



Software Design and Analysis III

CSCI 335 - Spring 2016

Class Hours: Monday/Thursday from 11:10 AM to 12:25 PM, in HN1516

Coordinator	Ioannis Stamos, istamos@hunter.cuny.edu
Instructor	Felix Grezes, fg297@hunter.cuny.edu
Webpage	http://speech.cs.qc.cuny.edu/~felix/Courses-Taught/Hunter-Spring2016-CSCI335.html

Textbook

Required: Data Structures and Algorithm Analysis in C++, 4 Edition, Mark Allen Weiss.

Recommended: Cormen, Leiserson, and Rivest, Introduction to Algorithms, McGraw-Hill (comprehensive and advanced).

Knuth, The Art of Computer Programming, Volumes I through III, Addison Wesley (valuable texts for computer scientists).

Algorithms in C++, Parts 1-4 and Part 5, Robert Sedgwick, AddisonWesley (in depth algorithm analysis).

Course Content and Objectives

This course is the sequel to CSCI 335, and as such, a major objective is to deepen and enhance your programming skills as well as knowledge of various advanced algorithms. The course will cover Algorithm Analysis, Advanced Trees, Special Priority Queues, Sorting Algorithms, Disjoint Sets, Graph Algorithms, Dynamic Programming, Randomized Algorithms, and Amortized Analysis. Beware though, you are expected to learn much of this material without much classroom time devoted to it.

Prerequisites

CSCI 235 and MATH 155.

How to NOT repeat this course

If you do not want to repeat this course, do *all* of the following:

- Do the assigned readings before the lecture, not after it.
- Make a list of questions before the class.
- Submit all assignments on time.
- Be well prepared for your oral presentation.
- Solve a set of problems at the end of each chapter.
- Study for exams.
- Do all assignments yourself! Expect severe penalties for plagiarism (see related section).

Syllabus and Readings

Which chapters and when they will be covered is described in the syllabus. You are responsible for everything in the listed chapters regardless of how much time we spend on them in class. As noted above, you should read ahead so that you can ask questions in class to clear up anything you find confusing.

Assignments

Every program must satisfy the programming rules stated in the "Programming Rules" (check Blackboard) document. The rules governing lateness, plagiarism, and how they are graded are also contained there. You are not allowed to work in groups! You can discuss about the assignments. However, copying of pieces of code among peers is a serious problem. It is critically important that all submitted program listings and executions be thoroughly documented. You are allowed to use the source code that is provided by the book. Some assignments may include theoretical questions.

All programs must compile and run. Zero credit will be given for programs that do not compile or do not run. Usually the homework assignments will only state the major objectives of the program to be written; it will be often up to you to make design decisions regarding I/O, efficiency, error handling, and so on. Make sure you provide adequate test cases to indicate the correctness and robustness of your approaches. In general, the failure of a grader to understand your work or to appreciate the thoroughness of its testing will be considered to be your error.

Assignments and Grading

I will assign five homeworks during the semester.

There will be one midterm exam and one final exam. The final exam is cumulative.

The exact formula combining those for a final grade will be determined at a later date.

Make-up Policy

All exams must be taken on time. Failure to take an exam counts as a zero grade on that exam. If you miss the midterm or final exam for a legitimate, documented medical emergency, I will find a way (if possible) to assist you.

Resources

Google C++ style guide: <http://googlestyleguide.googlecode.com/svn/trunk/cppguide.html>

Standard Libraries: STL <http://www.sgi.com/tech/stl/>

Containers cpreference.com/w/cpp/container

C++11 FAQ: <http://www.stroustrup.com/C++11FAQ.html>

Source code documentation generating tool (Doxygen): <http://www.stack.nl/~dimitri/doxygen/>

Learning Goals

This class satisfies the following learning goals as set forth by the Computer Science department:

(2a): Deep practical knowledge of one widely used programming language (C++).

(2b): Be experienced in working in at least two widely used operating system environments.

(2c): Be able to apply principles of design and analysis in creating substantial programs.

(1b): Understand the relationship between computer architectures and software systems.

(3a): Be able to communicate technical ideas effectively, both in writing and in oral presentations.

(3c): Be able to perform competitively on the Computer Science GRE exam.

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. Special attention is given to CONTRACT CHEATING (this is where students have work completed on their behalf which is then submitted for academic credit). Projects will be thoroughly checked for authenticity/originality with Turnitin.com.

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 (2127724857)/TTY (2126503230).

Tentative Schedule

Date	Topics	Readings	Deadlines
Monday Feb 1	L1. Introduction	Chapter 1	
Thursday Feb 4	L2. Algorithm Analysis (I)	Chapter 2	
Monday Feb 8	L3. Algorithm Analysis (II)	"	
Thursday Feb 11	L4. Review of List/Stack/Queue + Iterators	Chapter 3	
Monday Feb 15	Presidents' Day - College is closed		
Thursday Feb 18	L5. Trees (AVL)	4.1 - 4.3	HW #1 due
Monday Feb 22	L6. Trees (cont)	4.4	
Thursday Feb 25	L7. Trees (Splay Trees, B-Trees)	4.5-4.7	
Monday Feb 29	L8. Sets/Maps	4.8	
Thursday Mar 3	L9. Hashing	Chapter 5	
Monday Mar 7	L10. Hashing (cont)	"	HW #2 due
Thursday Mar 10	L11. Hashing (cont)	"	
Monday Mar 14	L12. Heaps	Chapter 6	
Thursday Mar 17	L13. Heaps (cont)	"	
Monday Mar 21	MIDTERM		
Thursday Mar 24	L14. Sorting	7.1 - 7.6	
Monday Mar 28	L15. Sorting	7.7 - 7.10	
Thursday Mar 31	L16. Union/Find	Chapter 8	
Monday Apr 4	L17. Graph Algorithms	9.1 - 9.3	HW#3 due
Thursday Apr 7	L18. Graph Algorithms (cont)	9.4 - 9.6	
Monday Apr 11	L19. Graph Algorithms (cont)		
Thursday Apr 14	L20. NP / NP - completeness	9.7	
Monday Apr 18	L21. Greedy Algorithms	10.1	
Thursday Apr 21	L21. Divide and Conquer	10.2	HW#4 due
Monday Apr 25	Spring Recess		
Thursday Apr 28	Spring Recess		
Monday May 2	L22. Divide and Conquer (cont)		
Thursday May 5	L23. Dynamic Programming	10.3	
Monday May 9	L24. Randomized Algorithms	10.4	
Thursday May 12	TBD		HW#5 due
Monday May 16	TBD		
Thursday May 19	Optional Study Session for Final		
Monday May 23	FINAL		11:30 - 1:30 pm

Academic Calendar for Spring 2016 available here: <http://www.hunter.cuny.edu/onestop/calendars>

Disclaimer

This document is a guideline, and subject to change.