CSCE 155N - Comp Sci 1: Engineering & Science

Course Information

Course Number :	CSCE 155N
Course Title:	Computer Science 1: Engineering and Science Focus
Credit:	3
Days:	Monday, Wednesday & Friday
Time:	2:30-3:00 pm
Location:	BESY 117
Textbooks:	MATLAB: A Practical Introduction to Programming and Problem Solving 4th edition (ISBN-13: 978-0124058767)

Instructor Information

Name:	Reginald Bolman	
Email:	rbolman2@unl.edu	
Professor for online section:	Charles Riedesel	
Email:	chuckr@unl.edu	
Office Hours:	To be determined	
Teaching Assistants:	Mutabazi	mutabazi@huskers.unl.edu
	Cai	jcai@cse.unl.edu
	Lafferty	rclafferty@huskers.unl.edu

Required Software

Basic computer with a Matlab installation. Matlab can be acquired at https://www.mathworks.com/ where the student license is acquired at https://itprocurement.unl.edu/matlab-license-details. . A working version of Windows/Linux (Mint/Ubuntu are recommended). Students may use a MAC computer at their own risk as there may be compatibility issues with MATLAB for certain assignments. Note that the lab computers run Matlab on Windows computers.

Tentative Course Schedule

Week	Date	Subject
1	1/13	Syllabus
	1/15	What is Computer Science?
	1/17	History of Computer Science
2	1/20	MLK Day

CSCE 155N - Comp Sci 1: Engineering & Science

	1/22	Computational Thinking
	1/24	Pseudocode
3	1/27	Intro to MATLAB 1.1-1.4
	1/29	Intro to MATLAB 1.8
	1/31	Practice Problems from Ch 1
4	2/3	Vectors and Matrices 2.1 – 2.2
	2/5	Vectors and Matrices 2.3 – 2.4
	2/7	Chapter 2 live coding
5	2/10	Vectors and Matrices 2.5
	2/12	~~Intro to MATLAB Prog. 3.1- 3.4~~ Live coding Chapter 2.5
	2/14	~~Intro to MATLAB Prog. 3.4- 3.7~~ Chapter 2 problem sets
6	2/17	~~Intro to MATLAB Prog. 3.8 and Practice Problems from Ch 3~~ Matrices
	2/19	~~Selection Statements 4.1 – 4.3~~ Matrices live coding
	2/21	~~Selection Statements 4.3 – 4.5~~ Matrix problem sets review
7	2/24	~~Practice Problems from Ch 4~~ Chapter 3.1-3.4
	2/26	~~Loop Statements 5.1 – 5.2~~ Chapter 3 live coding
	2/28	~~Loop Statements 5.3~~ Chapter 3 orange/blue problem sets review
8	3/2	~~Loop Statements 5.4~~ Chapter 3.4-3.6 material
	3/4	~~Loop Statements 5.5 and Practice Problems from Ch 5~~ Chapter 3.4-3.6 live coding
	3/6	~~Review for Midterm~~ Chapter 3 problem sets
9	3/9	**Midterm Review**
	3/11	No class (Midterm open-study)
	3/13	Midterm
10	3/16	~~MATLAB Programs 6.3 – 6.4~~ Changed to Chapter 4 material → University Cancelled
	3/18	~~Debugging 6.5~~ University Cancelled
	3/20	~~Practice Problems from Ch 6~~ University cancelled

CSCE 155N - Comp Sci 1: Engineering & Science

11	3/23 – 3/27	Spring Break
12	3/30	Selection Statements 4.1 – 4.3
	4/1	Selection Statements 4.3 – 4.5
	4/3	Practice Problems from Ch 4
13	4/6	Loop Statements 5.1 – 5.2
	4/8	Loop Statements 5.3
	4/10	Loop Statements 5.4
12	4/13	Loop Statements 5.5 and Practice Problems from Ch 5
	4/15	MATLAB Programs 6.1
	4/17	MATLAB Programs 6.2
14	4/20	MATLAB Programs 6.3 – 6.4
	4/22	Debugging 6.5
	4/24	Practice Problems from Ch 6
15	4/27	Final Review
	4/29	No class – Dead week, study for finals
	5/1	No class- Dead week, study for finals
Finals Week		**You will have from May 4th to May 7th (midnight) to finish the final**

Course setup (In-Class sections)

The course grade is composed of 5 main sections:

- Labs 40%
- Diary 20%
- Midterm Exam 20%
- Final Exam 20%

Labs

The lab is designed to be a guided learning environment where students will be exposed to more complicated problems and with the guidance of the TA's/GA's be able to work through those problem sets. Students will complete a lab report after each lab and submit them and/or any accompanying files in order to receive credit from the TA/GA for a specific lab. Treat the lab reports as you would an English paper or history essay, we're grading for both **content and quality** within this course.

