FALL 2025 - CS 5001/5003 - Intensive Foundations of Computer Science. Online (4 Hours)

Course Code & Title	CS 5001: Intensive Foundations of Computer Science CS 5003: Recitation for CS 5001
Number of Credit Hours	4
Semester & Year of Offering	Fall 2025
Class Meeting Days & Times	No required meetings, weekly team activities, weekly optional live session, Thursdays 6PM ET
Class Meeting Location	Online
Required Textbook or Materials	No textbook required Python 3.10+ required, Github.com account required, MS Teams required

Instructor:
Albert Lionelle (he/him/his)
Director of Align, Associate Teaching Professor
Contact Information listed on **Course Home Page in Canvas**

Course Description

Introduces systematic problem solving through programming. Offers students an opportunity to learn how to analyze a problem, how to divide and organize the problem into appropriate components, how to describe the problem in a computer language, how to analyze and understand the behavior of their programs, and how to test that their programs are working correctly. Additionally, introduces a method of program design called object-oriented programming and various ways to organize data, including a discussion of their advantages and disadvantages. To practice the course concepts, students undertake assignments ranging from small, highly specified programming tasks to larger open-ended problems where students design and code their own solutions.

Course Prerequisites

The course is suitable for students in the ALIGN MS in CS program. It assumes no previous programming experience.

Corequisite(s): CS 5003 * CS 5003. Recitation for CS 5001. (0 Hours)
Provides a small-group discussion format to cover material in CS5001.

Course Objectives

1. Develop a basic understanding of how to analyze and break down large problems in order to implement efficient solutions using the Python programming language.
2. Determine functionality of code written by oneself and others through reading and tracing short segments of code.
3. Write correct and clearly-documented small-to-medium sized programs that others can read, understand, and modify.
4. Use generalization for data and functions to limit code duplication.
5. Develop tests to exercise implemented code to ensure quality testing in the software development process. Develop skill in debugging for when code does not perform as expected.
6. Select appropriate data types to represent information including using common library classes.
7. Assess the impact of data structure and algorithm choice on the run time and storage space needed to execute a program.

Course Structure

Week	Topic	Assessments
1	Residency	-
2	Getting Started	Homework 01
3	Conditionals	Team Activity, Homework 02
4	Functions	Team Activity, Homework 03
5	While loops	Team Activity, Homework 04
6	Strings and Lists	Team Activity, Homework 05
7	For Loops	Team Activity, Reflection, Code Walk due
8	Recursion	Team Activity, Homework 07
9	Files and Error Handling	Team Activity, Homework 08
10	Dictionaries and Sets	Team Activity, Homework 09
11	Classes and Objects	Team Activity, Final project discussion due
12	Stacks and Queues	Team Activity, Code Walk due
13	Catch up week	-
14	Sorting and Searching	-
15	Final Project	Final Project Due

Student Participation

It is extremely important for you to become engaged in this course. Like most programming courses, we are asking you to change the way you think so that you can express your thoughts to a computer. *This is not an easy task*. More importantly, this task takes **practice**, **repetition**, and **dedication**.

This course has a combination of videos, assignments, and team meetings. Teams will be assigned week 2, and students will be expected to meet with their team for once a week based on a time they arrange as a team. Students will be expected to participate in the team meetings, and will be expected to contribute to the team's work.

In general, not including time spent interacting with videos, you should be prepared to spend 3-4 hours per credit hour for this course. This means that you should plan on spending a minimum of 12-16 hours per week on this course. 16 hours is a rough average of 2.2 hours per day, every day of the week. Many students find this course takes about 20 hours/week to successfully complete. 20 hours a week is a rough average of 3 hours per day, every day of the week. Some students may spend more time than that on certain weeks. Time-on-task also does not always translate to work accomplished; if you find you are spending more time than this on the course, talk to the TAs about how to make your work and study time more efficient.

