

CPSC 5157G – Computer Networks

Fall 2013 (Online and Onsite)

Instructor

Dr. Jianhua Yang
Center for Commerce and Technology 440
(706) 507-8180
e-mail: yang_jianhua@columbusstate.edu
website: <http://csc.ColumbusState.edu/yang>

Office Hours: TBA

Class Meetings:

TR: 3:00pm-4:15pm (for onsite only)

Course Prerequisites

A knowledge of Data Structures covered in CPSC 2108.

Textbook: Computer Networking: A Top-Down Approach
6th Edition
James F. Kurose & Keith W. Ross
Pearson / Addison-Wesley, © 2011.
ISBN-10 0 – 13 – 285620 – 4
ISBN-13 978 – 0 – 13 – 285620 – 1

Other Required Materials: The student will be required to download and use a number of freeware computer tools, such as Ethereal (Wireshark).

Additional textbooks and References

1. Andrew S. Tanenbaum , Computer Networks, Prentice Hall.
2. W. Stallings. Data and Computer Communications. Prentice Hall.
3. J. Walrand. Communication Networks: First Course. Aksen Associates.
4. D. Comer. Internetworking with TCP/IP, Volume I, Prentice Hall.
5. W. Stevens. TCP/IP Illustrated: The Protocols, Vol 1. Addison-Wesley.
6. D. Comer. Computer Networks and Internets. Prentice Hall.
7. L. Peterson and B. Davie. Computer Networks: A Systems Approach. Morgan Kaufman.

Course Catalog Description:

Local area networks, wide area networks, and internets. Protocols and the ISO Open Systems Interconnect reference model. Design, analysis, and performance evaluation. Emphasis on data link, network, and transport protocols.

Course Description

This course is a broad introduction to networking concepts and emphasizes the following:

1. Local area networks, wide area networks, and internets.
2. Protocols and the OSI (Open Systems Interconnect) reference model.
3. Design, analysis, and performance evaluation.

Course Topics

1. The large-scale structure of the global Internet and its associated protocols;
2. The OSI Reference Model and the Internet Protocol Stack;
3. The Application Layer: Structure and sample applications;
4. The Transport Layer: Its function and services;
5. Reliable data transport and congestion control;
6. The uses of UDP and TCP; the trade-offs between them;
7. The Network Layer: Its function and services;
8. The structure of IP addresses: classful and classless addresses;
9. The DNS (Domain Name Service): “User friendly” names & IP addresses;
10. Routing on the Internet;
11. The Link Layer: Its function and services;
12. Error detection and correction techniques;
13. Collision detection and the Ethernet solution to that problem;
14. Collision avoidance and the Token Ring solution to that problem;
15. Link-layer switches: forwarding and filtering;

Learning Objectives

1. Become familiar with layered communication architectures (OSI and TCP/IP).
2. Understand the client/server model and key application layer protocols.
3. Learn sockets programming and how to implement client/server programs.
4. Understand the concepts of reliable data transfer and how TCP implements these concepts.
5. Know the principles of congestion control and trade-offs in fairness and efficiency.
6. Learn the principles of routing and the semantics and syntax of IP.
7. Understand the basics of error detection including parity, checksums, and CRC.
8. Know the key protocols for multimedia networking including IntServ and DiffServ for IP.
9. Familiarize the student with current topics such as security, network management, sensor networks, and/or other topics.
10. Increase the ability to write a report to a project or a hand-on exercise and make a presentation.

Learning Outcomes

Upon successful completion of the courses in this discipline, the student will:

1. Demonstrate oral and written communication skills and increase ability to be effective team members.
 - Strategies:
 - Study of current research topics in computer network
 - Use Wireshark hands-on exercise report writing and presentation
 - Write team project report and make a presentation
 - Summarize research paper in computer network and make a presentation
 - ABET Criteria: A, B, C, F, I, and J
 - CS Program Objectives: 2, 3, and 4
 - Assessment: Project, hands-exercise report writing, research paper summary, oral presentation.

2. Demonstrate the basic knowledge of intranet, internet, and the Internet.
 - Strategies:
 - Study chapter 1 of the textbook
 - Conduct hands-on exercise to design a and set up a local area network
 - Exercise tools to explore intranet and the Internet infrastructure
 - ABET Criteria: A, B, and C
 - CS Program Objectives: 2, 3
 - Assessment: Written assignment, Hands-on lab report, quiz, and network online tools practicing report

3. Demonstrate the knowledge of computer network programming, such as socket programming.
 - Strategies:
 - Study chapter 2, and 3 of the textbook
 - Conduct hands-on socket programming on TCP and UDP
 - Exercise Wireshark hands-on lab
 - ABET Criteria: A, B, C, I and J
 - CS Program Objectives: 2, 3
 - Assessment: Programming assignment, Programming Project, Hands-on labs