CSCE 155N - Comp Sci 1: Engineering & Science

Diary

The Diary component is similar to HW in other classes, students will work through the blue/orange exercise within their textbooks and complete the problem sets each week. Additionally, for each day of class students will record the programming of the professor completed in class and save this as its own **.txt** file. Students must turn in the blue/orange/problem HW sets in addition to the lecture notes in order to receive full credit. This is designed to actively reinforce individual learning within a structured environment.

Exams

The final and midterm are comprehensive examinations which will test your learning comprehension and retention of the course content.

Course setup (Covid19 Updates)

Office hours will no longer be held in person in conformance with the University "social distancing" policies. For your own safety as well as that of your instructor and/or the safety of those who work within the Schorr/Avery buildings, it's to the best interest of everyone to have as few students running around the university as possible (especially those without N95 and P100 respirators). In order to compensate for this change, we will hold online office hours which will be listed below.

Professor's office hours (CSCE 101 and CSCE155N) : Friday 9:00-11:00am
https://unl.zoom.us/j/581935981

Lab times and lead URLs:
8 AM - Tyler Hahn - https://unl.zoom.us/j/918744495
- Backup: Trevor Thomazin
3:30 PM - Junzhe Cai - https://unl.zoom.us/j/2468194410
- Backup: Tyler Hahn
5 PM Tuesday - Junzhe Cai - https://unl.zoom.us/j/2468194410
- Backup: Tyler Hahn
5 PM Thursday (both rooms in one Zoom meeting) - Junzhe Cai - https://unl.zoom.us/j/2468194410
- Backup: Tyler Hahn

Office hours times and URLs:
Junzhe Cai: 3:30-5:30 - https://unl.zoom.us/j/2468194410
Casey Lafferty: 2:30 - 4:30 Thursdays - https://unl.zoom.us/j/2012366569
Bill Mutabazi: 4 -6:00 Thursdays – https://unl.zoom.us/j/4971304135
Gui Eymael: 12:30 - 2:30 M/W/F - https://unl.zoom.us/j/507953388
Emme Campbell: 6:30 - 7:30 T/H - https://unl.zoom.us/j/9364382973
Thomas Vierk: https://unl.zoom.us/j/9906061020
Maia Ramsay: https://zoom.us/j/3724507520

CSCE 155N - Comp Sci 1: Engineering & Science

Course setup (Online sections)

Online students will not be allowed to cooperate in project groups and/or "lab groups" with the in-class section. This policy is due to the fact that this confuses the grading and makes grading projects and/or applying bonus points to a given project an added challenge. **All labs for online sections will be released through CANVAS through the main class directory with all other assignments. I will not be using the CANVAS lab tab for online labs since this is superfluous.**

Diary:

Online sections will be graded slightly differently due to the fact that there isn't an "in-class" diary section. Instead of having a 50% in-class lecture notes txt files. 100% of your grade will be the orange/blue chapter sections and problem sets.

Class Setup

The lecture is formatted into two components: first, we will discuss some sort of a problem situation which arises in the engineering world and/or a problem within computer science. Second, I will then show you how to overcome this challenge with in-class examples. Programming is best learned by doing, much like any other trade. The purpose of the lecture is to show HOW to program and/or how to solve a given problem using programming. I will oftentimes prefer using the chalkboard to a power point presentation as a power point does not motivate student learning but student snoring! If you have trouble seeing the board, feel free to move forward and/or ask that something be rewritten – I have no problem rewriting anything during the lecture. I will often ask the class if they've understood a concept, if a concept has not been sufficiently explained, feel free to ask a question even if other students choose to remain silent. If you, as a student, didn't understand a concept, then it is highly likely that there exist other students in the class who likewise didn't understand that particular concept. Additionally, many students would ask questions after class has ended. I don't have any problem staying after class talking with students for as long as they need (pending that I don't have other work which must be attended to).

