Essentials: Communication, Content, and Structure

Communications

Class Meetings:	Monday, Thursday 11:10 - 12:25, Room HN 1516
Office:	HN 1090J
Office Hours:	Wednesdays, 10:30 - 12:30
Email:	stewart.weiss@hunter.cuny.edu
Telephone:	(212) 772-5469

You can see me during my office hours without an appointment. If you need to see me at a different time, you need an appointment. The best way to make an appointment is by email. I can sometimes schedule appointments if you see me before or after class. You can also call my office and leave a message. Regarding email, please note that I read only plain ASCII plain text messages, not HTML or MicroSoft Word-encoded documents. Also note that any email concerning anything that might fall under the FERPA regulations (e.g. questions about grades or other personal issues) must be sent from your "myhunter" account.

Resources

Textbooks: Frank M. Carrano and Janet J. Prichard, *Data Abstraction and Problem Solving with C++*, Addison Wesley, ISBN-13: 978-0134463971, ISBN-10: 0134463978. 7th edition. If you wish to obtain a previous edition, usually at a significantly smaller price, you may do so, but only if you accept the possibility that references to page numbers, exercises, and headings may be different.

Computing Facilities: Registered students are given user accounts on the Computer Science Department's network, which includes the Linux hosts in the 1001B lab of the Computer Science Department, located on the tenth floor of Hunter North. Access to this lab is limited to students enrolled in selected courses. In addition to the physical lab access, students will be able to use the secure remote login service, *ssh*, to access these accounts.

Website: All course materials, including lecture notes, slides, assignments, syllabi, and other resources, including this document, are posted on the course website, at `http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci235/csci235_fall17.php`

Discussion Board: This class will use *Piazza* as a discussion board. Please see the section below entitled "Course Materials, the Web, Piazza, and Blackboard" for the details.

Prerequisites

You are required to complete CSci 135, CSci 150, and Math 150 with a grade of C or better to take this course.

Departmental Learning Goals

Material in this course supports or partially supports the following departmental learning goals: 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; 2c: (ability to apply principles of design and analysis in creating substantial programs and have experience working in teams on projects of moderately realistic scope); 3a: (ability to communicate ideas effectively) by requiring homework that is graded in part on clarity and proper use of the English language.

Course and Learning Objectives

The primary objective of this course is to further your understanding of the design and analysis of algorithms and data structures. The course introduces the concept of abstract data types, as well as the specific abstractions of queues, stacks, lists, and various types of trees. It also introduces algorithms for certain common problems such as sorting, as well as algorithm paradigms such as backtracking and divide-and-conquer. More concretely, it will further your programming skills by covering recursion, pointers, and error handling. It will show you how to improve your software engineering skills, and it will give you practical experience for more productive programming. This course demands that you write more complex software than you did in Software Analysis & Design 1. In addition to the above concrete objectives, the course has certain "soft" skill objectives, including the use of version control, improving your ability to express ideas in written form and in spoken form, improving your ability to work as part of a community of software developers, and being able to identify the processes that you use to solve problems and explain these to others. A complete list of the expected outcomes for this course, as codified by the Computer Science Department, is contained in the document

`http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci235/csci235_curriculum.pdf`.

We may not cover all of the content described in that document, as it is more extensive than can be achieved in a one semester course. For a list of concrete topics that we will cover, see `http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci235/csci235fall17_schedule.pdf`.

Doing Well in This Course

If you want to do well in this course then you should do all of the following:

- Read the assigned reading ***before*** the lecture, not after it.
- Make a list of questions and ask them during the lecture. If I do not think a question is appropriate for the class, I will answer it at another time.
- Submit all assignments on time.
- Study for exams.
- Do as many of the textbook's sample questions as you have time to do.
- Do your assignments yourself.

Assignments, Exams, Grading, and Lateness

Your grade is computed strictly from three components, a programming component, an exam component, and a non-programming assignment component. The programming component is 20% of your grade and the exam component is 70%. Exams include quizzes. The assignment component is 10% of the grade. The weights for programs and exams are given below. Assignment weights will be determined during the semester depending on the number of assignments.

Programming Assignments

I will assign four programming projects during the semester. This is not enough to become proficient. If you want to be proficient and have the time, you should make up your own small problems and write test programs to solve them. Every program must satisfy the programming rules stated in the Programming Rules document provided on the course website. Programs will not be accepted beyond their due dates, without exception. The precise rules governing lateness, plagiarism, and the evaluation of programming projects are also contained there. I take plagiarism very seriously. The due dates and weights are listed below.

