Course Information

Course Overview

Are there "laws of physics" in computing? Are there fundamental restrictions to what computers can and cannot do? If so, what do these restrictions look like? What would make one problem intrinsically harder to solve than another? And what would such restrictions mean for our ability to computationally solve meaningful problems?

In CS103, we'll explore the answers to these important questions. We'll begin with an introduction to mathematical proofs and discrete structures, which will enable us to model problems that arise in computer science. In the course of doing so, we'll explore mathematical logic, discrete structures, and the mathematical nature of infinity.

We'll continue by exploring finite automata (mathematical models of computers with finite memory) and from there will explore context-free grammars and Turing machines (mathematical models of computers with unbounded memory). As we explore these models, we'll see their strengths and their weaknesses and will explore questions like "what does it mean to solve a problem?" and "why does this problem seem to resist a solution?" Finally, we'll conclude with a quick introduction to complexity theory and explore what we know – and what we don't – about efficient computation.

In the course of the quarter, you'll see some of the most impressive (and intellectually beautiful) mathematical results of the last 150 years. You'll see what proof-based mathematics is all about and will gain confidence using mathematics to model and solve problems. You'll learn about various discrete structures that arise throughout computer science. You'll learn how to think about computation itself and how to show that certain problems are impossible to solve. Finally, you'll get a sense of what lies on the frontier of computer science, especially with regards to the **P** $\stackrel{?}{=}$ **NP** problem.

Instructor

Keith Schwarz (htiek@cs.stanford.edu)
Office: Gates 178
Office Phone: (650) 723-4350

TAs

Fidel Salgado (fidels@stanford.edu) *(Head TA)*

Amy Liu
Chioma Agu
Diego Hernandez
Guy Amdur
Ingerid Fosli
Ivan Suarez Robles
John Wang
Jordan Hank
Karen Wang
Kenny Xu
Kyle Gulshen
Nick Guo
Robbie Ostrow
Shalom Rottman-Yang
Sudarshan Seshadri

Website

The course website is cs103.stanford.edu and it's loaded with resources for this course. There, you'll find all the handouts and lecture slides, along with additional links you may find useful. I would suggest periodically polling the website to stay on top of any important developments in the course.

Email

The course staff can be reached at cs103-aut1718-staff@lists.stanford.edu. Please don't hesitate to email us! We're here because we genuinely love this material and want to share it with you. If you have any questions on the material, or if you're interested in exploring more advanced content, please get in touch with us. We'd be happy to help out.

Piazza

We have a class Piazza forum you can use to ask questions about the material and to get help and advice on the problem sets and discussion problems. Our policies regarding Piazza use are covered in a handout on problem set policies.

Lectures

Mondays, Wednesdays, and Fridays, 3:00 – 4:20 in Nvidia Auditorium. Lectures will be recorded and are available for SCPD students. If you're an on-campus student, we ***strongly recommend*** attending lectures in person if at all possible.

Units

If you are an undergraduate or taking this class through SCPD, you need to enroll in CS103 for five units (this is department and university policy). If you are a matriculated graduate student, you may enroll for anywhere between three and five units, depending on what best fits into your schedule. Regardless of how many units you are enrolled for, the course content and requirements will be the same. The unit flexibility is simply to make life easier for graduate students.

Five-unit courses at Stanford vary greatly in their difficulty. Based on past student experiences, you should expect that this course probably will require a time investment proportional to its unit load. Expect to put in a total of 15 hours each week – including lecture time – working on CS103. We'll offer a lot of support through office hours, extra practice problems, and practice exams, and if you're willing to put in the effort to learn the material, the course staff will be behind you every step of the way.

Prerequisites

Starting this quarter, CS103 has CS106B/X as a prerequisite or corequisite. This means that if you want to take CS103, you must either have completed or be concurrently enrolled in one of CS106B or CS106X (or have equivalent background experience).

Over the course of the quarter, we will be giving out a number of small programming assignments to help you better understand the concepts from the course. Those assignments will assume a familiarity with C++ and programming concepts (especially recursion) a level that's beyond what's typically covered in CS106A.

Although CS103 is a course on the mathematical theory behind computer science, the only actual math we'll need as a prerequisite is high-school algebra. We'll build up all the remaining mathematical machinery we need as we go. We've released another handout detailing the mathematical prerequisites for this course, so if you have any questions, check it out and see what you find!

If you're interested in taking this course but feel that you might not have a sufficient mathematical background, you may want to check out our new add-on course, CS103A, which is discussed later in this handout.

Office Hours

The TAs and I will be holding *lots* of office hours during the week so that you can stop by and ask questions about the material. Feel free to stop on by if you need any help. We will also hold dedicated sets of office hours for our SCPD students. We'll post a schedule later this week.

CS103A

CS103A is an optional, one-unit add-on course for CS103. CS103A meets once a week for two hours and offers extra review and practice problems related to the current course content. If you're interested in taking CS103 but feel like you might need a little bit of extra practice and review, I'd strongly recommend checking out CS103A.

