

# CSCE 120 – Learning to Code

---

*Fall 2017*

“If you really want to understand something, the best way is to try and explain it to someone else. That forces you to sort it out in your own mind... that’s really the essence of programming. By the time you've sorted out a complicated idea into little steps that even a stupid machine can deal with, you've certainly learned something about it yourself.”

–Douglas Adams, *Dirk Gently’s Holistic Detective Agency*

“The world of A.D. 2014 will have few routine jobs that cannot be done better by some machine than by any human being. Mankind will therefore have become largely a race of machine tenders. Schools will have to be oriented in this direction.... All the high-school students will be taught the fundamentals of computer technology, will become proficient in binary arithmetic and will be trained to perfection in the use of the computer languages that will have developed out of those like the contemporary ‘Fortran’”

–Isaac Asimov 1964

## Course Info

Prerequisites	A genuine curiosity about coding and a perseverance to maintain that curiosity in the face of challenges; Placement in MATH 101 or higher
Instructor	Dr. Chris Bourke <a href="mailto:cbourke@cse.unl.edu">cbourke@cse.unl.edu</a> Avery Hall 363
Office Hours	See Course Web Page

## Bulletin Description

Introduction to coding in the context of current web development technologies (JavaScript, HTML, CSS). Basic coding skills and an introduction to computing with an emphasis on processing data: data formatting and structure, data manipulation, data presentation and the basics of an interactive program.

## Course Description

Computing and automation has become ubiquitous and will only continue to grow and permeate all aspects of our daily lives. The time is soon coming where computational thinking and the ability to develop software will be considered a basic skill necessary to every discipline and a requirement for many jobs. The Department of Computer Science & Engineering's new Software Development minor seeks to equip students from diverse backgrounds and disciplines with competency in the key principles and practices of software development. The Software Development minor is open to non-CSE students and consists of a sequence of 5 courses.

The first course in the sequence, CSCE 120 - Learning to Code provides students with basic coding skills and an introduction to computing with an emphasis on processing data: data formatting and structure, data manipulation, data presentation and the basics of an interactive program.

## Skills Outcomes

A student who has successfully completed this course will have the following skills.

- Be able to identify, interpret, and contextualize various electronic data formats
- Be able to code a program solving simple problems or processing data
- Have mastery over basic programming language structures (variables, conditionals, loops, I/O) in at least one programming language (JavaScript)
- Be able to read, understand, and utilize libraries and APIs
- Be able to design a simple HTML/CSS/JS application that processes and visualizes data from a variety of sources/formats and provides rudimentary user interaction
- Have practice developing code in pairs and teams, resolving conflicts and effectively sharing common resources
- Be able to present and defend code and designs; be able to analyze and critique code created by others and receive critiques of code and incorporate appropriate changes

## Class Format

This course is offered as a “flipped classroom.” Traditional classrooms use class time to lecture and with exercises being performed outside of class as “homework.” A flipped classroom reverses this. Lectures are prerecorded and other material is available online. Prior to each week’s class meetings, you will be expected to do the following:

1. Watch a series of short video(s) containing lecture material.
2. Read a handout giving further details on the module’s topics.
3. Engage in at least one other activity of your choice such as watching or reading an online tutorial or doing an interactive programming exercise.

During each class, you will be assigned into pairs and engage in programming and other exercises. In a flipped model, your instructor is less of a lecturer and more of a *facilitator*. Following each module, there will be supplemental readings and/or exercises available to gain a deeper understanding of the topics.

## Accommodations for Students with Disabilities

It is the policy of the University of Nebraska-Lincoln to provide flexible and individualized accommodations to students with documented disabilities that may affect their ability to fully participate in course activities or to meet course requirements. To receive accommodation services, students must be registered with the Services for Students with Disabilities (SSD) office, 132 Canfield Administration, 472-3787 voice or TTY.

## Grading

Grading will be based on the completing hacktivities, assignments, and a group project with the following **weighted** contributions.

Hacktivities	50%
Assignments	30%
Group Project	20%

## Scale

Letter grades will be awarded based on the following scale. This scale may be adjusted upwards if the instructor deems it necessary based on the final grades only. No scale will be made for individual assignments.

