

Homework 6

Due: 4 March 2025

v20250225-1600

Input/output examples for some of these tasks are at the end of the handout. As with the previous assignment, you should know of the existence of the built-in functions `lc`, `sqrt`, `sort`, and `keys` (some of which are mentioned on the `perlintro` page but which have more documentation on the `perlfunc` page or via `perldoc -f`).

Problem 6.1 — practical

Quick reminder of some basic “hash” (i.e. map/dictionary) syntax:

<code>my %stuff = ();</code>	Construct an empty hash
<code>\$stuff{"datum"} = 5;</code>	Store a value associated with a particular key
<code>\$stuff{"datum"}</code>	Access a value associated with a particular key
<code>%stuff</code>	Refer to the entire hash

Write a perl script that reads in word/frequency pairs on alternating lines of input, stores them in a hash, and then produces a report that lists all the words in alphabetical order along with their *total* frequency. Words can appear multiple times with various capitalisations, and the total frequency should reflect all of them. For example, the input

```
the
25
brown
3
quick
5
brown
7
THE
5
```

would produce output that looked something like

```
brown    10
quick    5
the      30
```

The exact formatting on the output is not important as long as it's legible.

Problem 6.2 — practical

Write the following two short Perl programs using regular expressions to process their input and then print it back out. Note that in addition to simply matching using

```
if (/regex/) { ...
```

you can explicitly do a replacement *of* the matched portion using

```
if (s/regex/replacement/) { ...
```

which will definitely be helpful here. See documentation of regex generally in `perlre` and of this operator in particular in `perlop` under “Regex Quote-Like Operators”.

- a. Write a script whose output is identical to its input (file or standard input) except that on lines that start with tabs, the tabs are replaced with the corresponding number of spaces (eight per tab).
- b. In HTML, some people prefer uppercase tag names and others prefer lowercase. (Start tags are surrounded by angle brackets, and end tags are surrounded by angle brackets and preceded by a slash.) Write a script whose output is identical to its input (file or stdin) except that tag names are lowercased.

Problem 6.3 — practical

My outgoing personal emails always have a random quote at the bottom, which I generate with a script reading from a file. For this problem, write a script to perform a slightly simplified version of this task. The input to the script is a file of quotations. Its format is primarily lines of text separated by blank lines—quotations can be several lines long but will never contain a blank line within them. Other than the text itself and the blank lines that separate quotes, the only thing in the file are “comments”—if the first character on the line is a pound sign (`#`), the whole line is ignored.

Your script will read input in this format, select one quote from it at random, and print out that quote (not including any associated comment lines).

(Hint: you'll want to read each line and decide what to do with it, but you'll also need to remember what you've seen recently (i.e. are you in the middle of a quote) and over the longer term (i.e. what are all the quotes you've seen). That may help you decide what your variables look like.)

(To generate the random numbers, you'll want the `rand` function. You can look it up on with `perldoc -f`.)

EXAMPLES

For the main part of 6.2b, given a line like

```
<P>And I can be <EM>emphatic</EM> about many things (<STRONG>really
```

it would output

```
<p>And I can be <em>emphatic</em> about many things (<strong>really
```

For problem 6.3, here is some example input data:

```
"Maintenance, cleanup on aisle three please. We have a can of worms
opened on aisle three."                --Eva Schillace
```

```
#Rob gets his bluetooth phone and...
```

```
"Why are you so defensive about this? I *like* that you have neat cool
impractical things."                  --Angela Feraco
```

```
"I think sometimes we focus a little too much attention on whiz-bang
medical technology (like stents) while losing the broader picture that,
basically, the only things we have found to save people from the #1
#note: this line is a comment, but the previous line isn't!
killer is what you learned in second-grade nutrition lessons."
                                         --Keith Winstein
```

```
"Not only does the English Language borrow words from other languages,
it sometimes chases them down dark alleys, hits them over the head, and
goes through their pockets."          --Eddy Peters
```

Collaboration policy: collaborative. You each have to hand in your own version of the assignment, but you can talk to other people about the problems.

Mention in a comment or readme who you worked with. (Still no copying, though.)