

CSCI 235

Software Design & Analysis II Hunter College - Spring 2024 Syllabus

Course Information:

Name: Dr. Tiziana Ligorio

email: tligorio@hunter.cuny.edu

Office: 1001K (through 1001L) Hunter North

Office Hours: posted on Blackboard

Course Webpage: https://tligorio.github.io/CSCI_235/

Please **visit this page regularly** for announcements, updates, lecture notes and schedule.

Course Meetings: Tuesday/Friday 10:00-11:15am, room HW 714

Course Credits: 3.

This 3-credit course requires 3 hours of classroom or direct faculty instruction and **on average 4-6 hours of out-of-class student work each week**. Out-of-class work may include, but is not limited to, required reading, programming projects, and studying for quizzes and exams.

Textbook

Data Abstraction and Problem Solving with C++: Walls and Mirrors, 7th Edition, Frank M. Carrano, ISBN-13: 978-0134463971.

Other useful resources:

- Open Data Structures in C++ (<http://opendatastructures.org/>)
- Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14, Scott Meyers
- Prof. Stewart Weiss' notes (http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci235/csci235_lecture_notes.php)

Course Objectives

This course is the second in a three-course series, and it is a major stepping-stone in your software development journey. The primary focus of this course is the **design and analysis of algorithms and abstract data types**. To this end it introduces elementary data structures with related algorithms and their use in problem solving. The course also covers core computer science concepts such as abstraction, algorithm complexity, performance analysis and the tradeoffs between running time, storage size, clarity and extensibility that are at the core of software design. As a sequel to CSCI 135 it will also enhance your programming skills in C++ and Object Oriented Programming by introducing new tools such as Templates, Inheritance, Polymorphism, extend your understanding of pointers and dynamic memory allocation.

Prerequisites

CSCI 135 and CSCI 150.

Communication

We will use Ed Discussion for communication regarding assignments and course content. Invitations will be issued via email the day before our first lecture. This should be your first go-to place when you have a question about course content or project assignments. Please **keep all conversation on this forum about course content ONLY**. To discuss any other course-related matter you may speak with the instructor during office hours. You must also **check your Hunter email regularly**. This is where you will receive important course-related communication.

Programming Projects

I hope you will enjoy the programming projects! There will be **7 programming projects, the lowest project grade will be dropped**. Every program must comply with the **Programming Guidelines** provided in a separate document. You **MUST READ** this document , it contains guidelines about **submission, lateness, plagiarism, grading** and **logging into your Linux accounts**.

Course Grading:

Component	Per Item %	Total %
Lecture Attendance		5%
Weekly Quizzes		5%
Programming Projects	Project 1 5%, all others 9% No project extra-credit available. The lowest project grade will be dropped. If project 1 is dropped, the remaining projects will count up to 50% maximum	50%
Exams	Midterm Exam 10%	40%
	Final Exam 30%	

There will be no makeup exams. If you miss the midterm exam due to a documented emergency, the final exam grade will replace it.

NOTE: MANY EXAM AND QUIZ QUESTIONS WILL BE DIRECTLY BASED ON THE PROGRAMMING PROJECTS

EXTRA CREDIT: up to +2 points towards the course grade will be awarded **proportional to attendance in the Technical Interview Prep workshops**. More information about the workshop will be announced in class and by email.

Suggestions for Success:

- **READ AHEAD:** Do the assigned reading (as per the tentative schedule on the course webpage) BEFORE class. It will help you understand and it will help you ask the right questions. The lectures may cover some material that is not in the book chapters. The lectures may also assume you know things that are covered in the assigned book chapters. You are responsible for ALL the material (lectures + book chapters). The only way to truly follow is to READ AHEAD.
- **ASK QUESTIONS:** Do not be shy to ask questions, it is the best way to learn, and there is a lot of support available to you!!! The first step is to post your question on the Ed Discussion. If you find that the same question has already been answered there, you are done! We will strive to answer questions that come up on the forum daily. I may also address relevant topics that come up on the forum in subsequent lectures as I see fit. If the help you need is not easily expressed/resolved with a single question, you may seek help from our wonderful TA's. **TA's** will be available for drop-in help in lab 1001B Hunter North (see the schedule on

the course webpage). Finally, you can always talk to me during office hours (listed at the top of this syllabus), or make an appointment.

