

Notes: Where to Go From Here

Tuesday, 29 April

This Week

Tuesday, 29 April, 4:00pm (Harrison-Small Auditorium, Special Collections Library) Curtis Wong (Microsoft), *From Beethoven to Betelgeuse: 20 Years in the Quest for the Holy Grail of Interactive Storytelling*.

Wednesday, 30 April, 9:30am-10:30am (Olsson 236A) Last regularly scheduled office hours.

Wednesday, 30 April, 3:00pm (Harrison-Small Auditorium, Special Collections Library) *Recall Bill Wull* celebration. (This should be a fun and unique event for all CS students!)

Wednesday, 30 April, 6:00pm (Olsson 228E) Student group review session (organized by Sam Block).

Thursday, 1 May, 5:00pm (Olson 236D) Assistant Coaches' review session.

Saturday, 3 May, 9am-noon (Olsson 120) Final Exam.

Fall Courses

CS432: Algorithms (abhi shelat) — most of you are in majors that require this course, but if not, you should consider taking it anyway. CS432 focuses on devising and analyzing algorithms that solve hard (but not typically NP-Hard) problems. When you take cs432, keep in mind that all of your algorithm analyses assume a particular computing model (usually a Turing machine), even if you do the analysis at a higher, more abstract, level.

CS660: Theory of Computation (Gabe Robins) — this is the graduate theory course. Many of the topics covered in cs302 are also covered in cs660, and students who did well in cs302 should do well in cs660. This course usually uses the same textbook we used in cs302.

MATH540: Introduction to Quantum Computing (Slava Krushkal) — you will be ahead of most of the students in this class having a theoretical computing background, but the mathematics for quantum computing gets pretty difficult (you should be comfortable with linear algebra to take this class).

Other Courses

These courses are not offered next fall, but may be offered next Spring.

PHIL 233: Computers, Minds, Brains (Paul Humphreys) — if you want to ponder more deeply the questions about whether a human brain is a Turing machine, this is the place to do it.

CS 588: Cryptography — cryptography is perhaps the most exciting area that builds directly on theoretical results. I'm not sure if this course will be offered next year, but look out for an opportunity to take a cryptography course if you are interested in creating and breaking puzzles, and reasoning about hard problems.

Research Lunches

“Come for the food, stay for the research!”

Students who have completed cs302 should know enough to participate in the theory and security lunches, and are always welcome to attend these meetings. These groups have traditionally met every other week during the semester, but there will be a new schedule announced for the summer soon. If you're not around during the summer, you can still join the mailing list now so you will hear about meetings next year.

Theory Lunch (coordinator: Isabelle Stanton) To join the mailing list, visit

<http://www.cs.virginia.edu/mailman/listinfo/theory>

The theory lunch group meets every week or two to discuss interesting research that involves some theoretical aspects.

Security Lunch (coordinator: Karsten Nohl) To join the mailing list, visit

<http://groups.google.com/group/uva-cs-sec>

The security reading group meets every week or two to discuss interesting research in security and cryptography.

Actions

When you encounter a new problem, try to think about it generally and abstractly before diving into finding a concrete solution. Many problems you will encounter are either undecidable or NP-Hard. This doesn't mean you should give up, but that you should think carefully about what it is you are solving.

Next time you see someone playing Sudoku, explain to them that generalized Sudoku is an NP-Complete problem, and what it would mean if they could solve any size Sudoku puzzle quickly. You might also find surprising success using “Did you know you're working on an NP-Complete problem?” as a pick-up line.

If you are bored during your post-graduation or summer job, try to solve Busy Beaver puzzles instead, or try to find a way to think about the dull problems you need to solve for your job as more interesting generalizations of those problems.