Please plan ahead! It can be hard to estimate when you might get stuck, so make sure to have extra slack time in your schedule to accommodate tricky problems or new concepts that are harder than you except. Sometimes a problem comes along that you really need to sleep on. Finish your work as early as you can, so that when problems come up that require extra time, you have that time to spend. If you have substantial work and/or family responsibilities ($>$ 24 hours per week), and you are also enrolled in CS5002, you might want to explore reducing your course load, if allowed, before the drop date.

Khoury Student Expectations

1. Respect should be shown in all communications and interactions with faculty, staff, industry, peers, and all others on campus. This includes respecting the preferred methods and response times of faculty and staff.
2. Students come to class prepared and having engaged with the online course materials.
3. Students are to actively participate in course activities and discussion.
4. Any issues that arise should be communicated to the appropriate faculty or staff member proactively.
5. All course interaction including instruction, TA advising, and course activities are to be done in English.
6. Students should come in to classroom with the goal of learning and have a "growth mindset".

Assessment

Course Assignments will be broken down into categories that can be found on the **Assignments page of the course canvas shell**. There you will find percentages of each category, along with policies for individual categories such as dropping the lowest grade or not. In general if we are dropping the lowest grades in a category, we will NOT drop the last assignment of the category (as it is essentially the culmination of all the other assignments).

In general assignments are either formative or summative. Formative assignments document your learning process, and there are often ways they can be redone showing that you learned the material. Summative assignments are your opportunity to demonstrate what you know, and are show your ability level on the topic. Summative assignments are often time restricted and have limited attempts.

Please note:

- Categories are subject to minor adjustments / changes throughout the semester.
- Category assignments will include a variety of assessment types, including participation / attendance activities
- Ungraded Assignments, there will be assignments that are ungraded. The most important goal is your learning, so you should not let grades or points affect your reason for doing something!

Late/Make-Up Policy

Homework activities will have a **Due Date** and **Available Until** date. The due date is the expected due date for the item. The available by date is a no questions asked late window (usually 3 days). This is to take into account different learning rates, standard accommodations, and sometimes we all need accommodation because of life. After the *available until* date is a hard cutoff, no exceptions (even if your internet goes down!). Why? Because the due date had already happened. Issues like internet going down, your dog ate your homework, etc are meant to be handled by the extra window already! For documented *extraordinary* circumstances (being hospitalized multiple days, etc), we will consider additional accommodations. However, these are only based on the **due date**, as you will have already have a week or more before the available until date.

Tier Mastery Grading (Formative)

Most formative assignments focus on Tier Mastery. Meaning the assignment is graded using a 4 point scale. Each point is mapped to a growth metric of

1. Learning
2. Approaching
3. Meets
4. Exceeds

Tests are then placed in those groups with increasing difficulty. If at any point you *fail* to pass all the tests in a group, grading *immediately* stops. However, you will be encouraged (and expected) to resubmit your assignment to eventually work your way through Mastery (ideally exceeds)!

Auto-graded assignments will have the opportunity from the day they are released, so you can check in regularly. Some assignments have a mix of auto-grading and hand-grading, in which case, it will often take a week for us to give feedback for the assignment to be redone. This means you are limited on manually graded assignments just based on realistic time constraints.

The goal with this type of grading is to give you goals that you can hit, and to let you focus on redoing work to make sure you get it done correctly. It provides an extreme level of flexibility, but it also means there isn't any "squeezing by" on your grade.

Resubmission Conditions

For homework assignments, you will be able to resubmit usually without meeting any conditions other than it being submitted before the "available until/late due date". For other assignments, you may need to meet conditions for resubmitting (going back and earning 4 points in all previous HW or meeting with an instructor) to resubmit.

Needless to say, unless specified otherwise if the available day has passed for any assignment with open resubmissions, that means resubmission windows are also not available!

Grading Scheme

All grades in this course, while on a point scale, translate into a final percentage scale once all categories are calculated. This is the following scheme we use.

