Texas A&M University
College of Engineering
Computer Science Department

CPSC 612 Computer Networks and Distributed Processing
Spring Semester 2003
SYLLABUS

Instructor:	Dr. Michael E. Thomadakis
Office-Hours:	TBA, or by appointment
Location:	H. R. Bright Building, room 315B
e-mail:	miket@cs.tamu.edu
Tel:	845-3907

1 Course Information

CPSC 612 is an introductory but comprehensive graduate level course on the fundamental theoretical and technological aspects of computer networks and inter-networking. No previous knowledge on computer networks is assumed. Students will be exposed to material that will help them prepare for more advanced studies in computer networks. Graduate students can benefit by learning how to taylor system building and evaluation techniques on particular application areas of interest to them.

Both design and analysis methods will be emphasized. Several mini-projects will be assigned to corroborate the in-class lectures and allow students to develop advanced networking skills. The development work will involve TCP/IP Berkeley sockets (or WinSocks 2.0), experimental evaluation of intra-domain and inter-domain routing protocol, and experiments with basic quality of service architectures. The mini-projects will be carried out on the UNIX/PC network and the Networking Laboratory at the Computer Science Department. The laboratory part complements the material covered in the lectures and it will allow students to obtain hands-on experience in using state-of-the-art equipment.

Familiarity with an imperative programming language such as C/C++ is expected. Knowledge or familiarity with basic operating systems mechanisms or programming are not required but will be helpful.

Note that CPSC612 can be taken instead of CPSC463 or ELEN602

Official Course Site and Newsgroup Visit often

- http://courses.cs.tamu.edu/miket/cpsc612/ and,
- news:tamu.classes.cpsc612

for official announcements and up-to-the-minute information. Visit often both sites.

Course Catalog Description Basic hardware/software, architectural components for computer communications; computer networks, switching, routing, protocols and security; multiprocessing and distributed processing; interfacing operating systems and networks; case studies of existing networks and network architectures.

Course Objectives CPSC 612 has the following goals

- study the design, implementation and pragmatics of computer networks, their protocols, services, and distributed applications;
- emphasize in a balanced way both theory, applications and pragmatics;
- provide hands-on experience on networking applications and on setting up and tuning networking equipment to provide installation objectives;
- let students improve their network design, modeling and distributed processing skills.

Prerequisites: CPSC 410. Familiarity with at least one imperative programming language, such as, `C/C++`. Familiarity with operating systems (*e.g.*, UNIX), basic machine organization, data structures and algorithms would be helpful but is not required.

Main Textbooks

- (**Required for 612**) [GaWi2000] Alberto Leon-Garcia and Indra Widjaja, *Communication Networks: Fundamental Concepts and Key Architectures*, First Edition, McGraw-Hill Companies, Inc., 2000, ISBN 0-07-022839-6. Publisher's URL: `http://www.mhhe.com/leon-garcia`.
- (**Reference # 1 for 612** strongly recommended) [RSt1997] W. Richard Stevens, *Unix Network Programming, 2nd Edition, Volume 1; Networking APIs: Sockets and XTI*, Prentice Hall, October 1997, ISBN 0-13-490012X.
- (**Reference # 2 for 612** recommended) [WSt2002] William Stallings, *High-Speed Network TCP/IP and ATM Design Principles*, 2nd Edition, Prentice Hall, November 2001, ISBN 0-13-032221-0.

2 Course Topics

All the course material will be covered in the lectures. The specific contents will be taylored to emphasize the different interests of the student audience.

This course will examine the following topics, which may not be covered in this order, or not at all if time does not permit it.

- [**Ch 1 and 2: Introduction to Computer Networks and Inter-networking**] The ISO and TCP/IP models. Basic services and functions of each layer; basic network topologies; framing, naming, multi-plexing, de-multiplexing, error control, flow control, congestion control, routing; connection-oriented and connectionless services.
- [**Ch 3: Physical Layer**] Introduction to physical media employed in networks; properties, impairments and fundamental limits in the transmission of data; channel capacities and data bandwidth; guided and unguided media (copper, fiber-optic, wireless transmission, *etc.*); channel sharing (FDM, TDM and statistical TDM); introduction to WANs and data communication infrastructures for long-haul networks.
- [**Ch 5: Data-Link Layer**] Services, operations and design issues of the DLL. Flow and Error Control techniques; stop-and-wait, sliding window flow-control, stop-and-wait, go-back-N, selective reject ARQs; basic performance analysis; point-to-point and multi-point configurations topologies; binary and character based DLLs; PPP, HDLC.

