Introduction to Computer Science
Summer 2018

Course Syllabus

Overview:
Computer Science is a broad field that studies methodological problem solving with computational devices. This course will provide a taste of the various areas in CS, primarily those that will show students how to think computationally (or algorithmically) such that they can use computers to solve problems efficiently. The successful student of this class will understand CS superpowers: abstraction, concurrency, scale and some of the minor powers: algorithms, data structures, encapsulation and programming.

Students in this class will be introduced to a wide variety of computer languages: Scratch, C, Python, SystemVerilog, Javascript, CSS and HTML. However, it is not the intent of the class to teach programming.

Instructor:
Bill Nace

Teaching Assistant:
TBD

Time and Location:
TBD

Textbook:
No textbook is required.

Other Reading:
The C Book by Banahan, Brady and Doran. A decent reference document, I like it because it is freely available online. Find it at
http://publications.gbdirect.co.uk/c_book/

Essential C by Nick Parlante. Another good reference, also available for free. It is maintained by Stanford University, and is perhaps better as a teaching book. A PDF, this one can be saved for access offline. Find it at http://cslibrary.stanford.edu/101/EssentialC.pdf

Course Website:
All course materials (this syllabus, lecture slides, reading materials, instructions for assignments, etc) will be made available on the course website or distributed via WeChat

Grade assignment: The course grade will be calculated from a weighted average of the individual event scores. The events and weights are as follows:

Weight	Event
60%	Problem sets.
30%	Comprehensive final exam.
10%	Class participation, based on the questions and comments you contribute in class. Just showing up for class and not saying anything is not considered to be participation. Be ready by reading any assigned material ahead of time.

Late Policy: All assignments must be submitted by the deadline. No late work will be accepted. If you have a life-changing event, talk to the professor (ahead of time if at all possible) for a potential extension.

Grading Appeals: If you believe there has been a mistake in the grading of any assignment or exam, then please bring it to our notice so we might correct it. Grade appeals must be submitted **in writing** within 1 week to the instructor**.**

Academic Integrity: *Introduction to Computer Science will adhere to the strictest standard of academic honesty.* All work presented for a grade must be your own -- you are never permitted to copy someone else's work to present as your own. You must also identify the conceptual sources for all work submitted (i.e. if you discuss the work with anyone other than an instructor or TA, you must identify that person by name in your submission). All parties involved in an infraction are subject to disciplinary actions to the fullest extent permitted, generally failure of the course. I will also punish any infraction that I believe occurred (Standard of evidence is "Preponderance of the Evidence"). Such punishment will generally be assignment of a zero grade for the event, as well as retroactive examination of previous work for signs of cheating.

Working and learning in a group is a fine thing to do. Teaching others in a group is often the easiest way to make sure you learn the material yourself. However, ensure that each person in your group is doing their own work and not simply copying down a group-generated answer. Please, please, do the right thing.