Letter	High	Low
A	100	>= 90
B+	89.9	>= 87
B	86.9	>= 83
B-	82.9	>= 80
C+	79.9	>= 77
C	76.9	>= 73
C-	72.9	>= 70
F	69.9	0

The above grading scheme is slightly modified from your traditional +/- scheme as follows:

Grade boundaries are subject to change at the discretion of the instructor, but they will not become more difficult to achieve a grade.

To progress in the ALIGN program, students are required to meet the grade point average (GPA) requirements for the MS Computer Science – Align as determined by Khoury College of Computer Sciences. Currently, it requires a **B or higher** in all bridge courses!

Course Materials

The Canvas site for this course will be used to post weekly reading and video-watching assignments, lecture materials, labs, feedback, and grades.

Textbooks

Students are not required to purchase a textbook for this course. Two free online textbooks are listed on the Resources page in canvas, along with additional, recommended materials. While these are not the only materials available on the web, they are the ones that faculty members have identified as being most useful to students. Students are also encouraged to share resources that they find useful so that instructors may consider adding them to the list for future cohorts. It is important to read the materials and watch the videos before attempting the assignments.

Python

This class will be using the Python 3 programming language. You can download this for free from https://www.python.org/. By default, Python installs the Integrated Development and Learning Environment (IDLE) that can be used to do all of the development in this course. You are welcome (and encouraged) to install the most current version of python.

Visual Studio Code

For this semester, we will be using Visual Studio Code. While we won't be using the advanced features, we work with you on getting the system setup so you have a code editor working. As the semester progresses, we will add features for you to use. For more about installing VSCode, go to the public Resources page.

Gradescope

Gradescope is used in this course to manage some lab assignments and give students rapid feedback. Most Coding assignments will have a dedicated slot on Gradescope to accept submissions from students, to provide limited feedback to students before the deadline, and to provide manual feedback after grading. For any given assignment, please be sure you know whether to upload to Canvas or Gradescope. Follow all due dates listed **in canvas**, as Gradescope due dates may be later to accommodate syncing errors between systems.

MS Teams

Working in environments such as slack, discord, or MS teams is expected for most modern day programmers. It is expected students have MS Teams installed on their local system (don't just access through the browser). Please note, different instructors have different preferences for contact, and those preferences should be followed.

General Policies

5.2 Digital Etiquette (netiquette)

For much of what we do, even as face-to-face interactions become possible again, a primary form of communication in the modern world, especially in CS, is the written word. Because this means you are missing body language cues and immediate feedback from your "listener", it is very important to understand some common rules for good online etiquette: (adapted slightly from 7 Rules for Online Etiquette)

1. **Be respectful.** While it is easier to say hurtful or disrespectful things without standing face-to-face with someone, it is important to remember that your classmates and instructors are real people who are affected by the words you say and write. It is essential to keep in mind the feelings and opinions of others. If you wouldn't say it to someone's face, you shouldn't say it online either.
2. **Be aware of strong language, all caps, and exclamation points.** It is easy for written text to be misread and misunderstood. Have you ever sent a text message with good intention but your recipient thought you were being rude? If so, then you've experienced this firsthand. By being aware of strong language, you can identify potential confusions before sending messages. Tip: Read everything out loud before you send it.
3. **Be careful with humor and sarcasm.** Certainly, you shouldn't avoid being funny. We love to see your personality shine through in our classes. Many of the instructional staff are exceptionally funny too. But make certain that it is clear you are being funny and not being rude. Emoticons and smileys can be helpful when conveying humor or sarcasm so that it is read correctly.
4. **Yes, grammar and spelling matter.** While texting, "textspeak" can b gr8 4 ur friends. In an educational setting (even online), however, keep it a bit more formal. Your written communication should be professional and reflect proper writing style. Save written shortcuts and less than stellar grammar for Snapchat if you must, but follow grammar rules for school.
5. **Don't post or share (even privately) inappropriate material.** Enough said here, nothing is truly private online.

Be forgiving. Remember that not everyone will know these rules before posting. Try to be understanding of others when they struggle with written communication. It is very different than simply talking to a person face-to-face.