4. Demonstrate the understanding of internet model, or OSI model and the TCP/IP protocol suit.
 - Strategies:
 - Study chapter 2, 3, 4, 5
 - Use Wireshark hands-on labs
 - Learn network online tools
 - Conduct computer network projects
 - Read some TCP/IP related research papers

- Read protocol-related RFC files
 - ABET Criteria: A, B, C, E, I, J, and K
 - CS Program Objectives: 2, 3
 - Assessment: Writing assignment, project, exam, hands-on labs
5. Demonstrate the ability to apply computer network knowledge to solving a real world application.
- Strategies:
 - Study one of two of chapter 6, 7, 8, 9 of the textbook
 - Conduct a comprehensive project
 - Conduct a short term internship if possible
 - Conduct a field trip to local IT company, campus IT center
 - ABET Criteria: A, B, C, D, E, F, I, J, and K
 - CS Program Objectives: 1, 2, 3, 4
 - Assessment: Project report, presentation, internship report, or field trip report.

Course Methods

1. The class will be a lecture course, taught face to face.
2. Laboratory experiences will be part of the course.
3. Students will be expected to complete hands-on exercises and a series of programming assignments.

Student Responsibilities

1. Attend class regularly (if so enrolled) and participate in all class discussions.
2. Complete all reading assignments and all homework assignments.
3. Ask the instructor questions.
4. Send the instructor e-mail with both comments and questions.

Instructor Responsibilities

1. Give lectures on the course material.
2. Assign appropriate homework that illustrates the concepts of the course, and grade and return the homework in a timely manner with adequate explanation.
3. Give tests over the material and grade and return the tests in a timely manner
4. Provide a website that supports the course.
5. Provide at least four hours of office time primarily designated for assistance of students in this class, at times expected to be convenient for the students. It is expected that the instructor be available to the students during these hours.
6. Reply promptly (within one business day) to all student e-mail communications.

Methods for Evaluating Students

The evaluation methods will include homework and a comprehensive final exam. In addition there is a “class participation” component to the grade.

Assignments	10%
Hands-on labs	10%
Programming Project	30%

Online Quiz	10%	
Online Discussion	10%	
Mid-Term Exam	10%	
Final Exam	20%	When scheduled by the University Registrar.

The Exams: All exams will be given in class. They may be either open–book or closed–book, depending on the context. The time allocated will be the standard.

The Mid–Term Exam will be a one–hour exam.

The Final Exam will be a two–hour exam.

Assignment of Letter Grades

The method of assigning letter grades based on overall course averages is fairly standard. The basic method for assigning grades is described as follows:

Average	Letter Grade
90 – 100	A
80 – 89	B
70 – 79	C
60-69	D
Below 60	F

Tentative Topic Schedule (online and onsite)

(the due date is tentative and subject to change. For exact due date and time, please check D2L)

Week	Date	Topics to be covered	Ch	Asgn	Lab	Quiz
1	08/13	Introduction to the class				
	08/15	The Internet and network edge	1			
2	08/20	The Network Core Delay, Jitter, Loss, & Throughput	1			
	08/22	Protocol Layers History	1			
3	08/27	Network Applications The Client / Server Model The Web and HTTP	2	A1		Quiz 1
	08/29	File Transfer: FTP Electronic Mail: SMTP	2		Lab 1- HTTP	
4	09/03	The Domain Name Service Peer-To-Peer Applications Introduction to TCP and UDP Socket Programming with TCP Socket Programming with UDP End-to-end vs. host-to-host	2			
	09/05	Transport layer vs. Network Layer Connectionless Transport: UDP	3	A2		Quiz 2
5	09/10	Reliable Data Transfer Connection-Oriented Transport: TCP	3		Lab2- DNS	
	09/12	Reliable Data Transfer Flow and Congestion Control	3			Quiz 3
6	09/17	Connection Management The TCP “Handshakes”	3			Quiz 4
	09/19	Forwarding and Routing Real circuits vs. virtual circuits	4	A3		
7	09/24	Routers and Switches	4			
	09/26	Format of an IP packet IP version 4 address format	4		Lab 3 TCP	
8	10/01	Midterm review				
	10/03	Midterm				
9	10/08	Fall break, no class				
	10/10	go through the Midterm				
10	10/15	Classful and classless addresses CIDR (Classless Interdomain Routing)IP version 6	4			Quiz 5

	10/17	Routing algorithms More on DNS	4			
11	10/22	DHCP (Dynamic Host Configuration) NAT (Network Address Translation)	5	A4	Lab 4-IP	Quiz 6
	10/24	Link Layer and Network Layer Link layer services Error Detection and Error Correction	5			
12	10/29	Managing a Shared Channel The Ethernet and the Alohanet Link Layer Addressing	5			
	10/31	Ethernet Addressing CSMA/CD Ethernet vs. Token Ring	5			
13	11/05	Principles of Security	8	A5		Quiz 7
	11/07	Message Integrity and End point authentication	8			
14	11/12	Securing E-mail SSL	8			
	11/14	IPSec and VPN Securing Wlan	8			
15	11/19	Firewall and IDS	8		Lab 5-Security	
	11/21	Firewall and IDS	8	A6		Quiz 8
16	11/26	Last day class (final review)				
	11/28	Thanksgiving				

Course Assistance

Student assistants in the Computer Center can help you with basic computer-related problems (such as logging on to the network, saving your work, etc.), but they are not trained to help you with your assignments. We have several tutors at the Department of Computer Science who can help you with programming assignments. Their schedule is posted at the department office. You can always contact me during the posted office hours or by appointment. The best way to get in touch with me is by e-mail at yang_jianhua@columbusstate.edu.

