MATHEMATICAL INSIGHTS IN COMPUTING

Chelsea Voss
csvoss@mit.edu
Junction 2014

Welcome to *Mathematical Insights in Computing!* "Problem-solving meets philosophy as we explore mind-blowing ideas from the theoretical study of complex systems: computers, minds, and beyond."

DESCRIPTION

The existence of logical paradoxes, the mathematics of infinity, the possibility of DNA computing: We will tackle these and many other curiosities as we delve into fascinating results from theoretical computer science, the mathematical exploration of what computers can and cannot do.

Come learn how you can use prime factorization to encrypt your secrets, why there are some problems that no computer can ever solve, and what makes "This sentence is false" the surprising centerpiece of an intriguing theorem. We will explore programs that can output their own source code, debate philosophical questions in artificial intelligence, and learn about the legendary unsolved problems that have left modern theoretical computer scientists still puzzling.

Expect an exciting mixture of mathematics, philosophy, and theoretical computer science, with a focus on solving and understanding problems.

Please note that we will only study *theoretical* computers. Although this class will cover information that is essential for a future computer scientist to know, it will not teach programming. On the flip side, programming experience is not required.

COURSE INFO

Web site

Lecture notes, in-class problems, homework assignments, and more will be posted to the course website, http://csvoss.scripts.mit.edu/junction/, during Junction.

Homework

Homework for each day will be posted in advance to the website and is also listed at the end of this syllabus.

Readings There will be *reading assignments* on a daily basis, assigned at the end of class and due at the beginning of the next class. Each reading will come with a very short *reading question* to assess whether you have completed the reading; the reading question is due at the beginning of the next class on a half-sheet of paper.

We will spend the first part of every day doing an in-class discussion of the previous day's reading assignment, so it is very important that you read the assignment so you can participate in discussions. Make notes of things that you would like to discuss, so that you remember them in class; sticky notes are especially helpful for this. Some of these readings will be from your course reader, and some of them will be from Sipser's *Theory of Computation*. You will receive a copy of *Theory of Computation* at the start of Junction.

Challenge problems There will be a *challenge problem* assigned every week on Monday; it is due on Thursday. There will be four challenge problems in total, one each for week 2 through week 5.

You are responsible for writing up a solution to the challenge problem by the end of the week and turning it in (on Thursday). Over the weekend I will grade them, and you'll receive feedback the following Monday.

Puzzles

Remember this part of the application?

> *You Give Me a Puzzle* Tell me a good math problem or logic puzzle that you have come across, one that you that really got you thinking, wondering, and exploring. Write as if you were describing the problem to a fellow student and include enough detail for someone else to understand and try to solve your problem.

Each of you was admitted to this class, in part, because I enjoyed your response to this question. Good taste in puzzles is one of the hallmarks of a good puzzle-solver.

But I can't keep the fun of solving the puzzles you sent in all to myself; there's a reason I asked you to make sure someone else could read and solve your problem! While there are still puzzles to spare, I will give you one of the puzzles submitted by your peers each day before you head to dinner. Solving them is 100% optional, but it is my hope that this will provide you with something entertaining to tackle during your dinnertimes, even to share with friends in other core classes.

On a logistical note, I will email you before it is your puzzle's turn to debut. If you want to make any edits to your puzzle or double-check it to make sure it is worded how you want it to be, you will be able to do so.

Final project

On the second-to-last day of class, we will run a "Firestorm". Each student will teach a 5-minute class on a topic of their choosing. This is your opportunity to teach your peers about stuff that you find cool and interesting!

You can either teach a Firestorm class on a theoretical computer science topic that you research, or pick something in which you have expertise that you wish to share with your peers; it can be academic, a hobby, or anywhere in between. Anything you want!

Please feel free to ask me for advice at any time, whether it's thinking of a class idea (that's half the battle!), researching that idea, or preparing to teach it.

Timeline

- *Thursday August 7* The tentative title of your Firestorm class, and a one-sentence description, are due. Shortly thereafter, the list of Firestorm class titles will then be sent out to the entire class.
- *Monday, August 11* Deadline for any edits to your tentative title and description.
- *Wednesday, August 13* Firestorm happens!

Prerequisites

Since you were admitted to this course, I have full confidence in your preparedness and your ability to succeed. In the application, I described the prerequisites as Algebra II and Precalculus.

The following are the concepts from these courses that I would most like you to be familiar with:

- You should have some experience with solving challenging problems and the intellectual maturity to approach new ideas and concepts.
- You should understand what a function is. You should be able to visualize or draw the difference, for example, between a graph of $y = 2x$ and a graph of $y = x^2$.
- You should understand logarithms and exponents, both when it comes to manipulating algebra and when it comes to visualization and intuition.
- It is helpful to know how to write a simple mathematical proof.

COURSE LOGISTICS

Suggestions for school supplies

- *Notetaking* You will want an empty notebook or two, or whatever you prefer as a place to take notes.

  Personally, during the school year, I like to have one notebook for each class I'm taking, plus exactly one "Life, the Universe, and Everything" notebook per semester. Then I can be as messy as I want in the "Everything" notebook: it gets filled with scratch work, project ideas, and doodles. I imagine an "Everything" notebook could be useful to you, especially for seminar classes and for our in-class problem-solving. Notebooks which contain graph paper instead of lines are generally good for the "Everything" notebook. Do whatever you prefer.