Programming Assignment	Weight Towards Final Grade	Due Date
1	4%	September 28
2	4%	October 26
3	6%	November 16
4	6%	December 11

Exams

There will be two midterm exams, one final exam, and some unspecified number of quizzes (pop quizzes). The exams will require that you write pseudo-code and possibly C++ code to describe algorithms and/or solve conceptual problems related to the course material. There are also conceptual questions on most exams, that may require short answers of various kinds. The quizzes will always be given at the start of class and last no more than five minutes. They will cover material you were expected to read for the class, or material that was covered in a preceding class. Quizzes are worth a total of 10% of the final grade.

Exam	Weight Towards Final Grade	Exam Date
1	15%	October 2
2	15%	November 6
3	30%	December 18, 11:30 - 13:30

Incomplete Grades

All assignments must be submitted by their due dates. Late assignments will not be accepted. Failure to take an exam counts as a zero grade on that exam. The only exceptions to these two rules are in the case that you have a legitimate medical or personal emergency that prevents your timely completion of homework or sitting for an exam and have notified me in a timely manner about this emergency. I will schedule a make-up exam or allow a homework extension only in that case. I do not give incomplete (IN) grades except to those students who were unable to complete the work because of legitimate, documented medical or personal problems, and this is entirely at my discretion.

Class Schedule

The document at

`http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci235/csci235fall17_schedule.pdf`

contains the detailed class schedule.

Class Calendar

There is no class on Monday September 4 (Labor Day), Thursday September 21, Monday October 9, and Thursday, November 23. The last day of class is Monday, December 11. Tuesday, September 19 follows a Thursday schedule and thus there is class.

Programming and System Access

All students enrolled in the class are given accounts on the Computer Science Department's network. This entitles you to around-the-clock access to the 1001B lab, which is equipped with Linux workstations. This lab is normally open "24/7". The account also enables you to work work from home or another remote computer by connecting to any of the lab machines remotely. The details are described below.

The advantage of working in the lab, as opposed to working remotely, is that you will be sitting at the console of a Linux host and will not be subject to potential disconnections that can take place when working remotely. You will also be much less affected by network problems than if you connect remotely from outside of Hunter. The disadvantage is that you have to be in school to do this.

When you are in the lab there are a few important rules that must be followed:

- Never power down a machine for any reason.
- Never leave a machine without logging out.
- Never use lockscreen to lock the screen in your login.

There are several other rules regarding lab use; they are posted there. Please take the time to read them and then follow them.

The advantages of working in the lab, as opposed to working remotely, are that you will be sitting at the console of a Linux host and will not be subject to potential disconnections that can take place when working remotely, you will also be much less affected by network problems than if you connect remotely from outside of Hunter, and you will learn how to use the Linux desktop environments. The disadvantage is that you have to be in school to do this.

When you are in the lab there are a few important rules that must be followed:

- Never power down a machine for any reason.
- Never leave a machine without logging out.
- Never use lockscreen to lock the screen in your login.

There are several other rules regarding lab use; they are posted there. Please take the time to read them and then follow them.

The Computer Science Department makes a UNIX host, named

`eniac.cs.hunter.cuny.edu`,

available to students who have accounts on the network. `eniac` is a gateway computer - you will be able to login to this host from any computer that has *ssh* client software and is on the Internet. Once you login to `eniac`, you must login from `eniac` to one of the computers in the Linux Lab, which are named `cslab1`, `cslab2`, `cslab3`, and so on, up to `cslab29`[1]. You cannot *ssh* directly to those machines from outside of Hunter College for security reasons. For example, you can first login to `eniac`, and then when it gives you a prompt such as "$", you would type

[1] or less than 29 if some machines have been taken out of service

```
ssh cslab5
```

and re-enter your network password at the prompt from `cslab5`.

Many computers come with a version of *ssh* already installed. If yours does not, you can get one for free. In particular, all Apple computers have the *ssh* client installed and available by opening the terminal application and typing the *ssh* command, such as

```
ssh eniac.cs.hunter.cuny.edu
```

There are several free versions of *ssh*. *OpenSSH* is an open source version developed for the *OpenBSD* project. *PuTTY ssh* is a free version for the Windows operating systems, available at

`http://www.chiark.greenend.org.uk/~sgtatham/putty/`.