Course Policies and Objectives

Academic Integrity Policy (Department of Computer Science & Engineering)

While I encourage discussion, I abhor academic dishonesty. Discussion should be done at a level above pseudocode. Discussions of what and why are perfectly acceptable, but discussions of how something is implemented are prohibited. According to the Department's policy: Unless specifically prohibited by the instructor, it is acceptable to discuss the meaning of assignments. Discussing general approaches and strategies for solutions may be permissible, but unless specifically allowed, such communications should not include written material or code and should not transmit substantive or specific elements of a solution. Discussions about assignments (and external references consulted) must be cited in assignment reports. **The use of solutions provided by other students (both current and former)**, "solution manuals", and other external sources is prohibited. Project group code which is too similar to other groups will be assumed to be an example of plagiarism. Project group code which is too similar to an online solution will be assumed to be an example of plagiarism. There are many different ways one can come to obtain a given solution to their project work and some programming solutions are better than others (i.e. more computationally efficient, more structured thinking/understanding of the problem, more clever, etc.). If student project groups are truly working independently, their programming code should theoretically look wildly different. Students are responsible ultimately for

making sure that other groups do not obtain their solutions and/or being able to explain their solution to the professor when there is an incident of plagiarism that should arise**.**

First-time offenders will receive a 0 on the entire assignment in question and be given an Incomplete in the class until they successfully complete an online ethics course administered by the Office of the Dean of Students.

Repeat offenders will receive an F in the class and will be recommended to the Academic Integrity Committee for dismissal from the Department and all of its programs.

Late Work

Late work will be handled on an individual by individual basis and is entirely up to the GA/TA. If a student requires an extension on an individual assignment, they must make an attempt to then turn in that assignment and keep up with their individual student grades. **Students are responsible for their individual grades!** If a grader doesn't grade your paper and/or if a grader gives a student a 0, it's up to the student to figure out why that 0 exists and/or why a paper wasn't graded. We have 100+ students! It's very easy for an incomplete to turn into a 0 by the end of the year when students are turning in late work!

After an assignment has been handed back to the student, no consideration for regrading will be given to assignments which are over 1 weeks old. Students are ultimately responsible understanding how to communicate with graders if/when a student isn't satisfied with the grade they've been given. Students will be given a 2 day grace period when turning in assignments for a penalty of 30% off that particular assignment, all additional extensions will be on a person by person basis and are up to the TA's/GA's.

Exams

Collaboration is not allowed on exams. Exams will be administered through the Testing Center in Love Library North. There will be a window of days in which you may take an exam

Assignments

The course is composed primarily of labs which are designed to teach the basic course content, ability to perform outside research and problem solving skills, midterm/exam work which tests retention and overall knowledge of the course content and diaries which allow a student to receive in class grades for class assignment work.

ACE Learning Objectives

This course is approved for ACE Student Learning Objective 3, according to the following criteria: SLO3: Use mathematical, computational, statistical, or formal reasoning (including reasoning based on principles of logic) to solve problems, draw inferences, and determine reasonableness.

1. Describe opportunities students should have to learn the outcome. How is the learning objective embedded in the course? The course presents many opportunities to learn computational and formal reasoning methodologies and skills to solve problems, draw inferences, and determine reasonableness. Specifically, the lectures, together with the programming assignments and the weekly laboratory sessions, teach students both algorithms and the implementation of those algorithms to solve problems. That is, the course not only teaches students about how to design algorithmic solutions to solve

problems, but how to engineer designs into working programs. This engineering process involves significant debugging, testing, and refining code. These activities teach and reinforce inferencing: a student must draw inferences when diagnosing why a program crashes, does not compile, or generates incorrect output; after making fixes, a student must re-evaluate the design to see if the outcome meets expectations, and further draw inferences on how to proceed. Finally, an algorithm is fundamentally a logical sequence of steps that, given a set of input, generates output. Specifications for the output determine reasonableness. Through algorithmic development, with top-down design, problem analysis and specification, step-wise refinement, and modularization, the students, when programming, are trained to determine the reasonableness of their solution. For example, students are trained to examine how their algorithms handle exceptions (which could terminate an algorithm prematurely if handled incorrectly), deal with boundary conditions (to prevent their programs from crashing), and prevent infinite loops (which could prevent reaching an outcome). The course has approximately 45 hours of lectures each designed to explore concepts and paradigms that are central to the field of computer science. Students will master control, repetition, selection, input and output processes, and procedure and function design and invocation. Students will learn fundamental problem solving paradigms, including abstraction, encapsulation, exception handling, and event-driven programming. Through lectures, laboratory sessions, and programming assignments, students learn about problem analysis and specification, top-down design, algorithm development (including recursion), step-wise refinement, and modularization. Students will also be exposed to various algorithms, such as for searching and sorting. The course has approximately 14 hours of laboratory sessions, each designed to train students to apply what they learn in the lectures to actual implementation and analysis of algorithms and software programs. Laboratory sessions require students to solve problems, to debug or revise programs, to analyze programs. These activities reinforce the students on problem solving, drawing inferences from their design and implementation, and determining reasonableness of a solution. The course includes several programming assignments designed to help students learn about designing algorithmic solutions and the practice of implementing solutions as correct software programs, involving key steps such as program analysis, solution identification and evaluation, solution-to-algorithm mapping, initial feasibility analysis, coding, debugging, testing, and refinement. These steps provide ample opportunities for students to apply their computational and formal reasoning skills to solve problems, draw inferences, and determine reasonableness.