A+	>= 97	B+	>= 87	C+	>= 77	D+	>= 67	F	<60
A	>= 93	B	>=83	C	>= 73	D	>= 63		
A-	>= 90	B-	>= 80	C-	>= 70	D-	>= 60		

## Hacktivities

In each class meeting you will engage in a number of “hacktivities” and programming exercises. This will be done in a *peer-programming* model. At the start of class, you will be randomly assigned a partner by your instructor to work with for the duration of that class (each class meeting the partners will change). One of you will be the *driver* and the other the *navigator*. The driver is in charge of keyboard/mouse and will be responsible for typing (you may use the lab computer or your laptop, but only one computer per pair). The navigator is in charge of the handout and directing the activity. However, *both* of you are responsible for contributing and discussing solutions.

This is to be a true collaborative environment: neither partner should dominate, nor should either partner be passive. You are both expected to contribute equally. You should both actively voice your opinions in a civil manner and make alternative suggestions.

Because this is a flipped classroom, it is **absolutely essential** that you complete the pre-class activities before coming to class, as we will not be spending class time to cover concepts. Failure to engage in the activities means that you will be unprepared to engage in classroom activities. This is especially unfair to your partner. Therefore, each in-class assignment will be graded not only on completion, but how well each of you contributed and whether or not you properly prepared for class.

Hacktivities must be completed in class and failure to complete a hacktivity on time may result in no grade being awarded. There are (tentatively) about 27 hacktivities, thus each is worth about 1.85% of your final grade.

## Assignments

There will be several programming assignments. Assignments are due at the beginning of class. Code and other relevant files must be submitted using CSE’s web handin (<https://cse-apps.unl.edu/handin>).

## Group Project

In addition to regular assignments, there will be a substantial semester group project. More details are provided in a separate document.

## Late Work Policy

Assignments will have a strict *in-class* (at the beginning) due date. No late work will be accepted.

## Dead Week Policy

In conformance with UNL's 15<sup>th</sup> Week Policy (see Registration and Records main webpage, <http://www.unl.edu/regrec/>), be aware that the final homework and project may be due during the final week of classes.

## Academic Integrity

All homework assignments, programs, quizzes, and exams must be your own work unless otherwise stated. No collaboration with fellow students, past or current, is allowed unless otherwise permitted on specific assignments or problems. The Computer Science & Engineering department has an Academic Integrity Policy. All students enrolled in any computer science course are bound by this policy. You are expected to read, understand, and follow this policy. Violations will be dealt with on a case by case basis and may result in a failing assignment or a failing grade for the course itself. The most recent version of the Academic Integrity Policy can be found at

[http://www.cse.unl.edu/undergrads/academic\\_integrity.php](http://www.cse.unl.edu/undergrads/academic_integrity.php)

## Communication

The best way to communicate with your instructor is through email. The instructor and teaching assistants will communicate with you either directly or through the Blackboard email system. You are responsible for ensuring that the email associated with your Blackboard account is up-to-date and that you are regularly checking it.

There is an anonymous suggestion box (available via Blackboard) that you may use to voice your concerns about any problems in the course if you do not wish to be identified.

In addition, the Department of Computer Science & Engineering also maintains an anonymous suggestion box that you may use to voice your concerns about any problems in the course or department if you do not wish to be identified. It is available at the following URL:

<http://cse.unl.edu/department/suggestion.php>

Finally, I will hold regular office hours (to be announced) and will make myself available by appointment; please email me to set one up.

## Help

Your success in this course is ultimately your responsibility. That said, there several outlets for you to seek help and assistance.

1. Your Instructor – Attend lecture regularly and engage in class discussions, ask questions in class, visit me during my office hours or setup a meeting time to see me, email me!
2. Your TAs – TAs hold regular weekly office hours, visit with them and ask questions, ask for examples, etc.

3. Student Resource Center – Though they may not be your direct TA, all Graduate Teaching Assistants (and some advanced undergraduates) hold regular office hours in the Student Resource Center (<http://cse.unl.edu/src/>) open Monday thru Friday 9AM to 7PM and should be staffed most hours. Ask for help from anyone in the SRC.
4. Course materials – start on assignments early, attend lectures and labs, work extra problems from the book, read all required (and optional!) materials.
5. Your colleagues – Chances are, if you are having problems, your classmates are having them too. Discussion and dialog among students is encouraged (within the parameters set by CSE's academic integrity policy, this course's policy, and policies set for individual assignments).