Endicott College
Beverly, Massachusetts

School of Arts and Sciences
Computer Science Department
Course Syllabus

Course No:	CSC 160.01 / CSC 160L.L1 / CSC 160L.L2
Course Title:	Introduction to Programming
Credits:	4
Class Type:	Lecture/Lab
Semester and Year:	Spring 2022
Meeting Times:	MWF 9–9:50am; Labs T *or* R 2–3:50pm all in JSC312

Faculty: Henry Feild, Ph.D.
- Office Location: JSC 113A
- Telephone/Zoom: 978-998-7793
- E-mail: hfeild@endicott.edu
- Office Hours: MWF 10–10:50am, TR 4–4:25pm, F 12–1:50:pm

Catalog Description

Provides an introduction to computer programming concepts and functions. Introduces problem-solving methods and algorithm development using software programming. Includes procedural and data abstractions, program design, debugging, testing, and documentation. Covers data types, control structures, functions, parameter passing, library functions, and arrays. Weekly programming laboratory exercises. *Must be registered for lecture and lab sections. Satisfies Quantitative Reasoning General Education requirement.*

Learning Outcomes

At the completion of this course, you should be able to:

1. demonstrate an appreciation for the practice and theory of programming languages
2. demonstrate mastery programming in Python
3. apply problem solving skills and program modeling techniques
4. use basic programming concepts, i.e.: program structure and flow control, simple internal data structures, program switches, iteration, selection and functions
5. appreciate the need for documentation, brevity, and memory usage in programming

Teaching/Learning Strategies

Programming is an active sport, and like a sport, you need to practice a lot to get good at it. Practice will give you the muscle memory you need to type programming statements, and will help you understand the ins and outs of all the programming constructs and how they interact in a program. This course is set up to give you time and space to practice in a variety of contexts: small problems at home using just the readings, larger problems in class just after we've talked about the topic, and finally even larger problems in labs where you can apply what you learned from the homework and class.

You are expected to come to each class having read the assigned reading for the day and submitted your answers to the corresponding homework problems. These will prepare you for the class meeting, which will generally consist of a short, interactive lecture summarizing the reading and addressing your questions, followed by an in-class activity. There will also be a lab each week in which you will work with a partner on a larger problem that you are expected to complete by the following week.

Instead of two or three big exams, you and I will assess your learning through four quizzes and a final exam. Use these more frequent assessments to help you refocus your learning and identify the concepts you understand well and those you can improve your understanding of.

Required Texts/Technology

Havill, Jessen. *Discovering Computer Science.* 2nd edition, Routledge, 2021.
Softback ISBN 9780367472498
eBook ISBN 9781003037149

The textbook includes some sections only available online, as well as resources such as data and program files; you can find the webpage here: https://www.discoveringcs.net.

Assignments and class exercises rely on writing and running code. Some programming will take place online, but most will require you to install a text editor and Python interpreter on your computer. If you do not have a computer capable of installing the required software, please let me know. The course Canvas page includes links to videos and written resources for how to obtain and use a text editor, the Python interpreter, and a command line interface.

Evaluation Methods

The course is made up of the following graded components. Each component will be graded pass/fail based on clearly stated specifications, indicated below where possible. The percentage that a given component contributes to your final course grade is also listed in parentheses.

- Homework (20%), done individually or in small groups prior to each class. These involve reading a section or two of the book and completing a few exercises at the end. Most homework will be graded for effort, for which you will receive 5 points for submitting a reasonable attempt at the solutions. I may select problems to grade for accuracy; these will be marked with an asterisks (*) in the homework assignment on Canvas. Such problems will be graded as follows:
    - Full credit (5pts): your solution is correct or nearly correct
    - Half credit (2.5pts): your solution requires much additional work, but you clearly put in a reasonable effort
    - No credit (0pts): your solution is non-existent or you did not appear to make a reasonable effort at solving the problem
- Participation and class activities (10%). I expect that you and your classmates will fully engage with the class. Much of class time will be spent working on problems in pairs;

you will do well in this category if you actively engage with your partner to solve the problem at hand each class and that you respond to questions or pose your own questions once every one or two class periods.

