_v1.txt . ***2 points***

$cat>sedp

1i\

Table: Eleven highest mountains in Georgia 1 point

$sed -f sedp mountainList\_v1.txt 1 point

Note: For each minor problem -0.5, minus 1 point in maximum

$sed ‘1i\Table: Eleven highest mountains in Georgia ’ mountainList\_v1.txt

1. **Sort** *mountainList\_v1.txt* according to the names of mountains (1st field). 2 points

$sort mountainList\_v1.txt

Note: if they put more options, you do not need to check it and students still get full mark.

**Part 2 - optional: More challenges about file processing.**

Note: this part is not for grading. But TA will check your answer and provide you feedbacks.

1. Use **awk** to remove the first comma in each line of mountainList\_v1.txt

$cat>awkp

{print $1 $2 “,” $3 “,” $4 “,” $5}

$awk –F, -f awkp mountainList\_v1.txt

1. “When a pattern groups all or part of its content into a pair of parentheses, it captures that content and stores it temporarily in memory. You can reuse that content if you wish by using a **backreference**, in the form:\1 or $1, where \1 or $1 reference the first captured group” (Refer to [1]). For example, the following command add a comma between Union and County

**sed -E ‘s/(Union)\s(County)/\2:\1/g’** **mountainList\_v1.txt**

Attach a screenshot of the output of above **sed** command.

1. Now can you write a command to remove the first comma in each line of mountainList\_v1.txt using **sed** ?

$sed -E 's/(.\*),(.\*),(.\*),(.\*),(.\*)/\1\2,\3,\4,\5/g' mountainList\_v1.txt

**Useful Links:**

[1] Introducing Regular Expression - Capturing Groups and Backreferences

<https://www.safaribooksonline.com/library/view/introducing-regular-expressions/9781449338879/ch04.html>

[2] Drew's grep tutorial

<http://www.uccs.edu/~ahitchco/grep/>

[3] Grep and Regular Expressions!

<http://ryanstutorials.net/linuxtutorial/grep.php>

[4] Web Scraping with Regular Expressions

<https://www.datascraping.co/doc/22/regular-expression>

***Submssion***:

* Upload an electronic copy (MS word or pdf) of your answer sheet to the folder named “**PC4**” of the dropbox in the iCollege system.
* Please add the program challenge number and your name at the top of your answer sheet.
* Name your file in the format of PC4\_FisrtnameLastname (eg. PC4\_YuanLong.docx, PC4\_YuanLong.pdf)