- **GIVE YOURSELF PLENTY OF TIME:** For the course in general!!! Read and research on your own. Give yourself **MORE THAN PLENTY** of time for the programming projects. **Start working on a project as soon as it is released, you will need the whole allotted time.**
- **CODE CODE CODE.** There is nothing better you can do than **design/code/debug/test/code/debug/test/...** the more you do it the better you will get at it, and the more fun you will have!!!!
- **SUBMIT ALL ASSIGNMENTS ON TIME.**
- **STUDY WELL FOR EXAMS**, and make sure to attend the exam review lectures.

Syllabus Compliance

Except for changes that substantially affect implementation of the grading policy, **this syllabus** is a guide for the course and **is subject to change**. In particular, the course schedule on the webpage, including topics, readings and project assignments, is subject to change. Any changes will be announced in class and on the course webpage. Be sure to **check for updates online regularly**.

Hunter College Policy on Academic Integrity

Hunter College regards acts of academic dishonesty (e.g., plagiarism, cheating on examinations, obtaining unfair advantage, and falsification of records and official documents) as serious offenses against the values of intellectual honesty. The college is committed to enforcing the CUNY Policy on Academic Integrity and will pursue cases of academic dishonesty according to the Hunter College Academic Integrity Procedures.

In this course, special attention is given to contract cheating, where students have work completed on their behalf that is then submitted for academic credit. All submitted projects will be thoroughly checked for authenticity/originality with screening software in order to prevent contract cheating. Please read more information on Contract cheating from http://en.wikipedia.org/wiki/Contract_cheating

Clarification: There are plenty of resources and examples available that you may consult and understand to incorporate those ideas into your projects. However, you must ultimately write your programs yourself. What about generative AI? You may use it as a starting point, but **you must cite it**, and you must be very careful, it makes mistakes all the time! It makes mistakes that might be difficult to detect. If you use AI generated code, you must thoroughly test it and debug it, which means you must thoroughly understand it. Be careful, it may change your experience with coding, which does not reflect the abilities that will be expected of you on a

technical interview (can't use ChatGPT on your Google interview, not quite yet). You are strongly encouraged to work on the projects in groups and to discuss ideas with one other. However, unless otherwise stated, you may not give code to, or receive code from, anyone else. If you are uncertain about the appropriateness of a particular case, you may ask.

ADA Compliance

In compliance with the American Disability Act of 1990 (ADA) and with Section 504 of the Rehabilitation Act of 1973, Hunter College is committed to ensuring educational parity and accommodations for all students with documented disabilities and / or medical conditions. It is recommended that all students with documented disabilities (Emotional, Medical, Physical and / or Learning) consult the Office of AccessABILITY located in Room E1124 to secure necessary academic accommodations. For further information and assistance please call (212-772-4857)/TTY (212-650-3230).

Hunter College Policy on Sexual Misconduct

In compliance with the CUNY Policy on Sexual Misconduct, Hunter College reaffirms the prohibition of any sexual misconduct, which includes sexual violence, sexual harassment, and gender-based harassment retaliation against students, employees, or visitors, as well as certain intimate relationships. Students who have experienced any form of sexual violence on or off campus (including CUNY-sponsored trips and events) are entitled to the rights outlined in the Bill of Rights for Hunter College. Sexual Violence: Students are strongly encouraged to immediately report the incident by calling 911, contacting NYPD Special Victims Division Hotline (646-610-7272) or their local police precinct, or contacting the College's Public Safety Office (212-772-4444).

All Other Forms of Sexual Misconduct: Students are also encouraged to contact the College's Title IX Campus Coordinator, Dean John Rose (jtrose@hunter.cuny.edu or 212-650-3262) or Colleen Barry (colleen.barry@hunter.cuny.edu or 212-772-4534) and seek complimentary services through the Counseling and Wellness Services Office, Hunter East 1123.

<http://www2.cuny.edu/wp-content/uploads/sites/4/page-assets/about/administration/offices/legal-affairs/POLICY-ON-SEXUAL-MISCONDUCT-10.1.2015-with-links.pdf>

Departmental Learning Goals

This class satisfies the following learning goals, as set forth by the Computer Science department: (1a) Understanding the basic foundations and relevant applications of mathematics and statistics, particularly those branches related to computer science, by using mathematics to analyze algorithm performance. (1b) Understand the relationship between computer architectures and software systems. (2a) Deep practical knowledge of one widely used programming language (C++). (2c) Be able to apply principles of design and analysis in creating substantial programs.

Acknowledgments

The materials used in this course were adapted from materials publicly or personally shared with me. Many thanks to Simon Ayzman, Susan Epstein, Keith Schwarz, Ioannis Stamos and Stewart Weiss for kindly sharing and inspiring me with their work and knowledge.