

CS231 Course Information

1 Contact Information

Professor: Franklyn Turbak (please call me “Lyn”)
Office: SCI E121B (behind the mini-focus consultant’s desk)
Phone: x3049
E-mail: fturbak@wellesley.edu
Lectures: SCI E211, Tuesday/Wednesday/Friday, 8:30am – 9:40am
Office Hours: Tuesday: 4pm – 6pm
Wednesday: 8pm – 10pm
Thursday: 4pm – 6pm
Appointments can be made for other times. This semester, I will be spending most Mondays and Friday afternoons doing research, often at Boston University; I will almost never be available for appointments during these times. However, I am fairly flexible in terms of meeting at other times.
Web Site: <http://cs.wellesley.edu/~cs231>
First Class: CS231-S01 Announcements, CS231-S01 Q&A

2 Course Overview

Algorithms are recipes for solving computational problems. In CS231, we will study fundamental algorithms for solving a variety of problems, including sorting, searching, graph algorithms, computational geometry and cryptography. Even more important, we will focus on general design and analysis techniques that underly these algorithms: e.g., divide-and-conquer, dynamic programming, greediness, probabilistic approaches, and amortized analysis. You will get experience with these techniques by analyzing existing algorithms and designing some algorithms of your own.

Algorithms are judged not only by how well they solve a problem, but also by how effectively they use resources like time and space. We will learn techniques for analyzing the time and space complexity of algorithms and will use these to evaluate tradeoffs between different algorithms. We will also see that problems can be organized into a hierarchy that measures their inherent difficulty by the efficiency of the best possible algorithm for solving them.

3 Prerequisites

One prerequisite for CS231 is CS230, Data Structures. You should feel comfortable with simple data structures (arrays, lists, trees, stacks, queues) as well as iteration and recursion. However, unlike CS230, writing executable programs is not a focus of this course. Indeed, almost all the “programming” you will do will be via pencil and paper, using the pseudocode notation in the textbook. Coding in an executable programming language will only be required for certain extra credit problems.

A second prerequisite is a willingness to learn some math. In order to prove that algorithms are correct and to analyze the efficiency of algorithms, we'll need to use some mathematical methods that may be new to you. We will learn these methods in class, but they are also explained well in the textbook.

A third prerequisite is a willingness to work hard. The concepts in this course require serious mental energy to learn, and the problem sets are very challenging. Experience shows that most students work between six to ten hours on each problem set.

4 Textbook

Required: *Introduction to Algorithms*, by Cormen, Leiserson, Rivest, and Stein (2001). (Henceforth, I will refer to this book as *CLRS*.) This excellent and comprehensive textbook is a must for any computer scientist. It's a thick book, but don't worry — we're not going to cover the whole thing this semester! Most chapters are short and have good explanations, examples, and exercises. However, a few topics are somewhat dense; we'll spend some time unravelling these.

CLRS is the second edition of a book with the same name by Cormen, Leiserson, Rivest (henceforth *CLR*). We have used *CLR* in past versions of CS231. The differences between *CLRS* and *CLR* are not huge, so you are welcome to use a copy of *CLR* if you can get your hands on one. Assignments will specify the appropriate reading/problems from each of the two editions.

The book is expensive, but is one of the few textbooks that may actually be worth what it costs. You may be able to find a cheaper used copies at the Wellesley bookstore or at the MIT Coop. Another place to look is the "hurt books" section of the MIT Press Bookstore (which is right across the street from the MIT Coop). You may be able to borrow or buy a copy of the book from someone who took CS231 last fall. Finally, for those who may not wish to purchase the textbook, a copy will be on reserve in the Science Center Library.

To learn more about a particular subject or for hints on approaching a problem, you are encouraged to consult other algorithm and data structure textbooks. The Science Center Library has a large number of these, as does the Computer Science Resource Room (SCI 173).