Academic Integrity and Scholarship

Everybody is aware of vast amount of knowledge that is available on the Internet. While every assignment in this class is designed to be solvable using the methods we have discussed in class, your Professor recognizes that every student will end up using the Internet at some point or another. In general, web searches should be limited to how to small tasks in Python. As a student, your job is to be honest and forthright with your efforts. It is of utmost importance to your learning that you never just cut-and-paste a solution to a homework problem; instead make the effort to understand the solution well enough to put it into your own words and be sure to cite your sources. Citations should include references (paper, website, or other) for any site that you used to research a solution. Ideally, proper APA or ACM format should be used. For websites this includes name of website, title of the article, the url, and the date of retrieval. If you find yourself spending any significant amount of time searching the web, you should speak with a TA or the Instructor, because it is a sign that something is not working for you in this class. If you ever feel a need to "copy code" from a web site, please think again; any code snippet should by very short and prefaced by inline comments stating where it came from and giving credit to the actual author.

> It is better to read and understand the code, close your web browser, reconstruct the ideas from memory, and then write your own version. Even then, give credit to the original author for helping you think it through.

You are expected to read, understand, and follow the University's policies on Academic Integrity. With the exception of explicit, group projects, such as Pair Programming (which we will teach), each student is expected to do his or her own work. Violations of academic integrity will result in a **negative grade** on the corresponding assignment, along with harsher penalties for more widespread problems, including automatic failing of a course.

Copying code is cheating and lacks integrity. CMU provides some nice examples to follow:

For personal assistance, here are some of the things that are appropriate:

- Clarifying ambiguities or vague points in class handouts, textbooks, or lectures
- Discussing or explaining the general class material
- Providing assistance with the programming language, in using the system facilities, or with editing and debugging tools
- Discussing the code that we give out on the assignment
- Discussing the assignments to better understand them
- Getting help from anyone concerning programming issues which are clearly more general than the specific project (e.g., what does a particular error message mean?)
- Suggesting solution strategies

- In general, oral collaboration is OK.

Here are some things that are inappropriate:

- Copying files or parts of files (such as source code, written text, or unit tests) from another person or source
- Copying (or retyping) files or parts of files with minor modifications such as style changes or minor logic modifications
- Allowing someone else to copy your code or written assignment, either in draft or final form
- Getting help that you do not fully understand, and from someone whom you do not acknowledge on your solution
- Writing, using, or submitting a program that attempts to alter or erase grading information or otherwise compromise security
- Copying someone else's files containing draft solutions, even if the file permissions are incorrectly set to allow it
- Lying to course staff
- Reading the current solution (handed out) if you will be handing in the current assignment late
- Copying prose or programs directly
- Giving copies of work to others
- Coaching others step-by-step

If you do any of these, your should also acknowledge it in what you turn in; but expect to have a conversation with an instructor about it and, at least, suffer some penalty in the grade. If we discover you have done this and not acknowledged it, the penalty will be much more severe. In other words, dishonesty is much worse than stupidity.

Here are some gray areas:

- Reading someone's code for clarity or bugs, after you have completed your own
- Helping with debugging
- Looking at someone's prose or program but thinking about them and writing your own
- Following someone's advice or instructions without understanding them
- Many others

These, too, should be acknowledged.

A few resources

- Cheating versus Collaboration
- CMU Policy to acknowledge their examples above.
- Northeastern University Citation and Academic Integrity Checklist - this does apply to your code!
- OSCCR - Office of Student Conduct and Conflict Resolution
- Northeastern Academic Integrity Policy

The university's academic integrity policy discusses actions regarded as violations and consequences for students http://www.northeastern.edu/osccr/academic-integrity.

Generative AI (Claude, ChatGPTm and Copilot)

> [!CAUTION] Generative AI is a powerful tool, it's use is allowed, but use it wisely. See below for this course's policy, and not every course has a different policy based on the learning goals.