- Labs (20%), will take place every week. You will start a program with a partner during the lab and the completed version is due the following week. At the beginning of each lab, I will grade each pair's lab submission from the prior week. The grading criteria for each lab will be attached to the lab itself. You may use up to four Redo tokens (see below) if you would like an extension to improve a lab grade.
- Quizzes (40%), will be given four times during the semester; Each will be graded out of 50 points and you will have an entire class period to take it. These are generally cumulative due to the nature of programming (each new topic builds on what came before), but the most recent material will be the primary focus. You may bring a two-sided, 8.5"x11" cheat sheet on these. Your lowest quiz grade will be dropped.
    - Full credit (question level): the provided solution is correct or very close to correct
    - Partial credit (question level): the provided solution isn't very close to correct, but demonstrates an understanding of some of the material pertinent to the question
- Final exam (10%), is graded out of 100 and covers material from the entire class. The question-level grading is the same as for quizzes.

Topical Outline and Timeline

A more detailed outline can be found on the course website, including the sections due for each class and associated problem sets.

Week	Topic	Reading
1–2	Problem solving with programming	Ch. 1
2–3	Visualizing abstraction	Ch. 2
	Quiz 1 (Friday, Feb 11)	
4–5	Growth and decay	Ch. 4
5–6	How computers work	Ch. 3
	Quiz 2 (~~*Friday, Mar 4*~~ Mon, Mar 7)	
6–7	Branching	Ch. 5
	Spring break (Mar. 12–20)	
8–9	Text processing	Ch. 6
	Quiz 3 (Friday, Apr 1)	
10–11	Data analysis	Ch. 7
11	Multi-dimensional arrays	Ch. 8
	Quiz 4 (Friday, Apr 22)	
12–13	Recursion	Ch. 9
13–14	Searching and sorting	Ch. 10
14	Review	

Final exam—May 13, 2022 8–10am in JSC 312

Due Dates, Late Policy, and Redo tokens

Homework

Readings and problem sets are assigned for each class and should be completed before the class meets to be considered on time in order for class to make sense.

I recognize that sometimes life gets in the way or the material becomes overwhelming and you may find yourself behind. To encourage you to catch back up if that happens, I will accept assignments from the homework category late up until the last day of class for 50% credit. You must email me a list of late assignments you would like half credit for.

Class activities

You must attend class in order to receive credit for class activities.

Labs

Each lab is due a week after it is assigned. Part of your grade is that you attended the lab

Redo tokens

Because I would like to encourage you to correct your mistakes, if you receive a lower grade on the technical aspects of a lab than you're happy with, you can use a **redo token** to improve your grade. You will start the semester with **four** of these. Each redo token will extend the deadline for a lab by one week, but you must get feedback from me on your progress prior to invoking a token. You may use multiple tokens for the same lab (e.g., you could submit up to 5 versions of one lab if you wanted to—the original plus four redos), or you may spread them out and use one for each of four labs. It's up to you how you use them. If you are working with a partner, each resubmission costs *each* of you one token.

Quizzes/Final exam

Quizzes and the final exam must be taken during the scheduled time. If you have a conflict (e.g., a game), please let me know as soon as possible so we can make alternative arrangements. If you are sick the day of, please provide me with documentation of your illness.

Extenuating circumstances

If you have an extenuating circumstance that will cause you to miss a lot of work, please notify me as soon as possible so we can discuss and negotiate an alternative schedule if I feel it is warranted.

ADA Policy

If you, as a student, believe that you qualify as a person with a disability as defined in Chapter 504 of the Rehabilitation Act of 1973, the Americans with Disabilities Act (ADA) of 1990, and the Americans with Disabilities Act Amendments Act of 2008 (ADAAA), you are strongly encouraged to register with the Accessibility Services Office located on the 2nd Floor of the Diane M. Halle Library and online at the Center for Accessibility Services website (https://www.endicott.edu/academics/academic-resources-support/accessibility). Faculty will

then be notified directly from the Accessibility Services Office of any approved academic accommodations including extended time eligibility.

