3
Perl Basics

Perl Intro

- Invented by Larry Wall in 1987 to make report processing easier
- We will be learning Perl 5, Perl 6 is a completely different language, not just the next major release after Perl 5.
- Command line based, you can use any text editor to write Perl scripts
- Perl gain popularity in the 90’s due to its excellent support for string processing and regular expressions.
- Perl culture - many unique characteristics such as perl poetry
- Perl is about getting things done and having fun at the same time.

Larry Wall

- Linguist who wanted to develop written languages for people who only had spoken languages
- Often compares Perl with a natural language and many of Perl’s designs were influenced by his linguistic rationale. (bless, kill, die, wait, sleep, etc)
- Responsible for many of the idiosyncrasies of the language, Perl culture
- Author of the Camel book
- Serves as benevolent dictator for life of the Perl project:
  - 1. Larry is always by definition right about how Perl should behave. This means he has final veto power on the core functionality.
  - 2. Larry is allowed to change his mind about any matter at a later date, regardless of whether he previously invoked Rule 1.   Got that? Larry is always right, even when he was wrong.

Perl Compilation

- The perl interpreter (/usr/bin/perl) first reads a perl script and constructs a parse tree.
- The parse tree is then translated into a bytecode file. This completes the compilation stage.
- It is this bytecode file that is then interpreted by the perl interpreter at runtime, the execution stage.
- Perl scripts are optimized by the compiler in much the same way as a C program is optimized during compilation.

Scalar variables

- A scalar is an individual unit of data
- Scalar variables are defined with the $ preceding their name
- Perl has three types of scalars:
  - Numbers - signed and unsigned integers, floating-point values, numeric strings
  - Strings - ‘single-quoted’, “double-quoted”, version strings
  - References - simplified pointers
- Scalar Context - many operations in Perl return a single value, so we say that it operates in scalar context, as opposed to list context

Array variables

- An array is a sequence of scalar values
- Arrays in Perl have zero-based indexing
- An array variable is defined with the @ preceding the name
- In Perl it is not an error to access an element outside of the array bounds, this simply returns the null object undef.
- It is also not an error to assign a value to an array element whose index is outside the bounds of the array. Perl simply adds it and any necessary intervening elements, which are set to undef.

Hash variables

- A hash is a key-value collection of data items
- A hash is defined with the % preceding the name of the hash
- Assigning hash values uses the => operator:
  - %mascots = (csu => “bucs”,
    cofc => “cougars”);
- Accessing the value of a key is done in scalar context: $mascots{ csu } would return “bucs”
- Arrays of the hashes keys and values can also be accessed:
  - @k = keys %mascots;
  - @v = values %mascots;

Regular Expressions

- Powerful string processing tool
- Matches regex patterns to input strings
- Optimized to quickly process large amounts of text
- Many uses including searching, data validation, data cleanup, etc
- Will talk about more after we have introduced Python

Terminal I/O

- STDIN - standard input stream (keyboard)
- STDOUT - standard output stream (terminal window)
- STDERR - standard error stream (terminal window)
- <> - line input, or diamond operator
- Scalar Context - only one line at a time will be read from STDIN

Writing and Running Perl

- There is no specific IDE you have to use with Perl. Any text editor will do fine.
- A few popular text editors are Sublime Text - https://www.sublimetext.com/, Atom - https://atom.io/, Brackets - http://brackets.io/, and Visual Studio Code - https://code.visualstudio.com/
- Perl scripts have a .pl file extension.
- Run a Perl script at command line by typing perl and then the name of the .pl file that contains your code (Ex: perl lab2.pl)

Some useful Perl commands

- which perl (gets the location of the perl interpreter)
- perl -v (get the version of perl)
- perldoc perl (perl documentation)

Reading Assignments

- Learn Perl in about 2 hour and 30 minutes - https://qntm.org/files/perl/perl.html (read up to the section titled “Built-in functions” about half-way down the page)
- Beginning Perl - http://web.archive.org/web/20120709053246/http://ofps.oreilly.com:80/titles/9781118013847/index.html
- Chapter 1 - http://web.archive.org/web/20120627121215/http://ofps.oreilly.com:80/titles/9781118013847/whatapostrophes_perlquestion_mark.html
- Chapter 3 - http://web.archive.org/web/20120617145412/http://ofps.oreilly.com:80/titles/9781118013847/variables.html

Additional Perl Resources

- https://www.perl.org/learn.html
- http://learn.perl.org/
- http://learnperl.scratchcomputing.com/tutorials/getting_started/
- Installing Perl on Linux - http://learn.perl.org/installing/unix_linux.html
- Llama Book - http://shop.oreilly.com/product/9780596520113.do
- Perl Beginners - http://perl-begin.org/
- Interactive Tutorial - http://www.learn-perl.org/
- Larry Wall talking about Perl - http://www.wall.org/~larry/pm.html