

Week 3

Readings

Chapter 6

Chapter 7

Read Chapter 7 – “Grouping in Patterns” but do note that unless you are running Perl 5.10 you will not be able to use the new syntax for back references `\g{}` or the relative back reference (pages 111-112). Also note that without Perl 5.10 you will not be able to use the new character class shortcuts like `\h`, `\v`, or `\R` (top of page 115).

Key notes to keep in mind (AKA: how Perl differs)

- The key of a hash are converted to string so `5/2` is evaluated then converted to the string “2.5”
 - To use `5/2` as the key enclose it in quotes like “`5/2`” (page 95)
 - Although the keys must be strings, the values can be any combination of scalars
- Hash elements use `{}` around the key whereas array elements use `[]` around the subscript
- Like scalar variables and arrays, hashes have a leading prefix `%` (page 98)
- Hashes can be assigned into arrays and vice-versa , so there is the option to reverse a hash or other operations that can be performed on strings (page 99)
- If you are declaring a hash (or array) as local (using `my`) the entire hash must be declared
 - For example, use `my %hash` as you cannot declare individual parts local as in `my $hash{$key}` (page 270 sample solution to exercise 6.1)
- Take a look at the `perldoc perlre` page by typing `perldoc perlre` at the command prompt or visiting <http://www.perl.com/doc/manual/html/pod/perlre.html> for a quick reference to Perl’s regular expressions. Don’t worry if a lot of it looks new, a lot more of it will be explained next week in chapters 8 and 9.

Exercises

Book exercises from Chapters 6 and 7.

Write a program to read in a string from the command line followed by any number of given files, then print any lines in each file that contain the initial string. This is much like the unix `grep` command. The output should contain the file name, line number, and line that contains the given string for each match.

Some caveats:

- If the pattern string contains a space then it will need to be enclosed in quotes on the command line, otherwise everything after the space will be considered a file (this is because the `@ARGV` array is space delimited)
- If any metacharacters are used in the pattern and not escaped they may unexpectedly affect the regular expression