2. Describe student work that will be used to assess student achievement of the outcome and explain how the students demonstrate the knowledge and skills specified by the outcome. The assessment of achievement of the outcome consists of three primary tools: exams, programming assignments, and laboratory assignments. The programming assignments inherently embed the results of problem solving, inferencing, and reasonableness reasoning, because in order to produce a working program that compiles, runs, and computes the correct output, a student must devise and implement an algorithmic solution. For the laboratory assignments, worksheets where students submit their findings from solving the lab problems are then graded whereby students are then tested with pre and post tests in order to measure learning retention. Within this class there exists a midterm exam and a comprehensive final exam. Most of the questions are designed to measure comprehension, application, and analysis (according to Blooms level), and a few will be on evaluation and synthesis. These will assess student understanding of the fundamental CS concepts and paradigms covered in the course. Students will be asked to design algorithms to solve problems, to analyze algorithms to infer what they might or might not do, to evaluate algorithms to determine how well they meet their design goals. These questions allow us to assess the student's ability to solve problems, draw inferences, and determine reasonableness. There are 4 programming assignments. These will assess student's ability to apply CS concepts and paradigms to hands-on problem solving and actual software implementation. There are

CSCE 155N - Comp Sci 1: Engineering & Science

˜12 structured laboratory assignments. These will assess student's knowledge on programming, a specific programming language (syntax), debugging, and algorithm development.

3. As part of the ACE certification process, the department/unit agrees to collect and assess a reasonable sample of students' work and provide reflections on students' achievement of the Learning Outcomes for its respective ACE certified courses. Please comment on your plans to develop a process to collect and evaluate student work over time for the purpose of assessing student success for this ACE outcome. In accordance with the ABET accreditation process, the CSE Department systematically collects A, B, and C samples of every assignment (exam, laboratory, homework, quizzes, etc.). Reviews of these student work samples are made by the CSE Departments Assessment Committee when the CSE Department reports to the ABET accreditation board. A standardized online post-test is required for every student in CSCE155. Results downloaded, archived to a database, analyzed by the Assessment Committee, and reported to the CSE Department Chairperson and Curriculum Committee. Assessment reports are considered by the CSE Department Chairperson and Curriculum Committee in developing instructional and curricular strategies and implementation and in the ABET Accreditation Self-Study.

Reinforcements

According to the ACE document approved by faculty (Structural Criteria, item 9)," Every ACE course will reinforce at least one of the following skills listed below as appropriate for the discipline and as identified by the department offering the course..." Indicate skills that will be reinforced by the course by clicking on as many as apply and describe briefly how those skills will be reinforced. These areas are those OTHER THAN the one or two outcomes for which you seek ACE certification. Students will not receive ACE credit for the reinforced skills, and the reinforced skills do not need to be assessed for ACE purposes.

Reinforced Outcomes/Skills

Critical thinking is key in the development of algorithms and during the debugging process of implementing a program. The course provides numerous opportunities for critical thinking in lectures, programming assignments, and laboratories. The laboratories and assignments are problem-based and students are tasked to apply critical thinking to solve problems.

Problem Solving

The development of algorithms and the implementation of programs are inherently problem solving. The course provides numerous opportunities for problem solving in programming assignment and laboratories. The laboratories and assignments are problem-based and students are tasked to solve problems.

Students with Disabilities

Students with disabilities are encouraged to contact the instructor for a confidential discussion of their individual needs for academic accommodation. It is the policy of the University of Nebraska-Lincoln to provide flexible and individualized accommodation 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.

CSCE 155N - Comp Sci 1: Engineering & Science

Sources for Help and Assistance

UNL offers a variety of options to students to aid them in dealing with stress and adversity. Counseling and Psychological Services (CAPS) (https://caps.unl.edu) is a multidisciplinary team of psychologists and counselors that works collaboratively with Nebraska students to help them explore their feelings and thoughts and learn helpful ways to improve their mental, psychological, and emotional well-being when issues arise. CAPS can be reached by calling 402-472-7450.

