

Network Programming

Instructor: Dave Hollinger

Home Page:
www.cs.rpi.edu/~hollingd/netprog

Email: netprog@cs.rpi.edu

Course Home Page

- Announcements
- Homework Assignments
- Lecture Notes
- Links to required readings
- FAQs (for homework and tests)
- Links to web resources

Programming Environment(s)

- Unix workstation
 - Solaris, AIX, IRIX, Linux, BSD, ...
- All students will get an account on Computer Science Department Sun workstations (running Solaris).
- All grading will be done on CS Suns!

Homework Submission

- Submission will be via email.
- Date and Time of submission is when our mail server receives your submission.
- Automated system – you must follow the directions!

Homework (cont.)

- Directions for submission will be posted on the course home page with the first assignment.
- Submission guidelines will also be posted on the course web site
 - what we expect: comments, Makefiles, etc.

Email

netprog@cs.rpi.edu - for questions about homework or material covered in lecture.

hollingd@cs.rpi.edu - for questions about my kids or to find out what kind of cookies you can send me.

Email (cont.)

netprog-submit@cs.rpi.edu

- for homework submissions only!
- will be answered by a program (not a human).
- more details with the 1st assignment.

WebCT

WebCT will be used for many things:

- Links to course materials/assignments
- Grades
- Message Board
- Chat (during class and at scheduled times).

Course Topics

- Networking
- TCP/IP
- Sockets Programming
- The Internet and Internet protocols
- The WWW and Web programming
- Security
- Other Network Programming APIs

Grading

- 2 Tests 30%
- Projects 70%
- 7 projects!
- You must pass the tests to pass the course!
(test average must be 50%).

Netprog 2002 - Course Information

10

Project Grading

It is expected that everyone is a good programmer.

Comments are required.

Structured, readable code is required.

25% of the grade depends on the whims of the grader.
A quality of the code.

Netprog 2002 - Course Information

11

Code “quality”

- Is the code easy to understand?
- How hard would it be to make a small change to the functionality?
- Are all possible error situations handled?
- Can the code handle unexpected input?

↑
Security concern!

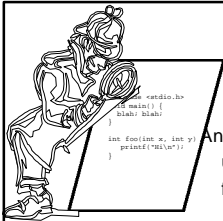
Netprog 2002 - Course Information

12

Academic Integrity

- All programming projects are to be done individually!
- Discussion is encouraged.
- No sharing of code in any form.

- Catching cheaters is my hobby!



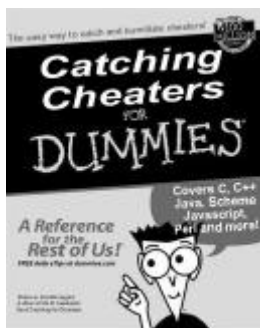
Really – I mean it!

Any projects submitted that are unreasonably similar will result in the following for ALL people involved:

- grade of 0 for the project
- and*
- a two-letter grade reduction in course grade.

Possible failure in the course.

Check out my new book:



Getting Help with Projects

- use netprog@cs.rpi.edu
- Discuss anything with anyone!
- DO NOT SHARE CODE IN ANY FORM.
- Feel free to use WebCT message board, but do not post code.

Lectures

Typically 60-90 minutes.

Questions/Comments are encouraged!

(otherwise Dave goes too fast...)

Remote Students:

- participate live via WebCT chat (or phone).
- please contribute!
- your real-world experience can make this a better course for local students!



Don't go to sleep!
(The *sleepcam* is always active)

- Ask questions!
- Correct Dave! *extra credit!
- Share anecdotes!
- Make suggestions!
- Tell jokes!

Use exclamation points!!!

Note for Remote Students

- Next lecture will include info on using CS workstation account:
 - how to access the account
 - review of a few Unix commands
 - file transfer
 - how to protect your files, etc.