- *Handouts* Most days, I will pass out handouts containing problems for us to solve as a class. You'll want somewhere to keep these. Personally I prefer folders, since there won't be too many pages; binders will also work. Do whatever you prefer.

- *Writing* Bring pens and/or pencils.

- *Laptops, tablets, and phones* You will get much, much more out of this course if you ignore your laptop and phone the entire time. The material is difficult, and learning it requires your undivided attention; using electronics for non-academic purposes distracts *you and distracts your classmates*.

  That said, I know that there are people who prefer to use electonics for academic purposes. If you are using your laptop or tablet in class, I will assume it is for good and not for evil. If you notice your classmates are using electronics in a distracting fashion, you have my full permission to glare disapprovingly.

- *Class time* Generally, class begins at 5:05 and goes until 6:55. There will be a five-minute break in the middle of class. Half-days are weird, and are an exception to this; consult the Junction calendar for details.

- *Asking questions* is encouraged! When I am not in the middle of talking about something, feel free to just say questions or comments to the entire classroom (or raise your hand if you prefer). However, when I am in the middle of talking or lecturing, or when a fellow student is speaking, please raise your hand.

My responsibility to you

- The classroom will be a welcoming and supportive environment for every student.
- After each class day, I will post online the homework to do by the next day.
- Before the start of the next class day, I will post online the notes and in-class problems from the previous class.
- I will answer your questions about the course material, whether in-class or, if the question is beyond my knowledge, with recommendations about how to learn more.
- I will respond to email – please feel free to email me with any concerns you have about anything, whether about the course or about Junction or about the meaning of life, and I will be happy to help however I can.

Certification

To earn a certificate of distinction* in Mathematical Insights in Computing, you must:

- By the last day of class, turn in your answer to the reading question for each day's reading.
- Turn in a solution to each of the five "challenge problems".
- Turn in more than 50% of all homework assignments on time.
- Participate in in-class discussions and problem-solving sessions.
- Teach a five-minute Firestorm class to your peers on the second-to-last day of Junction.

---

* On certificates:

A student who misses no more than three days of the program and displays reasonable engagement during class is eligible to receive a certificate of program completion, which acknowledges that a student has attended Junction 2014.

In addition, a student who completes course assignments, demonstrates reasonable mastery of the course material, and displays consistent and active engagement during class (as judged by that student's core teacher) is eligible to receive a certificate of program completion with distinction. The certificate with distinction acknowledges that a student has attended Junction 2014 and has completed satisfactory, college-level work.

Core teachers will notify students of their guidelines for earning certificates with distinction in their courses. There are no quotas or "curved grading"; students will be assessed based only on their own work and not directly with respect to that of their peers.

SYLLABUS AND HOMEWORK SCHEDULE

Week 1: Models of computation

1. (1/2) Fractals and L-systems
   Reading: "The Origin of Computing" by Campbell-Kelly
2. Circuits and logic gates
   Reading: *Theory of Computation* chapter 0 by Sipser
3. Finite automata and regular languages
   Reading: *Theory of Computation* chapter 1 by Sipser
4. Pushdown automata and context-free languages
   Reading: *Theory of Computation* chapter 2 by Sipser

Week 2: Computability

5. Turing machines and the Church–Turing thesis
   Assigned: Week 2 challenge problem
   Reading: *Theory of Computation* chapter 3 by Sipser
6. The halting problem
   Reading: *Theory of Computation* chapter 4 by Sipser
7. Sizes of infinity, and Turing degrees
   Reading: "Who Can Name the Bigger Number?" by Aaronson
8. The "busy beaver" numbers
   Reading: TBD
   Due: Week 2 challenge problem

Week 3: Computing and philosophy

9. Information theory
   Assigned: Week 3 challenge problem
   Reading: "The Limits of Reason" by Chaitin
10. Logic and paradoxes
    Reading: TBD
11. Artificial intelligence
    Reading: "Is the Brain's Mind a Computer Program?" by Searle
12. Gödel's incompleteness theorems
    Reading: "Minds, Machines, and Gödel" by Lucas
    Due: Week 3 challenge problem

Week 4: The P versus NP problem

13. Asymptotic analysis and exponential growth
    Assigned: Week 4 challenge problem
    Reading: TBD
14. P versus NP
    Reading: "The Status of the P Versus NP Problem" by Fortnow
15. NP-completeness
    Reading: certain sections of "NP-Complete Problems and Physical Reality" by Aaronson
    Due: Week 4 challenge problem

Week 5: Complexity theory

16. DNA computing
    Assigned: Week 5 challenge problem
    Reading: "Computing with DNA" by Adleman
17. Prime numbers
    Reading: "Quantum-Mechanical Computers" by Lloyd
18. Quantum computing
    Reading: "The Limits of Quantum" by Aaronson
19. Symmetric-key cryptography
    Reading: TBD
    Due: Tentative Firestorm title and one-sentence description
    Due: Week 5 challenge problem

Week 6: Cryptography

20. Asymmetric-key cryptography
    Due: Edits to your Firestorm title and description
    Homework: (optional) Install PGP and acquire your very own PGP keypair
21. Zero-knowledge proofs
    Homework: Prepare your Firestorm class
22. Firestorm
    Homework: Review the course material for the quiz game
23. (1/2) Final class day and quiz game