Great Ideas in Theoretical Computer Science
Advanced Course, Summer 2014

Welcome to the course "Great Ideas in Theoretical Computer Science". This course will overview major breakthroughs in theoretical computer science, and highlight their connections to other areas in computer science. In particular, we will discuss great ideas in the past 60 years that (i) provide deep understanding of the world, (ii) give computer scientists intuitions, (iii) have great influence in computer science, and (iv) create excitement.

Starting with the intriguing question about **P** vs. **NP**, we will overview our understanding to various aspects of the computation models, like time vs. space, randomized vs. deterministic computation, and finding vs. verifying solutions. We will discuss some of the fundamental techniques for designing algorithms, and some fantastic areas in theoretical computer science. This will roughly cover 5 Turing Award winners' work, and 8 Gödel Award papers. Some questions that we will answer during the course: How can we discuss a secret with friends? Is randomness necessary for designing algorithms? What are learning algorithms? Is there any connection between theoretical computer science and algebra, geometry, etc.?

Basic Information

- Lecturers: Kurt Mehlhorn, and He Sun
- Date and Time: Wednesday 2 pm - 4 pm. First meeting is on April 23.
- Place: Room 0.01, MMCI (Campus E1.7)
- Credits: 6 ECTS points
- website: `http://www.mpi-inf.mpg.de/departments/d1/teaching/ss14/gitcs/`

Schedule

We will discuss one topic per lecture. Instead of giving all proof details, we explain the intuitions and importance of these results, discuss the techniques used in their analysis, and connections to other areas in computer science. We encourage students to choose selected topics for further reading.

1. Time vs. Space, **P** vs. **NP**, and More
2. Interactive Systems
3. Expander Graphs
4. Learning
5. Streaming Algorithms
6. Public-Key Cryptography
7. Linear Programming
8. Randomness in Computation
9. Randomized Algorithms
10. Approximation Algorithms
11. Algebraic Techniques in Algorithm Design

Resources

Homepage. The course website is located at

`http://www.mpi-inf.mpg.de/departments/d1/teaching/ss14/gitcs/`.

You can find lecture notes, problem sets, and course announcements there.

References. There is no textbook for the course. We cover a diverse amount of material and no single book covers everything addressed in the course. We will put slides and lecture notes online, and these materials are good starting points to digest every lecture. We will give you some articles for each topic. Since we can only spend two hours discussing each topic or subarea, many details will be omitted. Hence we highly recommend you to read more references after class in order to obtain a deep understanding of these topics.

Similar Courses. Here are some similar courses that are offered by other universities and give a philosophical and historical perspective on computer science.

- Great Ideas in Theoretical Computer Science
  - Instructor: Prof. Scott Aaronson
  - Institution: MIT
  - Date: Spring 2008
  - `http://stellar.mit.edu/S/course/6/sp08/6.080/index.html`
- Philosophy and Theoretical Computer Science
  - Instructor: Prof. Scott Aaronson
  - Institution: MIT
  - Date: Fall 2011
  - `http://stellar.mit.edu/S/course/6/fa11/6.893/index.html`
- Great Theoretical Ideas in Computer Science
  - Instructor: Prof. John Lafferty, and Prof. Anupam Gupta
  - Institution: CMU
  - Date: Fall 2005
  - `http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15251-f05/Site/`
- Theory Gems
  - Instructor: Aleksander Mądry
  - Institution: EPFL
  - Date: Fall 2012
  - `http://thl.epfl.ch/gems/`

Grading

Homework. We will have 4 problem sets as homework, and they count for 50% of your final grade. Each problem set is given after three weeks, and every student needs to finish these problem sets individually in one week after each problem set is announced. The deadline for the homework is strict. We do not accept late submission of your homework.

Final Exam. There is a final written exam at the end of the course, and the final exam counts for 50% for your final grade. There will be questions corresponding to every lecture. However, you only need to choose half of the lectures that you find most interesting and answer the corresponding questions. You need to collect at least 50% of the homework points to be eligible to take the final exam.