Unfortunately, CS103A is not available to SCPD students. However, all the materials used in CS103A will be available on the CS103A website, https://cs103a.stanford.edu.

Readings

There are online course notes for the first few weeks of material. They go into a *lot* more depth than what we're going to end up covering in CS103, but hopefully you'll find them useful for getting a deeper understanding of the material. The course notes are still a work in progress, so please feel free to contact us with corrections of all sorts – logic errors, grammatical issues, formatting problems, etc. We also will release a bunch of handouts over the quarter to provide additional supplementary reading material. Additionally, we'll release a number of graphical guides to various concepts covered throughout the quarter.

There are two *recommended* textbooks for this quarter. The first is ***How to Read and Do Proofs*** by Daniel Solow, which is a great resource for learning how to approach mathematical problem-solving. The second is ***Introduction to the Theory of Computation, Third Edition*** by Michael Sipser. You might find this book useful in the second half of the quarter. Some of the readings in the syllabus are taken from this book, but we will not directly test you on any material in Sipser that is not covered as well in lecture or the problem sets.

There are copies of each of these books in reserve in the Engineering Library.

Problem Sets

There will be ten total problem sets in CS103, given out about once per week. You are welcome to work on them individually or in pairs. Our full policies with regards to problem sets (late policy, regrades, etc.) are in the Problem Set Policies handout.

Exams

In addition to problem sets, there will be a two midterm exams and a final exam. The first midterm exam will be held on Monday, October 23rd from 7PM – 10PM and the second on Monday, November 13th from 7PM – 10PM, both locations TBA. The final exam will be held on Monday, December 11th from 3:30PM – 6:30PM, location TBA.

SCPD students will be able to begin the midterm exams any time in the 24-hour window leading up to the start of the regularly-schedule exam time. If you're an SCPD student and happen to be in the area, you're welcome to take the exams on campus.

In accordance with university policy, with the exception of OAE accommodations, we will not offer any alternate final exam times. If you are unable to take the final exam at the stated time, you will need to take this class in another quarter.

Additionally, with the exception of OAE accommodations, we generally do not offer alternate midterm exam times. You should not enroll in CS103 unless you can make all three of the exam times.

Grading

Overall, your grade for this course will be determined as follows: the problem sets are collectively worth one-third of your grade, the midterms are collectively worth one-third of your grade, and the final exam is worth the remaining one-third of your grade.

Your problem set grade is computing by summing up the total number of points you earned across the problem sets and dividing by the total number of possible points. The denominator excludes extra credit problems.

Your midterm score is computed by weighing your two midterm scores as follows:

$$\text{Midterm Score} = {}^{2}/_{3} \cdot \text{Higher Midterm Score} + {}^{1}/_{3} \cdot \text{Lower Midterm Score}$$

For example, if you earned a 75% on the first midterm and a 90% on the second midterm, your midterm score would be an 85%. If you earned a 100% on the first midterm and a 50% on the second midterm, your midterm score would be an 83.3%. Our intent is that the midterms serve as a feedback mechanism that let you get a better feel for where you stand and what you need to improve, and we hope this scoring system reflects this.

Aside from the midterm calculation described above, we do *not* curve grades on individual assignments or exams. Rather, we use raw point totals weighted by the amounts given above to compute everyone's raw total score, then curve raw total scores. Historically, we've used the median raw score as the B/B+ cutoff.

Unlike some other courses, we will ***not*** drop your lowest problem set score and do not offer any make-up work. Your raw score will be computed purely by weighting your raw scores by the above amounts, and we will determine final grades based on those raw scores.

Many problem sets include an extra credit problem that you're welcome to work on if you'd like an extra challenge. When computing final grades, we determine the grading curve *before* factoring in extra credit points, so not doing the extra credit problems will not put you at a disadvantage. The extra credit points are purely a bonus.

Honor Code

We want to foster a collaborative and supportive atmosphere in CS103. We want you to really learn this material – and, ideally, we hope that you'll find it as exciting as we do. This is why, for example, we have so many office hours sections and why we let you work in pairs on the assignments.

That said, we expect you to abide by the letter and the spirit of the Stanford Honor Code in CS103. You are required to read and abide by the policies detailed in our handout on the Honor Code as it applies in CS103, which among other things discusses our expectations for what is and is not permissible collaboration on the problem sets.

We hope that you will respect the Honor Code, comport yourself with integrity, and work to create a learning environment where everyone feels supported.

Incomplete Policy

If you have a serious medical or family emergency and cannot complete the work in this course, you may contact Keith to request an incomplete. Our policy is to reserve incompletes only for emergencies, so we do not grant incomplete grades for poor performance on the assignments or exams, nor do we offer incompletes for busy work schedules.

In order to be eligible for an incomplete, you must have completed all of the assignments (except possibly the most-recently-due assignment) and must have a performance that is roughly on par with a B- overall grade.