- [Ch 6: **LANs**] Introduction to Local Area Networks technologies; MAC sub-layer; channel sharing and access techniques; contention *vs.* non-contention protocols; Aloha (pure and slotted), CSMA, CSMA/CD; Ethernet, 802.3, 100 mega-bit and 1 giga-bit Ethernet; token-ring, 802.5, FDDI; basic performance analysis of Aloha, CSMA/CD, token-ring and FDDI; introduction to ATM technology; critical discussion and comparison of the various technologies.

- [Ch 5 **and** 6: **LLC**] Logical Link Control; service access points, TCP/IP and other protocol encapsulation; networking devices, repeaters, hubs, bridges, LAN switches; LAN concatenation via bridging and switching; source routing, learning, transparent and remote bridges.

- [Ch 7: **Network Layer and Packet Switching**] Fundamental design problems, services, functions of the network layer; virtual-circuit and datagram service; naming, critical discussion of Distance Vector (DV) *vs.* Link-State (LS) routing algorithms; rate, flow and congestion control; IP Layer structure and functions; IP addressing, IP sub-protocols, ICMP, ARP/RARP (DLL protocol assisting IP); MTU, fragmentation and re-assembly; introduction to multi-casting; DHCP, BootP, packet tracing, tools.

- [Ch 7: **Inter-networking**] Basic inter-networking issues. Autonomous systems; intra-domain routing (ES-IS, IS-IS, RIP, OSPF), inter-domain routing (IDRP, BGP) protocols and problems; host and router architectures for IP; classful and classless addressing and routing; introduction to multicast routing, IGMP; IP over ATM.

- [Ch 8: **Transport Layer**] Introduction to design issues and functions of transport layers; addressing, flow-control and congestion control; UDP and TCP in TCP/IP stack, ATM ABR; connection establishment, data transfer and connection termination for TCP; flow-control and congestion avoidance in TCP and its different variants; OS implementation issues.

- [**Application and Presentation Layer**] Representation problem, ASN.1, encodings; Protocols using the TCP/IP stack, including, DNS and SNMP, RPC, NFS, HTTP, distributed systems, CORBA, security issues.

- [Ch 9 **and 10 (partial): Recent Technology**] Introduction to IPv6 (Next-Generation IP) [it is expected that within the next few years the entire networking infrastructure will be IPv6 based. All system vendors have started shipping computers and routers with IPv6 support.] Time permitting we will discuss basic Internet Service Architectures (Integrated and Differentiated Services); RSVP; Quality of Service problems; ATM.

- [[**RSt**] **Host Design Issues and Operating Systems**] We will survey design methodologies for practical networking platforms. Operating system programming methodologies. Multi-process and multi-threaded systems, concurrent, iterative and multi-threaded servers. Berkeley sockets on UNIX (or WinSocks on PC) will be used throughout the course for network programming assignments. We will discuss TCP and UDP sockets and IP multi-casting in parallel with the regular lectures. The discussion will focus on basic BSD programming, as well as, in host design issues; structure of kernel systems supporting networking, DLPI, TLPI, STREAMS.

Table 1: Course Requirements and Corresponding Weight

1.	mini-projects	40%
3.	1 mid-term exam	25%
4.	A final exam	35%

Table 2: Grading Brackets, V is your effective class average

$V \geq 90$	A
$80 \geq V < 90$	B
$70 \geq V < 80$	C
$60 \geq V < 70$	D
$V < 60$	F

3 Grading Scheme and Course Requirements

3.1 Grading Scheme

The course consists of lectures, reading, homework, programming assignments and examination periods. The weights and the grade brackets are shown in Table 1 and 2, respectively.

3.2 Course Requirements

3.2.1 Assignments

Assignment handouts will be placed on the course web page ahead of time.

Mini-projects are development assignments that will typically require about on or two weeks each to complete. There will be a number of mini-projects, each carried out by a group of two to four people. You are encouraged to form groups early. The mini-projects will involve programming in `C/C++` language on a UNIX system. You will need to have a student computer account.

