Objective:
This course is an introductory course on computer networks. This course introduces the underlying concepts and principles of modern computer networks with emphasis on protocols, architectures, and implementation issues. The main goal of this course is to understand layering in computer networks, understand different protocol stacks (OSI and TCP/IP), understand functions and protocols within a layer, understand how layers fit together and finally understand how the Internet works.

Catalog Description:
ISO model, protocol layers, circuit/packet switching, sockets, reliable transport, congestion control, routing, addressing, switching, multiple access, error correction, coding, and digital modulation

Prerequisites:
- Corequisite: CS 446.
- C/C++, Java or Python programming experience in UNIX-like environment.

Texts:
Required textbook:

Reference Books:
Syllabus (Tentative):
This is a tentative list of topics, subject to modification and reorganization.

1. Introduction to Computer Networks
   - Internet
   - Network Concepts
   - Protocols
   - Network Security
2. Application Layer
   - Web: HTTP
   - File Transfer: FTP
   - Electronic Mail: SMTP
   - Domain Name Service: DNS
   - Peer-to-Peer Applications
3. Transport Layer
   - Multiplexing and Demultiplexing
   - Connectionless Transport: UDP
   - Reliable Data Transfer
   - Connection-Oriented Transport: TCP
   - Flow/Congestion Control
4. Network Layer
   - Virtual Circuit and Datagram Networks
   - Routers
   - Internet Protocol: IP
   - Routing Algorithms
   - Routing Protocols: RIP, OSPF, BGP
   - Broadcast and Multicast
5. Link Layer
   - Error-Detection and -Correction Techniques: CRC
   - Multiple Access: MAC, ARP
   - Ethernet
   - Link-Layer Switches: VLAN
   - Link Virtualization: MPLS
6. Physical Layer
   - Data and Signals
   - Digital and Analog Data
   - Encoding
   - Bandwidth Utilization

Organization:
- Except this web page, all course materials will be posted at the WebCampus.
- The organization of the course may evolve as the semester progresses. I'm quite confident that it will be challenging but a fun course.
- Presentation slides are available on the class web page. However, they might be modified before/after the lectures.
- Unless instructed otherwise, use of electronic devices including laptops are not allowed during lectures.
- Class participation in terms of asking questions is highly encouraged. Please do not be afraid to ask
questions no matter how simple you might think the answer could be. This type of interaction helps improve the effectiveness of the class and breaks the monotony.

- Students are encouraged to bring articles, demos, web pages, news events, etc. that are relevant to course topics to the attention of the instructor. The success of the course depends on everyone in the class engaging the material and bringing energy, enthusiasm, and intellect to class activities.

- Regular attendance is highly recommended. If you miss a class, you are responsible for all material covered or assigned in class. You should arrive on time and be prepared to discuss the session’s topic. The underlying notion of the class is interaction, not passivity.

- There will be reading assignments. There will be six in-class quizzes. The lowest graded one will not affect your overall grade. These quizzes will be closed book/notes and extremely time-constrained, i.e., 5-10 mins. Questions in these quizzes will be designed to give you an opportunity to test and affirm your knowledge of the course content.

- There will be lab assignments. Providing hands on experience, the assignments will provide in depth analysis of some protocols. You will use Wireshark packet sniffing tool to analyze communication protocols. Late homework assignments will be penalized by 20% per day.

- There will be a programming project. It will require turning in code that compiles and runs properly and a report documenting the code. Late submissions will be penalized by 10% per day. Note that knowledge of C/C++, Java or Python is required for the programming project. In the project, there will be bonus components.

- There will be one midterm exam and one final exam. You should plan on taking the exams on the scheduled times. No late/early exams unless in case of an emergency situation such as health emergency or similar un-avoid-able situations and you need to provide convincing documentation for it. The exams will be closed books and closed notes but a single page cheat sheet (one side, letter) is allowed. No calculators (unless otherwise stated) and no other electronic devices such as laptops, cell phones, beepers, etc. should be used during the exam.

- From time to time, we may discuss vulnerabilities in networking systems. This is not intended as an invitation to go exploit those vulnerabilities! It is important that we be able to discuss real-world experience candidly; everyone is expected to behave responsibly.
• Assignments and exams must be prepared strictly individually. You are welcome to discuss the problems or solution strategies with your classmates but the resulting work should be your own. Copying from each other or from other sources is considered as cheating. Any form of cheating such as plagiarism or ghostwriting will incur a severe penalty, usually failure in the course. Please refer to the UNR policy on Academic Standards.

• Surreptitious or covert video-taping of class or unauthorized audio recording of class is prohibited by law and by Board of Regents policy. This class may be videotaped or audio recorded only with the written permission of the instructor. In order to accommodate students with disabilities, some students may be given permission to record class lectures and discussions. Therefore, students should understand that their comments during class may be recorded.

• If you have a disability for which you will need to request accommodations, please contact the instructor or someone at the Disability Resource Center (Thompson Student Services - 101) as soon as possible.

• Academic Success Services: Your student fees cover usage of the Math Center (784-4433 or www.unr.edu/mathcenter), Tutoring Center (784-6801 or www.unr.edu/tutoring), and University Writing Center (784-6030 or www.unr.edu/writing_center). These centers support your classroom learning; it is your responsibility to take advantage of their services. Keep in mind that seeking help outside of class is the sign of a responsible and successful student.

Grading (Tentative):
Both grading policy and scale are subject to change. Failure in either the assignments or the tests will result in failure in the course.

Programming Project: 20%
Homework assignments: 20%
In-class Quiz: 10%
Mid-term exam: 25%
Final exam: 25%

Grading Scale (Tentative)

A: 87 and above
B: 75 and above but below 87
C: 63 and above but below 75
D: 50 and above but below 63
F: 0 - 50 (or caught cheating)

Important Note: You will have one week to appeal for your grades after the graded assignments/tests are returned. So, please keep this in mind if you think that there is a problem/issue with the grading of your work.
ABET Criteria:

The course outcomes are skills and abilities students should have acquired by the end of the course. These outcomes are defined in terms of the ABET Accreditation Criterion 3 Program Outcomes which are relevant to this course.

Our graduates will have achieved:

- (2) an ability to design and conduct experiments, as well as to analyze and interpret data.
- (3) an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs, within realistic constraints specific to the field.
- (5) an ability to analyze a problem, and identify, formulate and use the appropriate computing and engineering requirements for obtaining its solution.
- (9) a recognition of the need for, and an ability to engage in continuing professional development and life-long learning.
- (11) an ability to use current techniques, skills, and tools necessary for computing and engineering practice.
- (12) an ability to apply mathematical foundations, algorithmic principles, and computer science and engineering theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.