CSCE 515: Computer Network Programming

1. Course number and name: CSCE 515: Computer Network Programming

2. Credit: 3-hrs; Contact: 3 lecture periods of 50 minutes or 2 periods of 75 minutes per week

3. Instructor: Xu, Nelakuditi, Matthews

4. Textbook: W. Richard Stevens, B. Fenner, A. M. Rudoff, *UNIX Network Programming: The Socket Network Programming API, Volume 1,* 3rd Edition.
    W. Richard Stevens, *TCP/IP Illustrated, Volume 1: The Protocols,* Addison-Wesley, Boston, MA, 1994.

5. Specific course information
    a. Catalog description: Computer networks and communication protocols; socket programming; interprocess communication; development of network software; case studies.

    b. Prerequisites: CSCE 311

    c. CSCE 5xx elective

6. Specific goals for the course
    a. Specific outcomes of instruction are that students will be able to:
        1. Demonstrate mastery of common network protocols: ARP, RARP, Ethernet, IPv4, IPv6, ICMP, TCP, UDP, DNS, HTTP, FTP, SNMP, and SMTP
        2. Demonstrate mastery of socket programming in C++ and Java
        3. Develop network applications such as ftp, remote login (telnet, ssh), web servers, network DB such as napster, and network games
        4. Demonstrate mastery of UNIX and Windows networking commands and managment: netstat, ifconfig, ping, traceroute, tcpdump, sock, telnet, rlogin

    b. As an elective this course cannot be counted upon to contribute to the attainment of any student outcome.
7. Topics covered and approximate weight (14 weeks, 4 hours/week, 56 hours total)
    1. Introduction, terminology, OSI and TCP/IP reference models, layered architecture, data link layer: protocols, Ethernet, bridges (6 hours)
    2. Protocols: IP, ICMP, TCP, UDP, FTP, PPP, ARP, RARP, DNS, SNMP, SMTP, NFS (8 hours)
    3. Networking Commands: netstat, ifconfig, ping, traceroute, tcpdump, sock, telnet, rlogin (8 hours)
    4. Berkeley sockets in C/C++ and Java, UNIX and NT, including both TCP and UDP sockets(6 hours)

5. UNIX concepts supporting network programming: processes, threads, signals, domain name system support, advanced I/O including timeouts and selection, daemons and Inetd (6 hours)
6. Application Layer: security, SNMP, SMTP, HTTP (5 hours)
7. Reviews and examinations (3 hours)

c.

Computer Engineering

Relation of Course Outcomes to EAC Student Outcomes*

Course Objectives	Program Outcomes										
	1. Logic & Math	2. Computing Fundamentals	3. Apply Computing Principles	4. Work on teams	5. Communicate Effectively	6. Liberal arts & Soc. Sciences	7. Basic Science and Lab Procedures	8. Learn New Tools & Processes	9. Employed upon Graduation	10. Application Area	11. Electronics and Digital Sys Design
1. Demonstrate mastery of network protocols		1						3	3		
2. Demonstrate mastery of socket programming in C++ and Java			3					3	3		
3. Develop network applications			3					3	3	3	
4. Demonstrate mastery of Unix and Windows networking commands		1						3	3		

* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related

Estimated Computing Category Content (Semester hours):

Area	Core	Advanced	Area	Core	Advanced
Algorithms			Data Structures		
Software Design		2	Programming Languages		
Computer Architecture		1			

Estimated Information Systems Category Content (Semester hours):

Area	Core	Advanced	Area	Core	Advanced
Hardware and Software			Networking and Telecommunications		3

Modern Programming Language			Analysis and Design		
Data Management			Role of IS in an Organization		
Quantitative Analysis			Information Systems Environment		