Academic Integrity Statement

Students are required to abide by the *Academic Integrity Policy* of Endicott College, as described at https://catalog.endicott.edu/content.php?catoid=42&navoid=1757#aca-integrity

Cheating and plagiarism are different but related. You are cheating if you break rules for an assignment, e.g., if you use your notes during a closed-note quiz. Plagiarism is a specific type of cheating wherein you submit someone else's work (or work that has been heavily influenced by someone else) as your own (i.e., without giving credit to the true author). This could be someone else's answer to a quiz question, someone else's solution to a homework problem, etc. Both are bad and extremely unfair to your classmates who follow the rules. No one wants a race to the bottom where everyone feels they need to cheat because their classmates are cheating. It undermines the purpose of taking a course, where the goal is to learn and demonstrate your mastery of the material.

Cheating on a quiz the first time will result in failure of the quiz and a 10% penalty on your final course grade. An additional offense will result in your dismissal from the class.

The line between cheating and not cheating on a lab assignment is gray. While you and your partner may receive help from classmates, friends, family members, and tutors on a on aspects of a lab, you are not permitted to let them formulate the code for any part of your program (it doesn't matter if they write the code in your source code file directly or simply tell you what to write). The first instance of cheating will result in failure of the assignment and a 5% reduction on your final course grade; any additional instance will result in dismissal from the class.

To avoid crossing the line from acceptable collaboration to cheating, follow these tips. First, before you seek help, follow the steps from class on planning out a programming solution. Break the problem down and think about how each of those pieces need to connect. Write a source code skeleton and pseudo code for those components.Then tackle the code for each component one at a time. Run and test often.

If you have a problem with one of those steps, make an appointment with me during office hours to go over it. If you seek outside help. don't show your code and don't write anything down—just talk. This will help forge better connections in your brain and when you end up implementing what you talked about, you'll be doing it from your own understanding of the solution. If you are unsure how to program a small component, try to come up with a parallel example of what you're doing before you seek help. For instance, if you aren't sure how to sum all of the numbers in a list, think of a parallel example—maybe how to *multiply* all the numbers in a list—and ask for help on *that*. You can discuss and write code for parallel examples 'till the cows come home and you won't be cheating. This is because at the end, you need to transfer the solution to that parallel example over to your problem. Doing so demonstrates that you understand what's going on and it is you who wrote the code.

The one exception to not showing your code is if you have a bug that you cannot figure out; sometimes you need another pair of eyes to help find the cause and I'm okay with that in this course.

If you feel the need to cheat, that is a sign that you probably feel lost in the course. Please make an appointment with me so we can regroup and come up with a plan. As each unit in this course builds on the last, it is critical that you keep up with the material. If you feel lost at the beginning, see me during office hours and consult the course tutors.

All cheating/plagiarism offenses will be reported to the Provost's office.

Course Expectations

For each credit hour, students are expected to spend a minimum of two hours on work outside of class each week. For this four credit course, I expect you to spend, on average, about eight hours on this course each week outside of class meetings and labs.

You must review the Academic Calendar published by the Registrar's Office online at:

https://www.endicott.edu/academics/academic-resources-support/academic-calendar

Attendance and Participation

Class attendance is expected of all students (including you!) up to and including the last day of scheduled classes during the semester. You must plan accordingly. You will not be directly graded on attendance, however, certain activities such as participation, class activities, and lab grades are dependent on you being in attendance. If you know ahead of time that you cannot make your scheduled lab session, you may inquire with me about switching to the other section for that week only. I will send a progress report if you are accrue excessive absences and it appears to impact your performance in the class.

You may join class via Zoom if you are in official, Endicott-sanctioned COVID isolation. The Zoom link can be found in the "Class Resources" document linked to from Canvas.

While in class, you are expected to be fully present and engaged. Perpetually coming to class late or doing other things during class—texting friends, watching videos, writing a paper for another class, etc.—is rude and distracting to your classmates and me. If I notice it, you will lose your participation and class activity credit for that class period. Please be courteous to me and your classmates.

Subject to Change Statement

This syllabus is subject to change. I will notify you of any changes throughout the semester and highlight them above.

Changelog

- 2022-01-29
  - corrected the verbiage about redo tokens
  - added the final exam
- 2022-02-07
  - added that lowest quiz grade will be dropped (see the Evaluation Methods section)
- 2022-02-14
  - changed the date of Quiz 2
- 2022-02-18
  - modified the language around "graded for correctness" homeworks in Evaluation Methods; it's not longer the case that exactly 10 homeworks will be graded for correctness, and you will be given warning