Alternatively, students with interests in specific topics (*e.g.*, real-time networks, distributed processing, routing, *etc.*) can substitute the mini-projects with their own projects, after obtaining approval by the instructor.

3.2.2 Examinations

All major examinations will be held in class with exact dates determined in class.

3.3 Excused Absences and Make-up Policy

Make-ups for assignments and exams will be given only under circumstances beyond student's control (*a university sanctioned excuse*). Prior arrangements with the instructor must be made when feasible and official verification of circumstances necessitating the absence will be required. The correct approach is to start working on assignments

as early as possible and contact us when you encounter difficulties. In general, the closer to the deadline you request our assistance, the harder may be to obtain our help.

3.4 Submission of Work

All assignments deliverables must be submitted by the due date, electronically, using the `turnin` method. The web page for the course explains in detail how `turning` is used.

4 Teaching Personnel and Resources

The Instructor and the Teaching Assistants are you most valuable resource. We are available during our published office hours or at other times by appointment. Walk in any time to see the instructor or the TAs. Students are strongly encouraged to discuss with your TA or the Instructor issues concerining the material or the assignments. Contact the TAs or the Instructor to collect your scores of all graded items. Observe the re-grade period mentioned above.

The Instructor and the TAs will provide all necessary resources for the course to progress smoothly, including, handouts, notes, programming code files, among others. Most items are posted to the web page for the students to down-load.

People having difficulties with either the material or finishing assignments, are strongly encouraged to discuss the problems with the Instructor and the TAs *as early as possible*. Things can be corrected when problems are addressed early. The Instructor will propose a recovery plan for the student. The recovery plan is a course of action that will assist the student boost his understanding of the material and perform better.

5 Scholastic Dishonesty

Plagiarism is the passing of someone else's work as one's own, without giving the original author due credit. Scholastic dishonesty will be treated very strictly as per Texas A&M University rules.

A Other Reference Textbooks

During the course of our lectures we will occasionally draw upon material from several other sources. The following books will be used only as references. They provide good background reading but are not mandatory.

- [CN] Larry Peterson and Bruce Davie, *Computer Networks: A Systems Approach*, 2nd Edition, Morgan Kaufmann Publishers (Academic Press imprint), 2000. [This text provides a clear discussion on the upper layers of the ISO and TCP/IP models and Inter-networking.]
- [DCC] William Stallings, *Data and Computer Communications*, 6th Edition, Prentice Hall, 1999. [This text has a very concise and technical discussion of the lower three layers of the ISO model, of LANs, and of the IP Module on TCP/IP stack.]

- Richard Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Addison-Wesley, January 1994, ISBN 020163346.
- W. Richard Stevens, *TCP/IP Illustrated, Volume 2: The Implementation*, Addison-Wesley, January 1995, ISBN 020163354X.
- W. Richard Stevens, *Advanced Programming in the UNIX Environment*, Addison-Wesley, May 1992, ISBN 0201563177.
- Dimitris Bertsekas and Robert Gallager, *Data Networks*, 2nd Edition, Prentice Hall, 1992. [This text is a classic introduction to advanced treatment of networks. Discusses very concisely performance of networks and routing and congestion control algorithms. Dated but still very relevant.]
- Jean Walrand, *Communications Networks: A First Course*, Second Edition, McGraw-Hill, 1998.
- (**463 Required**) [KuRo2001] James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet*, Addison-Wesley Longman, 2001, ISBN 0-201-47711-4.
- Andrew S. Tanenbaum, *Computer Networks*, Third Edition, Prentice Hall, March 1996, ISBN 0133499456.

Supplemental Textbooks The following classics on stochastic and performance analysis are recommended to those pursuing a more rigorous performance evaluation investigation of networks (or other systems).

- Erhan Cinlar, *Introduction to Stochastic Processes*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1975. [Concise introduction to stochastic processes and queuing theory. Recommended starting point for serious work on stochastic systems.]
- Leonard Kleinrock, *Queuing Systems Volume I: Theory*, John Wiley & Sons, 1975. [Relatively comprehensive introduction to queuing systems.]
- Leonard Kleinrock, *Queuing Systems Volume II: Computer Applications*, John Wiley & Sons, 1976. [Applications of queuing theory to computers.]