CSCI 311: Design & Analysis of Algorithms
Syllabus

Prof. Edward Talmage

Fall 2024
Note that this syllabus is subject to change.

Instructor Edward Talmage

- Office Hours (Dana 329): Monday 10AM-12PM, Wednesday, 9:30-11:30AM,3-4:30PM, Thursday 10:30-11:30AM, or by appointment.
- Email: [first].[last] [at] bucknell.edu

  Please include the text "CSCI 311" in the subject line, as well as a title indicating the email's subject. Example: "CSCI 311: Question about Dynamic Programming".
- Discord: etalmage

Schedule

- Lecture, Section 01: Monday, Wednesday, Friday, 1-1:50PM, Dana 132
- Recitation, Section 40: Thursday, 1-1:15PM, Dana 134
- Other section With Prof. Gutekunst:
  - Lecture, Section 02: Monday, Wednesday, Friday, 3-3:50PM, BRKI 165
  - Recitation, Section 41: Thursday, 3-3:50PM, Dana 134

Textbook

- *Introduction to Algorithms, 4th Edition*, Cormen, Leiserson, Rivest, Stein, MIT Press 2022, ISBN 026204630X/978-0262046305
- *Introduction to Algorithms, 3rd Edition*, Cormen, Leiserson, Rivest, Stein, MIT Press 2009, ISBN 9978-0-262-03384-8/978-0-262-53305-8
  - Either 3rd or 4th edition should work for this course.
  - The book is often/currently significantly cheaper on Amazon than at the bookstore.
  - The library has access to these as eBooks. I encourage you to purchase a physical copy if/when you are able, as this book is very useful as a reference and for self-education in your further classes and career after Bucknell.
- *Book of Proof, 3rd Edition*, Hammack, ISBN 978-0-9894721-2-8/978-0-9894721-3-5, `https://richardhammack.github.io/BookOfProof/`
  - This is an **optional** resource, but may be very useful if you are not super comfortable with proofs.
  - This book is available in print, or freely from the author's website, linked above.

Tools We will use the following online systems for this course:

- **Website**: `https://csci.courses.bucknell.edu/311/csci-311-01-fall-24/` I will post course resources (syllabus, notes, examples, etc.) here.
- **Moodle**: Learning Management System `moodle.bucknell.edu`. I will post assignments (homework and recitations) here.
- **Gradescope**: Assignment submission and grading system. Using gradescope makes grading *vastly* more manageable for me. You'll submit assignments and take quizzes here. `gradescope.com`. Your login will be your Bucknell email, and you should go to `https://gradescope.com/reset_password` initially to set up your password if you have not used gradescope before.
- **Overleaf** [optional]: Online, collaborative LaTeX editor. `overleaf.com`. I recommend you use this for writing documents (e.g. hw solutions) for this class, as you can collaborate with a partner and give me access to your LaTeX code for debugging.
- **Discord** [optional]: Online chat client, freely available at `https://discordapp.com/`. This is one way to reach me and connect with other students in the department.

Course Goals This course explores the concepts and methods of solving problems with computers. To that end, we explore a variety of standard tools and techniques which apply to a wide variety of problems. We emphasize the general over the specific, in that we want you to be able to use the knowledge you gain in this class to solve new and interesting problems, not just to regurgitate the solutions to the specific problems we consider in this class.

You will be able to approach a new problem and evaluate which of several standard techniques may apply. You will then be able to describe how to apply those techniques, separately or in combination, to solve the problem. Finally, you will be able to write proofs of your solution's correctness and efficiency. You will also be able to analyze an existing algorithm's correctness and runtime.

After taking this course, you should be able to:

- Analyze the performance of algorithms using standard techniques
- Apply known algorithms and algorithmic techniques to solve real-world problems
- Justify algorithmic solutions to computing problems
- Typeset a basic LaTeX document to clearly present ideas

Class Culture Having stated the course's learning goals, I want to clarify that the intent of this class is for every student to achieve them all. To that end, this class is a shared effort. I commit to working with you to help you achieve the course's learning goals. But I am also learning from you, whether new approaches to a solution or the limitations of my personal style and perspectives. Similarly, you are here to help each other learn. Every member of the class has a contribution, whether you grasp something quickly or, more often, when you don't. I expect you to support every one of your classmates, both in assigned groups and informally. Seek out opportunities to help and be helped by your classmates. While I would be the first to feel awkward reaching out to someone I don't know, in this class you are invited and expected to form connections with every member.

Every human is inherently valuable, and I expect you to treat all others, in this course and outside, in light of that reality. There are many aspects of our lives, both small enough to hardly notice and large enough we don't recognize them as abnormal, in which we put others down and treat than as valueless. As part of our education, we need to practice being cognizant of the effects our words, actions, and inactions have on those around us, both directly and indirectly. This is hard, but we can teach each other. If you feel I have offended, devalued, or mistreated you, I promise to listen to your concern and do my best to correct it. I expect you to extend similar grace to each other, and ask that you seek to improve yourself and your classmates. Remember, gentleness and humility go a long way!

Health Concerns If you are sick, please **do not attend class**. When you miss class, you are expected to communicate with me to answer any questions you may have and ensure you have a complete understanding.

The question will naturally arise: "How sick is too sick to come to class?" This is not an easy question to answer and you will need to use your best judgement, but I encoursage you to err on the side of safety. At a minimum, if you have a fever ($\geq 100^\circ$ F), please do not attend. If you have a minor illness, please wait until you have been fever-free for 24 hours before returning to class (`https://www.cdc.gov/flu/business/stay-home-when-sick.htm`). If you are missing class because you are ill, please notify me by email as soon as is reasonably possible.

