

CSE 391, Autumn 2019

Homework 4: More Regular Expressions and Sed

Due Tuesday, October 29, 2019, 1:00 PM

This assignment focuses on using regular expressions and related command `sed`. A set of files you will need for this assignment are available in the file `hw4.zip`, found on the Homework page.

Submit your answers to Gradescope. In response to each question, **write the command that will perform the task described**, *not the output that the command produces*. Please be sure to write the **entire command** (including the command name and the input file).

Task 1: Bash Shell Commands with `sed`:

For each of the next few problems, write a command that uses `sed` (with the `-r` command-line argument to enable full regular expressions) to search and replace text based on regular expressions. Each command should use **at most one call** to `sed`.

Your commands **should not create any temporary files** during their execution. Feel free to somewhat match your answers to the actual file you are given (e.g. only one word per line); your regexes do not have to work for a more general case). Write your commands in on the indicated lines in the `task1.sh` file in the `hw4` folder.

1. Output the contents of the file `email.txt` with all spaces replaced by dashes (-).
2. Josh and Zorah are debating which candy is better: kitkats or twix. We polled our friends and compiled the results in `candies.txt`. A quick run of `grep -i "kitkat" candies.txt | wc -l` and `grep -i "twix" candies.txt | wc -l` shows that 3 people prefer twix and 2 people prefer kitkats. Josh thinks this is a mistake, however, as people may have answered “kit kat” (with a space), “kitkats” (with an s) or “kit kats” (with both a space and an s). Write a `sed` command to replace all these variants with just the basic “kitkat”. You may assume that they are all lowercase.
3. Midterm season is upon us, and things get a little rough sometimes. Write a `sed` command that replaces all frowny faces in `boundless.txt` with smiley faces! You may assume that all frowny faces appear with the eyes to the left of the mouth.
4. Europeans format their dates differently than Americans. Where we would write a date such as "May 12, 2010", they would write it as "12 May 2010". Output the contents of file `dates.txt` but with all dates changed from USA format to European format. Don't worry about coming up with a fancy regex to match only legal months or only legal days of the month.
5. Convert all 10-digit phone numbers in the file `phone.txt` to 5-digit internal extension numbers (the last 5 digits of the phone number).
(For example, Abba, Cadabra x67890 and Timss, Aaron x62859.)
6. Java programs can contain single-line `//` comments and multi-line `/* ... */` comments. Sometimes a programmer uses a multi-line comment syntax, but the comment only occupies a single line. Write a command that finds `/* ... */` comments in `Questions.java` **that occupy a single line** and replaces them with a `//` comment. For example, `/* hello there */` would become `// hello there`. (Your command doesn't need to modify comments where the `/*` isn't on the same line as the `*/`. You may assume that any given line contains at most one comment.)

Hints: The regular expression syntax hints from the previous section on `grep` also apply to `sed`. Also recall that some special characters must be escaped by a `\` backslash to be used in a regex pattern. Remember that putting a `'g'` at the end of your pattern, such as `s/oldpattern/newtext/g`, processes all matches on a line.