Syllabus (Content)

Official name of the course:

Elementary Theory of Computation

My name for the course:

Great Ideas in Computer Science

THEME: There have been many theoretical ideas in computer science that revolutionized the entire field of computer science (not just theory). This course is about those great ideas!

All number-of-weeks are approx.

1. Regular Languages: DFA's, NFA's, Regular expressions, pumping lemma, applications to decidability. GREAT IDEA: A simple model of computation that is surprisingly powerful. (4 weeks)

2. P and NP: Turing Machines, Cook-Levin Theorem (SAT is NP-complete). Reductions. Some Complexity Theory. Ways to prove that a problem probably does not have a fast exact solution. Ways around NP-completeness. Classes above NP! Ways to prove that a problem is probably not even in NP! GREAT IDEA: Being able to prove that some problems DO NOT have fast algorithms. (4 weeks)

3. The power of randomness: Comm Complexity, Poly Identity testing, Interactive proof systems, Property testing. GREAT IDEA: Randomness is a surprisingly powerful computational aid. (1 weeks)

4. Decidable and enumerable Languages: Turing Machines and the HALTING problem. Ways to show that some problems are undecidable! Mention of Hilbert's 10th problem and Godel's theorem. GREAT IDEA: There are some problems that cannot be solved AT ALL. (3 week)