For this class, you are allowed the use of claude or similar product for the use of small code snippets, studying, and writing tests. You are not allowed to use it write your entire assignment, and you will find that it will often try to redesign against the expected design. As such, we only recommend its use to help you better understand and learn syntax. If at any point we feel you are using generative AI to build your entire assignment or write your reports for you, this is considered plagiarism and will be treated as an academic integrity violation. Simply, it is a tool to facilitate your learning and efficiency in writing code. It should not be a substitute for understanding the concepts.

Remember the mantra:

1. Define
2. Document
3. Implement
4. Test

If you follow those steps, you will find that the first two are needed before generative AI can be successful, and for your own learning, it is best to implement until have a better understanding what it is attempting to do. Additionally, if a test is not well defined, it will often give you wrong results!

Disabilities; Reasonable Accommodations

Our goal is to ensure that every student should be able to participate in this course. Students with disabilities who wish to receive academic services and/or accommodations should visit the Disability Access Services (DAS) (previously known as DRS) or call (844) 688-6287. If you have already done so, please provide your letter from the DRC to the instructor – early in the semester – to arrange for those accommodations. If you do require any special accommodations, let the instructor know immediately so that appropriate details can be worked out.

Student Feedback

Your opinions are very important to me. All students are strongly encouraged to use the TRACE (Teacher Rating and Course Evaluation) system https://northeastern.bluera.com/ the end of the course to evaluate this course. A reminder about TRACE should arrive via email about two weeks before the end of the course. In addition, I will be asking for your feedback at least once

about halfway through the semester. However, if you have concerns about the course, don't wait until you are asked. You can contact me any time!

Academic Standing Process: Offering Additional Support

Given the unique academic structure of the Align program and reality that our students are coming from many diverse backgrounds with varying levels of math proficiency and CS exposure, it is important that policies and procedures are set in place that can support our students while maintaining academic standards set forth by the College. Also, during the first two semesters, the pace of the course is typically faster and truncated, which may present some of its own challenges.

Thus, the advising team and faculty work together as a Student Success Team to discuss students who could benefit from more support at the midpoint and end of the semester.

If there are concerns about a student's academic performance at the midpoint, an Academic Advisor will reach out to schedule a check-in and possibly recommend resources like dedicated tutoring or frequent check-ins. Although these are recommendations, note that it is a student's choice to accept or decline, as the goal is to empower students to make that decision based upon what's best for them.

The College has defined that a B grade or higher is required for students to move on, and we have found that the midpoint check-in serves as an effective way to intervene before it's too late.

As an additional measure, the Student Success Team will discuss the academic outcomes of students earning less than a B grade after final grades have been submitted and determine whether they can proceed to the subsequent course(s).

If you have any questions regarding the Academic Standing process, please contact your Academic Advisor.

Title IX

Title IX of the USA Education Amendments of 1972 protects individuals from sex or gender-based discrimination, including discrimination based on gender-identity, in educational programs and activities that receive federal financial assistance.

Northeastern's Title IX Policy prohibits Prohibited Offenses, which are defined as sexual harassment, sexual assault, relationship or domestic violence, and stalking. The Title IX Policy applies to the entire community, including male, female, transgender students, faculty and staff.

Faculty members are considered "responsible employees" at Northeastern University, meaning they are required to report all allegations of sex or gender-based discrimination to the Title IX Coordinator.

The university offers confidential resources for medical treatment, emotional support and counseling through Confidential Employees. Confidential Employees are not required to disclose information about Prohibited Offenses to the Title IX Coordinator without prior consent of the student. Confidential Resources on campus include University Health and Counseling Services (UHCS) staff, Sexual Violence Resource Center (SVRC), Office of Prevention and Education and the Center for Spiritual Dialogue and Service (CSDS). [From Title IV Policy, Section III.C]

Alleged violations can be reported to the Title IX Coordinator within The Office for University Equity and Compliance at: titleix@northeastern.edu and/or through NUPD (844) 688-6287.

Reporting Prohibited Offenses to NUPD does NOT commit the victim/affected party to future legal action.

In case of an emergency, please call 911.

Please visit The Office for University Equity and Compliance for the full Title IX Policy, a complete list of reporting options and resources both on and off-campus.