SOC 103M Winter 2002 Midterm Solutions

Dana Dahlstrom

2002.02.13

1. [44 points]
   (a) [4 points] %s/utta/utter/g
   (b) [6 points] %s/Calcutter/Calcutta/g or /Calcutter followed by cwCalcutta or any other description of a working solution
   (c) [2 points] 3G or :3 goes to line 3
   (d) [2 points] 11\ goes to column 11 in the current line
   (e) [2 points] 2h4l or hh1111
   (f) [2 points] $ goes to the end of the line
   (g) [2 points] 4G or :4 (or j from line 3) goes to line 4. * goes to the first non-blank character. 0 goes to the first character even if blank. Also, 4G and :4 already go to the first non-blank character (j doesn’t).
   (h) [2 points] 2k or kk
   (i) [2 points] 6G or :6 goes to line 6; dd deletes the line
   (j) [2 points] p puts the text after the cursor
   (k) [2 points] 6G or :6 goes to line 6; "add deletes the line into register a
   (l) [2 points] "aP puts the text from register a before the cursor
   (m) [4 points] 1G or :1 goes to line 1; A appends at the end of the line
   (n) [4 points] 1G or :1 goes to line 1; I inserts before the first non-blank character on the line.
   (o) [4 points] Ctrl-D scrolls down half a screen; Ctrl-U scrolls up half a screen; H goes to the first non-blank character of the line at the top of the screen; L goes to the first non-blank character of the line at the bottom of the screen.
   (p) [2 points] :wq writes the current file and quits

2. [44 points]
   (a) [6 points] grep "t[ eoia]n" oink.txt
(b) [14 points]
  grep "[A-Z][a-h2-6,\[357]"
  grep \~"[A-Z][a-h2-6,\[357]"
  grep "^[A-Z][a-h2-6,\[357]"
  grep "^[A-Z][a-h2-6,\[357]$"

(c) [4 points] grep "^$"

(d) [4 points] sort -k1,1 clean_limerick.txt

(e) [8 points]
  ls
  ls > listoffiles.txt
  ls | wc -l

(f) [8 points]
  cd ..
  cd - or cd so103class
  mkdir schlemiel

3. [40 points]

(a) [2 points] \{ print $0 \} or just \{ print \}

(b) [6 points] \{ if ($2 == 2) print \}

(c) [12 points] There are slicker ways, but this was shown in class:

\{
  sex = $2
  if (sex == 1) sexmales = sexmales + 1
  if (sex == 2) sexfemales = sexfemales + 1
\}
END {
  print sexmales, "males"
  print sexfemales, "females"
}

(d) [12 points]
  NR  Number of the current record.
  NF  Number of fields in the current record.
  $NF  The last field of the current record.
  $(NF-1) The next-to-last field of the current record.
  $2  The second field in the current record.
  $0  The (entire) current record.

(e) [8 points]
  BEGIN {
    for ( x = 3; x <= 10; x = x + 1 ) {
      print x
    }
  }