CSCI 235

Software Analysis & Design II
Hunter College - Fall 2018

Instructor Information:

Name: Dr. Tiziana Ligorio
email: ligorio.tiziana@hunter.cuny.edu
        csci235.help@gmail.com
Office: Hunter North 1000-C
Tel: (212) 772-5213 (CS Department)
Office Hours: Wednesday 1:30 - 3:30pm, Room 1001-A Hunter North (*Please check the course webpage for announcements concerning office hours*)
                Or by appointment**

Textbook

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

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++ by introducing new tools such as Templates, Inheritance, Polymorphism, extend your understanding of pointers and dynamic memory allocation.

** I am usually available to meet by appointment on Thursdays before class at Silberman

Prerequisites

CSCI 135 and CSCI 150.

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

You can find course materials and updates here. Please check this page regularly for announcements, updates, lecture notes and schedule.

Blackboard

Please check the course page on Blackboard regularly. We will use the forum here to answer questions to FAQ about course material. 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**. You must also regularly check your Hunter email. This is where you will receive important course-related communication.

Programming Projects

I hope you will enjoy the programming projects, after all no one can write excellent software without actually doing a lot of programming! There will be **six** programming projects. Every program must comply with the ***Programming Rules*** provided in a separate document. This document also contains guidelines about **submission**, **lateness**, **plagiarism** and **grading**.

Course Grading:

Component	Per Item %		Total %
Programming Projects	10% per project Lowest grade dropped		50%
Exams	Max of	Avg(Midterm, Final)	50%
		Final Only	
YOU MUST PASS THE FINAL TO PASS THE COURSE			

MANY EXAM QUESTIONS WILL BE DIRECTLY BASED ON THE PROGRAMMING PROJECTS

Suggestions for Success:

- **READ AHEAD:** Do the assigned reading (as per the tentative schedule below) 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 Blackboard forum. If you find that the same question has already been answered on Blackboard, 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 Blackboard does not help, you can email csci235.help@gmail.com and the CSCI 235 staff will answer. If the help you need is not easily expressed/resolved with a single question, you may seek help from our wonderful TA's and Tutors. **TA's** will be available for drop-in help in labs at the Silberman campus before class and at 68th street on Wednesdays. Hours will be announced in class and on the course webpage, and are subject to change. **Tutors** are also available for drop-in tutoring at the Skirball Science Learning Center, on the 7th floor of Hunter East. You may check their website (https://library.hunter.cuny.edu/skirball-science-learning-center) for schedule and events, or drop in and look at their board for a daily-updated schedule. Finally, if you have questions that are not abut course material, you can 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 you 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.

UNIX Lab

All students enrolled in this class are given accounts on the Computer Science Department's network. **Students must reclaim existing Linux accounts by September 14th (see FAQ link below)**. This account entitles you to physical access to the Linux workstations in the labs on the 10th floor of Hunter North, and enables you to connect to any of the lab machines remotely via **ssh** from another computer. Many personal computers come with a version of ssh already installed. If yours does not, you can get one for free. There are several free versions of ssh–for example, OpenSSH for OpenBSD and PuTTY for Windows. Apple computers come with a command-line ssh client pre-installed (Terminal). Note, however, that the advantage of working in the lab is that you will be sitting at the console of the host and

will not be subject to network problems or potential disconnections that can take place when working remotely. The disadvantage is that you have to be in school to do this.

To learn more about logging in remotely, using Linux, following the lab rules, and dealing with possible issues, visit http://www.geography.hunter.cuny.edu/tbw/CS.Linux.Lab.FAQ/department_of_computer_science.faq.htm

Read the information on this page thoroughly if you have not done so already.

Please note that **all programming project submissions must compile and run without issue on the Linux lab machines**. These computers provide a common platform to evaluate program execution, free of issues related to OS or IDE. You should always confirm that your assignment code successfully compiles and executes on these machines before submitting.

Tentative Schedule:

LECTURE	DATE	TOPIC	READINGS	PROJECT
1	T, 8/28	Welcome / Intro	Syllabus, Programming Rules	#1 posted
2	TH, 8/30 **ROOM 115AB**	Abstraction / OOP	Chapter 1 Appendix B	
3	T, 9/4	Abstract Data Types / Templates / Intro to Inheritance	C++ Interlude 1	**#1 Due** #2 posted
4	TH, 9/6	Array-Based Implementation	Chapter 3	
	T, 9/11	**NO CLASS**		
5	TH, 9/13	Inheritance & Polymorphism	C++ Interlude 2	
	T, 9/18	**NO CLASS**		
6	TH, 9/20	-//-		**#2 Due** #3 posted
7	T, 9/25	Recursion	Chapter 2	
8	TH, 9/27 **ROOM 115AB**	Link-based Implementation	Chapter 4	
9	T, 10/2	-//-		

LECTURE	DATE	TOPIC	READINGS	PROJECT
10	TH, 10/4	Stacks	Chapters 6	
11	T, 10/9	Stack Implementation	Chapter 7	
12	TH, 10/11	Lists	Chapter 8	**#3 Due** #4 posted
13	T, 10/16	List Implementation	Chapter 9	
14	TH, 10/18	-//-		
	T, 10/23	Midterm Review		
	TH, 10/25	**Midterm**		
15	T, 10/30	Algorithm Efficiency	Chapter 10	
16	TH, 11/1	Sorting Algorithms and their Efficiency	Chapter 11	**#4 Due** #5 posted
17	T, 11/6	-//-		
18	TH, 11/8	Recursion as a Problem Solving Technique	Chapter 5	
19	T, 11/13	Queues and Priority Queues	Chapter 13	
20	TH, 11/15	Queue Implementation	Chapter 14	
21	T, 11/20	-//-		
	TH, 11/22	**NO CLASS**		#6 Posted
22	T, 11/27	Trees	Chapter 15	**#5 Due**
23	TH, 11/29	-//-		
24	T, 12/4	Tree Implementations	Chapter 16	
25	TH, 12/6	-//-		
26	T, 12/11	Final Review		
27	TH, 12/13	**Reading day NO CLASS**		**#6 Due**
	TBA	**Final**		

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, 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. You are actively encouraged 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.