66-2230-01 Programming in Perl
Spring 1998, Wednesday 5-6:50 PM, March 18 - April 29, DCC 324

Instructor:	Louis Ziantz	T.A.:	Yu Jiang	Zhong Zhang
Office:	AE 427		Sage 1112	AE 316W
Office Hours:	Tue 12-2		Mon 10-12	Wed 2-4
	Fri 1-3		Wed 10-12	Thur 3-5
	and by appointment		and by appt.	and by appt.
Phone:	(no office phone)		x4849	x8057
E-mail:	ziantzl@cs.rpi.edu		yujiang@cs.rpi.edu	zhangz@cs.rpi.edu

Prerequisites: The formal prerequisite for this course is 66.110 or 66.112 (*Computer Science I*) which is currently taught using C++. However, knowledge of at least one high-level programming language (*e.g.*, C, Pascal, or FORTRAN) is also a satisfactory prerequisite. If you have no knowledge of any high-level programming language or have never taken an introductory course in computer science, I would advise *against* taking this course.

Course Home Page: `http://www.cs.rpi.edu/~ziantzl/perl.html`

Required Textbook:

Learning Perl, 2nd Edition
R. L. Schwartz and T. Christiansen, O'Reilly, 1997.
Programming Perl, 2nd Edition
L. Wall, T. Christiansen, and R. L. Schwartz, O'Reilly, 1996.

Test Schedule:

Date	Time	Description
Wednesday, April 8	5-6:00	Test
Wednesday, April 29	5-6:50	Exam

Grading: There will be a homework, two programming assignments, a test, and an exam. Weights are shown on the following chart:

Component	Percentage
Homework/Projects	50 %
Test	20 %
Exam	30 %

Grades will be assigned based on the following scale:

Score	Grade
90–100	A
80–89	B
70–79	C
60–69	D
0–59	F

General Course Policies:

- You are responsible for all material covered in class.
- Example code and other course material may be found on the course web page.
- Questions regarding the final course grade should be directed to the instructor.
- **Any grading dispute must be resolved within ONE WEEK after the graded work was returned to the class.**

Course Policies Regarding Exams:

- An exam may test any material covered in class, in the assigned reading, or associated with assignments.
- There is no collaboration of any kind allowed during exams.
- Students who are forced to miss an exam must notify the instructor **IN ADVANCE**. Valid reasons for missing an exam include a family emergency, serious illness, required travel with an institute sports team or club, or a letter of excuse from the Dean. A makeup exam will **not** be given in other circumstances.

Course Policies Regarding Programming Projects:

- A description of each graded programming project may be found on the course home page (once it is assigned).
- Projects will be submitted electronically. The specifications for submitting assignments may also be found on the same web page.

- If you use code from a public source, please include a comment with that code indicating where the original code may be found. Note that code written by any student at RPI other than myself or the TA is **not** to be used in your submission. The *discussion* of assigments with other students is encouraged, but the *implementation* of programs is to be the result of your own work. Any significant portions of code that are "identical" may result in punishment in accordance with the stated course policy for Academic Dishonesty.

- All projects will be tested on IBM workstations. It is recommended that you test your programs using the appropriate hardware and software (*i.e.*, perl on an IBM workstation). Allowances will not be made for failure to properly and **completely** test your programs. Do not modify your files after you submit them until you have received a grade back from the TA and you are satisfied that you have been graded fairly.

- Each student will be allowed **2 late days** to be used *without penalty*. Any available late days are automatically deducted before a penalty will be given. Partial days will be deducted as a full day.

  Once these days are consumed, late assignments will be subject to the following penalty:

  - Late up to 24 hours: **-20 points**
  - Late up to 48 hours: **-40 points**
  - Later than 48 hours: **no credit**

  Exceptions to this policy will be considered in the case of family emergency, serious illness, required travel of sports teams or clubs, or a letter of excuse from the Dean.

  I strongly suggest you start working on each assignment as soon as possible after it is given out.

Academic Integrity:

- The Rensselaer Handbook defines various forms of Academic Dishonesty and procedures for responding to them. All forms are violations of the trust between students and teachers. Students should familiarize themselves with this portion of the Rensselaer Handbook and should note that the penalties for plagarism and other forms of cheating can be quite harsh. Credit will not be received in **any** case of Academic Dishonesty and may result in failure of the course along with disciplinary action.

- At *no time* should you copy *any part* of another student's work or *permit someone else* to copy any part of your work. For example, work done on campus or other computers should not be made available so that another student could copy any part of your work. If at any time you are not sure what constitutes "appropriate" collaboration, it is *your* responsibility to clarify it with me.

Tentative Course Outline:

Date	Topics	Reading
Mar. 18	Introduction, Scalar Variables, Arrays	Chapt. 2-3 (L)
Mar. 25	Control Structures, Hashes	Chapt. 4-5 (L)
Apr. 1	Basic I/O, Regular Expressions	Chapt. 6-7 (L)
Apr. 8	TEST Regular Expressions	 pp. 57-74 (P)
Apr. 15	Filehandles, Functions, Classes	pp. 108-111 (L), Chapt. 8 (L)
Apr. 22	Hard References, CGI	pp. 244-251 (P), Chapt. 19 (L)
Apr. 29	EXAM	

The notation (L) means the reading comes from the *Learning Perl* textbook while (P) means the reading comes from *Programming Perl*. Note that *Programming Perl*'s main use will come as a reference. Chapter 3 lists and describes every Perl builtin function in alphabetical order, and Chapter 7 lists and describes standard Perl library modules. Chapter 9 explains the diagnostic messages generated by Perl during compilation or execution.