5 Course Web Page and FirstClass Conferences

All course materials will be posted on the course web page at the following URL:

<http://cs.wellesley.edu/~cs231>

Additionally, there is a CS231-S01 conference in FirstClass with two subconferences:

- CS231-S01 **Announcements** will be used to make class announcements, such as corrections to assignments and clarifications of material discussed in class.
- CS231-S01 **Q&A** is a forum for you to post questions or comments. They will be answered by me or a classmate. This is also a good place to find people to form a study group.

You should plan on reading the CS231 conferences on a regular basis. It is strongly recommended that you add both subconferences to your FirstClass desktop

6 Homework

6.1 Problem Sets

There will be weekly problem sets that involve modifying, analyzing, and designing algorithms. All of these problem sets will be of the paper and pencil variety; no programming will be involved (except on some extra credit problems).

All problem sets are due at 11:59pm on the advertised due date, which will typically be a Friday. Problem sets will be graded on a 100 point scale. I will strive to have problem sets graded within a week of their submission. At this time, solutions will be distributed with the graded homework.

Problem sets will include three kinds of problems:

- **Suggested problems** are problems that you should think about, but you do not need to write them up and turn them in. They will typically be exercises from the book.
- **Required problems** are problems whose solutions you must write up and turn in.
- **Extra credit problems** are optional problems whose solutions you may write up for extra credit.

Your problem set grade will depend only on the required problems. However, doing the the suggested problems will help you understand the material better, and you will not have to spend any time writing them up. As motivation for doing the suggested problems, I offer the following pledge: at least one problem on the midterm exam will be a suggested problem from one of the problem sets.

To make up for points lost on problem sets and exams, students often request extra credit problems. In order to give everyone the same opportunity, I will often include extra credit problems on the problem sets. The extra credit problems are often more challenging than regular required problems, but they provide the opportunity to earn extra points toward your course grade. (An exception is implementation problems, in which you can write and test an executable program for algorithms that we have studied.)

Extra credit problems are entirely optional. Extra credit points count as problem set points, but they are only factored into course grades after I have partitioned the grade scale into letter grades. This means that doing the extra credit problems may raise your course grade, but not doing extra credit problems will not lower your course grade.

For maximum flexibility, you may turn in extra credit problems at any time during the term (through the end of finals week). However, experience has shown that students who leave extra credit problems until the end of the term rarely turn them in. It is in your best interest to complete extra credit problems in a timely fashion. I will not hand out solutions to extra credit problems, but you are encouraged to discuss them with me in person.

6.2 Problem Set Header Sheets

I would like to get a sense for how much time it takes you to do your CS231 problem sets. I use this information to design problem sets later in the semester, as well as for future semesters.

Please keep track of the time you spend on each problem of your problem sets, and include this information on the problem set header sheets that I will provide at the end of each problem set. Turn in this header sheet as the first page of your hardcopy submission.

6.3 Late Homework Policy

I realize that it is not always possible to turn in problem sets on time. On the other hand, turning in one problem set late can make it more difficult to turn in the next problem set on time. I have decided on the following policy for this course this term:

All problem sets will be due by 11:59pm on the later of (1) the specified due day (typically a Friday) and (2) the day on which the graded previous assignment is returned (if applicable). A problem set can be turned in n days late if it is accompanied by n Lateness Coupons.

At the end of this handout, you will find ten Lateness Coupons that you can use throughout the term. Use them wisely: you only get ten, and they are not copyable or transferable between students. (You also cannot use them on exams!)

You may turn in late problem sets by slipping them under my office door. Of course, if I post solutions before you turn in a late problem set, you are bound by the Honor Code not to examine these solutions.

In extenuating circumstances (e.g., sickness, personal crisis, family problems), you may request an extension without penalty. Such extensions are more likely to be granted if they are made before the due date.

6.4 Collaboration Policy

Since I believe that collaboration fosters a healthy and enjoyable educational environment, I encourage you to talk with other students about the course and to form study groups.

