**Required knowledge.**

* Networking
  + CompTIA Network+, CCNA, or an introductory course in networking should suffice.
  + Although we will cover the basic concepts, prior foundational networking knowledge is assumed.
* Programming
  + Proficiency in the C/C++ programming languages is preferred.
    - If you have experience in a C like language you should be able to get through the course, but you might have to spend some time learning the basic syntax, pointers, structures, and array. We might use Python in the later parts of the course to cover data serialization.
  + Compiling and linking using gcc or g++ as well as basic use of make.
  + Why C and not Python or another higher-level language? Python implements the socket API in a similar fashion to C. The skills and knowledge learned by not only will they transfer directly to Python (and other languages), but you will also gain a deeper understand of how the socket API is implemented in those higher-level languages.
* Basic Unix/Linux command line.

**Topics Covered**

* Networking review.
* Introduction to sockets.
* Socket API functions.
* TCP client and server socket API programming.
* UDP client and server socket API programming.
* HTTP client and server socket API programming.
* HTTPS client and server socket API programming.
  + TLS and OpenSSL.
* Briefly cover socket options, error handling, and data serialization.
* Module 1 - Networking Review
* Module 2 - Intro to Sockets
* Module 3 - TCP Sockets
* Module 4 - UDP Sockets
* Module 5 - DNS and Name Resolution
* Module 6 - HTTP.md
* Module 7 - HTTPS and Encryption
* Module 8 - Data Serialization and Additional Topics

Notes:

* Modules 1- 3 will take the roughly 6 weeks to get through given the amount of material and research required.

**Course Folder Structure**

* Root directory: CSC791- IS Network Programming.
* Modules: Contains the notes and source code for each of the covered topics.
* Syllabus: Outline of the course.
* Lab Source Code: @TODO
* Textbook Source Code.zip: Contains the source code and GitHub repo for the textbooks used throughout the course.

**Course GitHub Repository:**

<https://github.com/eventura1/network_programming>

**Course Materials**

**Textbooks required** (Free access through O’Reilly learning through DSU).

Lewis Van Winkle, “Hands-On Network Programming with C". Packt Publishing. May 2019. ISBN: 9781789349863. <https://learning.oreilly.com/library/view/hands-on-network-programming/9781789349863/>

W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, “The Sockets Networking API: UNIX® Network Programming Volume 1, Third Edition”. Addison Wesley. November 2003. ISBN: 0-13-141155-1. <https://learning.oreilly.com/library/view/the-sockets-networking/0131411551/>

Brian “Beej Jorgensen” Hall, “Beej's Guide to Network Programming, v3.1.11”. April 2023. <https://beej.us/guide/bgnet/html/split/>

**Optional**

Jon Erickson, “Hacking the Art of Exploitation 2nd ed”. No Starch Press. February 2008. ISBN: 978-1593271442. <https://learning.oreilly.com/library/view/hacking-the-art/9781593271442/>

Brandon Rhodes, John Goerzen, “Foundations of Python Network Programming, Third Edition”. Apress. August 2014. <https://learning.oreilly.com/library/view/foundations-of-python/9781430258551/>

\*Others will be provided in the relevant modules as needed. You are encouraged to Google as needed as there might be better or newer resources available.

**Additional Information**

Our focus will be primarily on the Linux Operating System. While the intent was to cover both Linux and Windows, a few weeks into the course it was determined that supporting cross-platform applications added a lot of complexity and distracted from the main goal which is to learn socket programming.

All the examples in the book “Hands-On Network Programming with C” are written for both Windows and Linux, for simplicity I’ve modified the code to only support Linux. You can work on the Windows portion of the code if you’d like to.