Dropping the Course

We hope that you will complete the course and profit from it. If it is necessary for you to withdraw from the course during the semester, you must follow all official CSU procedures for withdrawing. It is not sufficient to notify the instructor; you must use the ISIS system and withdraw officially. For details on how to withdraw from a course, see

the web page

<http://aa.columbusstate.edu/advising/w.htm#Withdrawal%20from%20a%20Course>.

I would appreciate it if you were first to consult with me before starting the procedure for withdrawing from the course. In some cases, we can agree on an arrangement that will allow you to complete the course with minor adjustments.

Academic Honesty

Academic dishonesty includes, but is not limited to, activities such as cheating and plagiarism (<http://aa.columbusstate.edu/advising/a.htm#Academic Dishonesty/Academic Misconduct>). It is a basis for disciplinary action. Any work turned in for individual credit must be entirely the work of the student submitting the work. All work must be your own. You may share ideas but submitting identical assignments (for example) will be considered cheating. You may discuss the material in the course and help one another with debugging; however, any work you hand in for a grade must be your own. A simple way to avoid inadvertent plagiarism is to talk about the assignments, but don't read each other's work or write solutions together unless otherwise directed. For your own protection, keep scratch paper and old versions of assignments to establish ownership, until after the assignment has been graded and returned to you. If you have any questions about this, please see me immediately. For assignments, access to notes, the course textbooks, books and other publications is allowed. All work that is not your own, **MUST** be properly cited. This includes any material found on the Internet. Stealing or giving or receiving any code, diagrams, drawings, text or designs from another person (CSU or non-CSU, including the Internet) is not allowed. Having access to another person's work on the computer system or giving access to your work to another person is not allowed. It is your responsibility to keep your work confidential.

No cheating in any form will be tolerated. Penalties for academic dishonesty may include a zero grade on the assignment or exam/quiz, a failing grade for the course, suspension from the Computer Science program, and dismissal from the program. All instances of cheating will be documented in writing with a copy placed in the Department's files. Students will be expected to discuss the academic misconduct with the faculty member and the chairperson. For more details see the Faculty Handbook: <http://aa.columbusstate.edu/faculty/FacHandbook0203/sec100.htm#109.14> and the Student Handbook: <http://sa.columbusstate.edu/handbook/handbook2003.pdf>

ADA Accommodation Notice

If you have a documented disability, as described by the Rehabilitation Act of 1973 (P.L. 933-112 Section 504) and the Americans with Disabilities Act (ADA) and subsequent amendments and would like to request academic and/or physical accommodations, please contact the Office of Disability Services in the Schuster Student Success Center (room 221), 706-507-8755, as soon as possible. Course requirements will not be waived, but reasonable accommodations may be provided as appropriate. **It is then your responsibility to contact and meet with the instructor. It is also your responsibility to present the instructor with a letter from the Center for Academic Support. Without this letter detailing the required accommodations, the instructor cannot help you.** The Center for Academic Support can assist you and the instructor in

formulating a reasonable accommodation plan and provide support in developing appropriate accommodations for your disability. Course requirements will not be waived but accommodations may be made to assist you to meet the requirements. Technical support may also be available to meet your specific need. For more information on services and support available, refer to http://uc.columbusstate.edu/disability_services.htm.

ABET Criteria:

- A. An ability to apply knowledge of computing and mathematics appropriate to the discipline;
- B. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
- C. An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;
- D. An ability to function effectively on teams to accomplish a common goal;
- E. An understanding of professional, ethical, legal, security, and social issues and responsibilities;
- F. An ability to communicate effectively with a range of audiences;
- G. An ability to analyze the local and global impact of computing on individuals, organizations and society;
- H. Recognition of the need for, and an ability to engage in, continuing professional development;
- I. An ability to use current techniques, skills, and tools necessary for computing practice.
- J. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
- K. An ability to apply design and development principles in the construction of software systems of varying complexity.

CS Program Objectives:

Our graduates will have achieved:

- 1) a broad general education assuring an adequate foundation in science and mathematics relevant to computing.
- 2) a solid understanding of concepts fundamental to the discipline of computer science.
- 3) good analytic, design, and implementation skills required to formulate and solve computing problems.
- 4) the ability to function and communicate effectively as ethically and social responsible computer science professionals.