A single person's actions and decisions may have small or no consequences for themselves, but may cause immeasurable harm to others in our community. Please, be safe and care even more for those around you than for yourself.

Assignments, Their Purpose and Grading

- Problem Sets (10 points): Weekly. Practice to develop a deep understanding of the material and your problem-solving skills. **Submissions must be typeset with LaTeX.**.
- Quizzes (20 points): Weekly. Validate your understanding of definitions and basic concepts. If you miss a question, you will have a chance to make it up the next week.
- Exams (10/10/10/20 points): Three "midterm" exams, one final. Evaluation of your progress towards the course learning goals.
- Participation (10 points): This refers to **active** engagement and interaction, not just attendance. This can be in-class, such as question asking and answering and interaction during in-class exercises, or out of class, attending office hours, asking questions via email or other means, or some other method. I need to see that you are engaging with the course and material, though it may look different for different students.
- Recitations (10 points): Interactive and group work to reinforce understanding and explore applications. Recitations are about practicing the ideas and techniques from the class, not getting the right solution. You will be solving problems in small groups, and your participation in those groups and effort towards solutions will be the basis of your grade, not your solutions. You will not submit recitation work.
- Programming Projects (10 points): 2 projects during the semester. Apply the algorithmic ideas we have learned.

Quizzes There will be weekly quizzes on Gradescope. A quiz will typically have four questions, each on a topic from lecture that week. The following week, in addition to a new quiz on the new material, there will be an opportunity to retake any missed quiz questions. Retake *only* those questions you got wrong on the original quiz. For example, if in Quiz 3 you get questions 1 and 4 correct, but 2 and 3 incorrect, then complete questions 2 and 3 in Quiz 3 - Retake, leaving questions 1 and 4 blank. Each of the four questions is worth the same amount on the original and retake quiz, so utilize the feedback to re-study any topics you got wrong and get the points on the retake. This includes asking me in office hours how to solve questions you did not understand the first time. There is historically a very strong correlation between quiz grades (particularly retake grades) and final grades.

Course Grade and Curves I do not curve individual assignments. At the end of the semester, I will curve all grades as needed, though will never curve down. There are also 110 points possible, though I will grade out of 100 to determine final grades. This is to give you multiple paths to success, and so that a single bad grade cannot ruin your course grade. All assignments are mandatory.

Late Policy Homework and project assignments are due promptly at the listed time and will lose 10% per 24 hours, rounded up, that pass after the due date. You may submit an assignment at most 48 hours late, without special permission. Beware: If we discuss solutions in class the day an assignment is due, no late assignments will be accepted, so if you know you will be submitting something late, email in advance of the due date. **Quizzes may not be submitted late.**

If you have a conflict with an exam, please inform me at the beginning of the semester and remind me at least a week in advance. There is no guarantee that you will be able to make up a quiz or exam missed without prior permission.

Homework Format Solutions must be typeset in LaTeX and uploaded as a PDF to gradescope. If you are unfamiliar with LaTeX, please see the resources on the course page and ask the professor for help. We will have a primer in our first recitation.

Cooperation No cooperation or communication of any sort is allowed on quizzes and exams, except for asking me to clarify a question. Cooperation on problem sets and projects is allowed, but you should first attempt to solve problems on your own, as the harder you work to solve a problem, the more you will learn.

You **must** include a complete list of collaborators and sources in your submission, which includes everyone in any group you worked with and any website or resource (including AI) from which you obtained relevant information. The LaTeX `\footnote{}` command will serve you well.

Programming projects will be pair- or group-based. You may have a conversation with someone outside your group, but do not share or look at each others' code.

You may look up **definitions of terms** online, though you are probably better served by using the textbook or asking the professor. You may not ever look up a problem's solution online. Ask another student in the class or the professor if you're stuck.

Disabilities If you need accommodations for any form of disability, please speak to Bucknell's Accessibility Services **first**, and then **also** the instructor ASAP so that we can provide appropriate accommodations. We want to provide the best learning environment possible for every student, but need your input to know how best to do so.

Religious Observance If some aspect of our class conflicts with your religious observances, please inform me as soon as possible. I will happily work with you to find another way for you to complete the class objectives.

Approximate Schedule This **will** change without further notice depending on the pacing of class.

#	Week of	Topic	Chapters	Notes
1	8/26	Introduction	1-2	
2	9/2	Proof Techniques		
3	9/9	Runtime Analysis	3,4	
4	9/16	Divide & Conquer	15	Exam 1 (Friday)
5	9/23	Dynamic Programming 1	15	
6	9/30	Dynamic Programming 2	15	
7	10/7	Greedy Algorithms	16	
8	10/14	Greedy Wrapup		No Class Monday (Fall Break)
9	10/21	ADTs, Search Trees	12,13,18	Exam 2 (Monday?)
10	10/28	Balanced Search Trees	17	
11	11/4	Amortized Analysis 1	17	
12	11/11	Amortized Analysis 2/Graph Algorithms 1	22	Exam 3 (Wednesday?)
13	11/18	Graph Algorithms 2	23-24	
15	11/25	Thanksgiving Break - No Class		
14	12/2	Graph Algorithms 3, P/NP	25, 26, 34	
16	12/9	Undecidability		