Unless otherwise instructed, feel free to discuss problem sets with other students and exchange ideas about how to solve them. However, there is a thin line between collaboration and plagiarizing the work of others. Therefore, **I require that you must compose your own solution to each assignment**. In particular, while you may discuss problems with your classmates, **you must always write up your own solutions from scratch**. It is **unacceptable** for two students to turn in copies (or near copies) of each other's solutions. I will interpret such a situation as a violation of the Honor Code, and will bring it before the General Judiciary. When in doubt about acceptable levels of collaboration, please ask me for clarification.

In keeping with the standards of the scientific community, you must give credit where credit is due. If you make use of an idea that was developed by (or jointly with) others, please reference them appropriately in your work. E.g., if person X gets a key idea for solving a problem from person Y , person X 's solution should begin with a note that says "I worked with Y on this problem" and should say "The main idea (due to Y) is ..." in the appropriate places. It is **unacceptable** for students to work together but not to acknowledge each other in their write-ups. The problem set header sheet includes a spot to list your collaborators.

When working on homework problems, it is perfectly reasonable to consult public literature (books, articles, etc.) for hints, techniques, and even solutions. However, you must cite any sources

that contribute to your solution. Assignments and solutions from previous terms of CS231 are **not** considered to be part of the “public” literature. You must refrain from looking at any solutions from previous terms of CS231. It is my policy that consulting solutions from previous terms of CS231 constitutes a violation of the Honor Code.

7 Exams

There will be two exams, both open book and open notes:

1. A take-home week-long midterm (date to be announced).
2. A final exam during the regular exam period.

8 Grades

The course grade will be computed as shown below:

Problem sets (total)	60%
Midterm exam	20%
Final exam	20%

The default ranges for grades are expressed as a percentage of total points (excluding extra credit points):

A	93.33 – 100
A-	90 – 93.32
B+	86.66 – 89.99
B	83.33 – 86.65
B-	80 – 83.32
C+	76.66-79.99
C	73.33-76.65
C-	70 – 73.32
D	60 – 69.99
F	below 60

I reserve the right to lower boundaries between grades, but I will not raise them. This means that I can grade on a curve, but only in your favor.

The above information is intended to tell you how I grade. It is not intended to encourage a preoccupation with point accumulation. You should focus on learning the material; the grade will take care of itself. If you are dissatisfied with the grade you will receive based on the above scale, I encourage you to turn in extra credit problems to raise your grade.

9 Finding Help

If you have any questions at all about the class (whether big or small, whether on problem sets, lectures, reading, or whatever) please contact me. That's what I'm here for!

The best time to see me is during my scheduled office hours (which are listed at the top of this handout). If these times are not convenient, we can set up an appointment at some other time. You can set up an appointment by talking with me in person, calling me on the phone, or sending me email. You can also ask questions by sending me email. I read my email on a regular basis, and will check it even more frequently in the few days before an assignment is due.

Drop-in tutors are available to answer your questions during certain hours. The names and schedules of the drop-in tutors will be made available early in the term. If you are having trouble with the course, you can request a one-on-one tutor from the Learning and Teaching Center (LTC). This service is confidential and free of charge; please take advantage of it if you would like some extra help! Contact me or LTC for more information about this service.

Finally, when looking for help, don't overlook your fellow students — not only those who have taken the course in the past, but your classmates as well. Your classmates are a valuable resource; make good use of them!

10 Students With Special Needs

If you have any disabilities (including hidden ones, like learning disabilities), I encourage you to meet with me so that we can discuss accommodations that may be helpful to you.

LATENESS COUPONS

Below are ten Lateness Coupons. A problem set that is n days late must be accompanied with n Lateness Coupons in order to be accepted. That is, each coupon gives you one extra day to turn in a problem set. You may use them in any manner in which you wish – e.g., turn in every problem set one day late, or turn in one problem set ten days late. Lateness coupons are not transferable between students, and may not be used on exams.

CS231 Lateness Coupon #1
CS231 Lateness Coupon #2
CS231 Lateness Coupon #3
CS231 Lateness Coupon #4
CS231 Lateness Coupon #5
CS231 Lateness Coupon #6
CS231 Lateness Coupon #7
CS231 Lateness Coupon #8
CS231 Lateness Coupon #9
CS231 Lateness Coupon #10