Big Red Resilience & Well-Being (https://resilience.unl.edu/big-red-resilience-well-being) provides fun events, innovative education, and dynamic services to help students understand emotions, manage stress, build strength, connect with others, develop grit, and navigate transitions.

CSE Department

The CSE Department has an anonymous contact form (http://cse.unl.edu/contact-form) 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 CSE Department policy that all students in CSE courses are expected to regularly check their email so they do not miss important announcements.

You are ultimately responsible for your success in this course. If you have questions on material covered or assigned in class, it is up to you to seek out assistance from the course instructors or TAs. Staff in the CSE Student Resource Center may also be able to assist you with general questions. The CSE Student Resource Center (http://cse.unl.edu/src) is located at Avery 12. The CSE Department also maintains a Frequently Asked Questions page (https://cse.unl.edu/faq).

Dead Week Policy

In compliance with UNL's 15th Week Policy (see the main Registration and Records webpage at http://www.unl.edu/regrec), be aware that the final project will be due during the final week of classes. Note also that all assignments, homework, labs, etc., will have a strict final due date during the final week of classes.

Course "College Academic Readiness" policy

There are many instances where a student might not have been exposed to one particular concept or another or perhaps that student doesn't remember a particular concept from secondary school. This is fine! This is what college education is all about!

Although this is not a math class, much of programming contains a decent dose of mathematics. Ergo, upon entering this class we are expecting students to understand things such as basic regression, normality, variance, mean, mode. Students are not required to have an understanding of differentiation/integration and most of this course will require math no higher than Pre-Calc. **When we discuss a topic, which contains "higher level" math concepts, these will be explained in detail within the class lecture. Complete mastery of mathematic topics will not be required in order to either take the quiz, complete the homework and complete the projects.**

CSCE 155N - Comp Sci 1: Engineering & Science

If you need help with the course

TA's/GA's exist to help facilitate student learning within this course although – each and every student is responsible for attempting to understand, learn, etc.

Proper Communication with TA's/GA's

Students should address every TA/GA as Mr. or Ms. _____ as a sign of respect when communicating with them. The TA's/GA's although young, are your supervisors and sending a nasty email to a TA/GA when a student receives a grade which they don't like is unacceptable. If there are questions concerning your grade, you should first talk with the TA/GA about the given issue. If the situation is not resolved, we can schedule a meeting between the GA/TA and myself to see if we can come to an agreement regarding a specific issue. The GA/TA's are responsible for regrading and updating grades.

All students must sign, date and submit the following:

I _____ have read and understand the syllabus in its entirety.

I _____ understand that there will be no reconsideration of grades if an assignment has been handed back for over a week.

I _____ understand that should grading questions arise that the grader who graded my work is the individual that I should contact first and foremost.

I _____ understand that this syllabus constitutes a written and verbal agreement that I'm contractually obligated to be responsible for completing any assignments which have been assigned and/or are due during the "dead week" and that this document ultimately constitutes a formal written notification that the **final lab** will be due on the day of the final.

I _____ have read and understood the grading criterion for project work.

I _____ understand that most courses require 2-3hours of homework and/or study time for each hour of in-class lecture. For this class that means that there will most likely be anywhere from 4-12 hours of homework per week (depending on student level/ability) in order to make sure that I'm able to perform well on my examinations.

I _____ understand that due to the class size and the professor's experience teaching many different students, that the professor reserves the right to change assignment **due date(s)** and/or **course content** at any time and for any reason. **I understand that changes to course content is designed with the intention to help students** as each individual class moves through the course. I agree that I am ultimately responsible for keeping updated with any changes made by checking CANVAS announcements and/or listening when announcements are made in class.

I _____ understand and agree to not receiving any credit for programming assignments where I cannot explain how the program works.

CSCE 155N - Comp Sci 1: Engineering & Science

I _____ understand and agree that verbatim copying solutions, code, etc. from previous students, online sources, etc. will result in my paper receiving a grade of a 0.

I _____ understand that I'm ultimately responsible for my academic performance/lack thereof. My code, my labs, my diaries, my projects and my exams within this class section being completed on time and correctly is my responsibility.

I _____ understand that I will receive a maximum of a 90 in the class if I miss more than 6 classes (6 course hours/2 weeks of class time) and agree to the diary lecture notes serving as an attendance counter (i.e. not submitting course notes for the diaries for more than 6 assignments will result in a maximum grade of a 90 for the course overall).

Attach photo of yourself here (so professor can put a face to a name):

Student Name: _____

ID: _____

Date: _____