

Perl for Introductory Programming Courses

Jason Moore, Mitchell A. Thornton
Southern Methodist University
Dallas, Texas

Ronald W. Skeith
University of Arkansas
Fayetteville, Arkansas

Abstract:

While Perl stands for The Practical Extraction and Report Language, even a cursory examination reveals that it is far more extensive and had much more capability than a reporting language. The Perl programming language, developed by Larry Wall, is a flexible and a dynamic scripting language. Perl is referred to as a dynamic language because it uses an interpreter instead of a compiler. Perl programming language information is available at www.perl.com and www.activestate.com.

Perl runs on most platforms including Windows XP, Mac OS X, Linux, and UNIX allowing students to gain development experience using all of the major platforms. Perl comes bundled with Linux distributions and the version of Perl and other details of the installation can be determined from the command line by invoking the `$perl -v` and `$perl -V` commands in a Linux or UNIX environment.

*An excellent way to use Perl on a XP platform is to install the **cygwin** emulator which can be downloaded from www.cygwin.com. **cygwin** comes bundled with the latest version of Perl. **cygwin** operates as if it were a platform in a DOS window while using the XP operating system.*

Perl is updated and maintained by a development team referred to as the Perl Porters. A vast number of users contribute to CPAN, a public depository of software applications, documentation, binary distributions, and FAQ's. CPAN, the Comprehensive Perl Archive Network, maintains a website that houses and distributes free Perl materials. No other language provides the user with such extensive support.

*Perl can be used as a general programming language, a system administration tool, and as a scripting language to perform the functions of utilities such as **tr**, **sed** and **grep**. However, Perl can out-perform these filters in most cases because of the extensive regular expression capabilities it supports. Perl is an excellent teaching language since it so comprehensive and relatively easy to use to demonstrate the basics requirements of a programming language including web-development through Common Gateway Interface (CGI) scripts and Object Oriented Practices.*

Introduction:

The point of any programming class at the college level is to teach programming skills which include shell scripting, object oriented programming, text manipulation, and Internet web-development. However, in order to teach students the programming skills that they will need

time must be spent teaching them one or more programming languages i.e. syntax. The fewer languages that students must learn results in less time spent teaching syntax and more time spent teaching valuable programming skills. Therefore, languages that are versatile enough to cover several of these topics should be used. The choice of what language to use for an introductory programming class is in fact a greedy algorithm in which you choose the most versatile language. This greedy algorithm should yield the selection of the Perl programming language since Perl is a versatile enough language to cover all of the mentioned programming skills plus many more.

However, programming skills to be taught is not the only variable that should be taken into consideration. One must also consider whether or not students will have access to the programming language on their personal computers. Since students are likely to spend more time working on programs and projects that can be done on their computers as opposed to the computer lab on campus. Perl is available for free on all three major pc platforms, Windows, Linux, and Apple. Perl comes bundle with Linux and Apple and can be downloaded for free for Windows as part of the cygwin application.

In the next section of this paper, we will cover how to structure an introductory class using Perl programming language. In the last section, we summarize many of the items covered in this paper in the conclusion.

Course Outline:

A logical place to start when teaching students the syntax of Perl is to give them small text manipulation programs. In addition to learning the syntax of Perl, they will learn how to use regular expressions. Another good exercise for early in the course is to have the students write two similar programs on two of the major pc platforms and Unix. Most students will likely choose Windows and Linux. In addition to teaching them how to develop on multiple platforms, this will give them a chance to learn a common Unix text editor such as vi. The skills learned writing the small text manipulation programs can be used while doing the larger programs in the later part of the course. One example of how to ensure these skills are reused is to have parts future assignments to be earlier assignments. This not only tests how well students are learning from the previous assignments but also teaches code reuse.

A logical way to build on the simple text programs is to have the students write programs that download the text files to be manipulated from the internet. The key here is to find files from the internet that are similar enough not to require multiple programs but different enough to cause some issues. An advanced Object Oriented Perl programming class used sport rosters from other teams [?]. However, these rosters have since become the exact same format. A possible solution might be to have students use college rosters.

Through the use of Common Gateway Interface (CGI), Perl can be used for web development. Perl is also has a similar syntax to the web development language php making it easier for students to learn in future classes. A good initial assignment for web development is a web page counter. This assignment also gives the instructor time to make sure the students have

completely grasp the text manipulation since they will need it for the final project that combines text manipulation, web development, and Object Oriented process (OOP).

The final project is assigned in two parts. The first part is a Unified Modeling Language (UML) diagram and the second part is the working website. Collecting the first part of the project separately accomplishes two goals:

- Forces students to go through OOP methods before completing the project.
- Gives time to fix problems that students may be having with web development.

Figure 1 shows a potential student top level UML diagram.

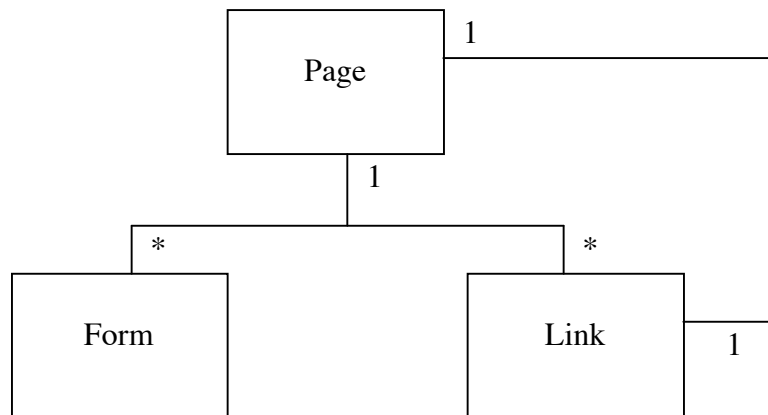


Figure 1: Top Level UML Diagram for the Project

The final project is a comment board. The comment board is to be structure as follows. A front page should contain a link to all of the threads or topics in the board. The last page is to contain comments from users and a form for users to add more comments. The project should be graded on the user interface and functionality. Partial credit is given for just getting the last level working.

Additional Help:

No matter how well the book is written or how well the course is taught some students will need additional help for various reasons. Perl has one of the most extensive online communities of any language. At www.perl.com, students can find articles on Perl in varying subjects of its 3-D capabilities to C to good programming practices such as better code through destructors. Also available at this site is a very comprehensive FAQ. The FAQ covers the basics of programming in Perl from how to write a “Hello World” program to how to do file input and output. More advanced topics are also covered in the FAQ such as accessing databases in Perl. This material is also available for download in both pdf and html format. [1]

While Perl can easily be programmed in a simple text editor, some may feel that they still need an integrated development environment (IDE) to make development easier. One such IDE

is the Komodo IDE from ActiveState. ActiveState also offers additional help on using the Perl language with their own FAQ and cookbooks. [2]

Conclusion:

When teaching an introductory programming course, one of the most important decisions that must be made is what programming language to use. The programming language must be easy learn with plenty of help available for the students and should be powerful enough for enough for students to continue using in more advanced classes. The Perl programming language satisfies both those needs with its extensive online community of users, FAQ, regular expressions, web development capabilities, and database features.

References:

[1] www.perl.com

[2] www.activestate.com