Course Materials, the Web, Piazza, and Blackboard

All lecture notes will be posted on the course's home webpage (whose URL is above), which does not require special privileges to access. The only thing for which I use Blackboard is for posting of grades, which will be posted in the grade center there. This term we will be using Piazza for class discussion. The system is highly catered to getting you help fast and efficiently from classmates and me. Rather than emailing questions to me, you are to post your questions on Piazza. If you have any problems or need feedback for the developers, email *team@piazza.com*.

You can find our class page at:

`https://piazza.com/hunter.cuny/fall2017/csci23501/home`.

An invitation to join the Piazza discussion board will be sent to your Hunter College email address close to the start of the semester. You should accept this invitation. Your Hunter email address can be used for reading and sending messages to the group, or you can change the email address or add another on the settings page. In fact, you can request to join the group with any email address you choose, at

`http://piazza.com/hunter.cuny/fall2017/csci23501`

I require that you use the following protocol if you have a question:

1. Check whether the question you want to ask has been posted and answered on Piazza.
2. If it has been answered, you are finished. If not, post the question on Piazza.
3. Anyone in the class can answer the question. If no one else answers the question in a timely manner, I will post an answer to it.

I will ignore any non-personal questions sent to my Hunter email address. Personal questions (such as a questions about a grade or a missed class or alternative times to meet with me) should be sent via private email to my Hunter email address, not to Piazza.

Academic Honesty

Unless I state otherwise, all assignments and projects are to be your work alone. If someone else does part of this for you, it is considered to be academic dishonesty. 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 class, I will enforce the University's Policy on Academic Integrity and bring any violations that I discover to the attention of the Dean of Students Office.

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, the student can call (212-772-4857)/TTY (212-650- 3230).

Changes to This Syllabus

Except for changes that substantially affect the implementation of the grading statement, this syllabus is a guide for the course and is subject to change with advance notice. Any changes will be posted to the course website and to the Piazza group for the course.

CSci 235 Syllabus

Topics Covered and Their *Approximate* Schedule

My lecture notes, posted on the course website, should be your guide to this course. You must read them completely, in the order listed in the table below. The textbook serves as a more expansive treatment of the topics from my lecture notes. It has many examples, exercises, helpful "interludes", and visual resources to clarify the concepts. The book covers more than we can cover in our course and is a good book to own and to refer back to in your future courses.

The following table outlines the topics that we will cover during the semester and the order in which we will cover them. You should read the material in my notes and as much of the relevant chapter of the book *before the class in which the topics will be covered.*

Some topics listed in the CSci 235 curriculum document (CSci 235 Curriculum: v.6.0), such as information security, secure programming, and exception handling, are interspersed throughout the chapters. There are topics not explicitly listed in that curriculum document that are integrated into this class because I consider them to be inseparable from the fundamental process of software development. These include version control, software testing and quality, and soft skills important to *free and open source software* (FOSS) development. They are not explicitly listed below but will be part of the course, either as independent learning modules, homework assignments, or lab exercises.

The schedule outlined in the following table is only an approximation because we are using a new edition of the book, and because of changes in technology. Every class has a different make-up and learns different things at a different pace. All of this implies that the exact timing of topics may be different than what is listed below, but I have built in extra time to allow for variation.

Class	Date	Lecture Note Chapter Covered	Textbook Chapters
1	8/28	1 Administrative business/ Software Development	§1, Appendix B
2	8/31	1 Software Development	§1, Appendix B
3	9/7	3 Data Abstraction and C++ Classes	§1, Appendix A
4	9/11	2 Recursion	§2
5	9/14	2 Recursion	§2
6	9/18	5 Backtracking and Divide & Conquer	§5 (excludes 5.4)
7	9/19	8 Advanced Class Relationships	Interlude 1, Appendix A
8	9/25	8 Advanced Class Relationships	Interlude 1, Appendix A
9	9/28	4 Lists	§8
10	10/2	**Exam 1**	
11	10/5	4 List Implementations	§9, Interlude 2
12	10/12	4 Lists	§9
13	10/16	4 Lists	§9
14	10/19	6 Stacks	§6
15	10/23	6 Stacks	§6
16	10/26	6 Stacks	§6
17	10/30	7 Queues	§13
18	11/2	7 Queues	§14
19	11/6	**Exam 2**	
20	11/9	9 Algorithm Efficiency/Sorting	§10
21	11/13	9 Algorithm Efficiency/Sorting	§11
22	11/16	9 Algorithm Efficiency/Sorting	§11
23	11/20	9 Algorithm Efficiency/Sorting	§11
24	11/27	10 Trees	§15
25	11/30	10 Trees	§15
26	12/4	10 Trees	§16
27	12/7	10 Trees	§16
28	12/11	TBD	