Computer Networks

CAS CS-555 – Spring 2000

http://www.cs.bu.edu/fac/byers/cs555.html

MWF 10:00 - 11:00 AM, CAS 313

	Instructor: Prof. John Byers	**Teaching Fellow: Stan Rost**
Office	MCS 280, Phone: 353-8925	MCS 209, Phone: 353-3326
Email	byers@cs.bu.edu	prgrssor@cs.bu.edu
Web	www.cs.bu.edu/fac/byers	-
Office Hours	M 2:00 - 3:30 & Th 2:00 - 3:30	Tu 3:00 - 5:00 & F 1:00 - 3:00
Lab Hours*	-	W 4:00 - 6:00

*: Lab hours will be held in the undergraduate programming lab when programming assignments have been assigned. Stan may also hold some of his regular office hours in the lab at these times.

Course Overview: This course serves as a graduate introduction to computer networks and is geared toward beginning graduate students and seniors. The course will investigate the design of computer networks and network protocols, from both a conceptual and engineering design standpoint. The primary focus of the course will be on the *software* used behind the scenes to build scalable, general-purpose networks.

Prerequisites: Protocols underlying networked communications are intricate and can be best understood by those with a broad background in both the systems-oriented and theoretical aspects of computer science. In particular, this course will assume intimate familiarity with CS 350 (fundamentals of computing systems) and MA 293 (basic discrete math and elementary probability theory). Also, students will be expected to have CS 330 (algorithms) or equivalent. Students without the prerequisites will not be allowed to enroll in the course.

Textbooks: The required text is Larry Peterson and Bruce Davie, *Computer Networks: A Systems Approach*, Morgan Kaufmann, 1999, available at the B.U. bookstore and at any online bookseller. Course materials will refer to the new second edition (blue cover) – please do not purchase the first edition (brown cover) used in years past. Another useful text which covers much of the material from a broader perspective is S. Keshav, *An Engineering Approach to Computer Networking*, Addison-Wesley, 1997. A small amount of supplemental reading will be assigned and distributed later in the course.

Workload: Be forewarned – the workload in this course will be heavy. To master the conceptual material covered in lecture, there will be written homework assignments due every other week in class (likely due on alternating Wednesdays). In addition, there will be two substantial programming assignments (written in a language of your choice), each of which is likely to amount to a thousand lines of code. If you are worried that your skills in writing and debugging large programs are weak, this course may not be for you. (On the other hand, this course can help you greatly improve those skills).

Exams: There will be a one-hour in-class midterm held during the week before spring break, likely Wed. March 1. The final will be held during the normal final exam slot: Friday, May 12 from 9 to 11AM. Please make your end-of-semester travel plans accordingly.

Grading: The course grade will break down as follows: 25% written assignments, 25% programming projects, 20% midterm, 30% final. The mean grade in this course in 1999 was "B".

Topics: We will no doubt drift from any formalized plan, but a rough schedule of where we are headed is provided in the pages that follow and a more detailed schedule of where we have been will be maintained on the course homepage.

Late Policy: Written homework assignments will not be accepted after their in-class due date, except in the event of illness documented by a doctor's note. Programming assignments will be accepted up to 48 hours after the submission deadline with a 10% deduction. In the event of serious, documented illness, makeup examinations will be given orally. Incompletes for the course will not be granted.

Academic Conduct: Academic standards and the code of academic conduct are taken very seriously at our university. Please take the time to review the CAS Academic Conduct Code if you are unfamiliar with its contents. The work that you submit must be your own original work – while it is acceptable to collaborate on written homework assignments, your writeups must be generated independently. For programming assignments, it is not acceptable to share code with your peers (although at my discretion, I may post some helpful routines on the website). Programming assignments will be tested for originality with an automated software tool.