Computer Science (CS) Track Core Course Description

Krishnendu Chatterjee, Vladimir Kolmogorov, Krzysztof Pietrzak

May 22, 2018

1 Course Overview

The purpose of the CS Track Core Course is to provide an overview of fundamental problems in computer science, and introduce to the students some key computational techniques that have broad applicability.

Course goals. The key goals of the course are as follows:

- Develop solid foundation about reasoning and thinking about computational problems.
- Algorithmic approaches for problems that are both fundamental in computer science and broad enough to be applicable to other disciplines of research.
- Build a community within the students through mini-projects.

Course structure. The course will be sub-divided into three segments. Each segment will focus on some key topic of computer science. Each segment will be of 4 weeks (8 lectures). The topic of each segment will be announced by the instructor in advance, and it will be ensured that it is a central topic in computer science as well as of broad relevance. The curriculum of the topics of the current segment is provided in Section 4.

Course format. Each segment will follow the same structure and will primarily consist of lectures from the instructors. The lectures will be either blackboard lectures or slide presentation.

2 Grading

The grading for the course will be done as follows:

- *Homeworks:* The homeworks will consist of solving theoretical exercises to provide hands-on-experience to tackle theoretical problems and apply the techniques learnt in the course. In addition there may be some programming assignments. The homework will be the major aspect of the grading.
- *Mini-projects:* A part of the grading (in some segment) will consist of reading research papers in groups and presenting a write-up summary.
- *Exam:* In some segment a written/oral exam may be conducted.

The weight of the homework, project, exam will depend on the instructor of the segment, and typically homeworks will be 60-100% of the grading. The instructor will present the precise breakdown at the beginning of the respective segment.

3 Prerequisites

The topics of the course will be theoretical in nature and suitable for students with background in CS undergraduate, or students with strong background in mathematics with some exposure to CS. The prerequisites for the course are as follows:

- Undergraduate CS: basic knowledge of algorithms and programming.
- Basic knowledge in probability theory.
- Basic knowledge of CS complexity theory.

An excellent book on the prerequisites is **Introduction to Algorithms** by *Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.* Some relevant chapters from the book are Chapter 6-7 (Sorting), Chapter 10 (Elementary data structures), Chapter 15-16 (Algorithmic techniques such as Dynamic programming), Section VI on Graph Algorithms, Chapter 29 (on Linear programming), and Chapter 34 (Complexity). While all of the above are not strictly prerequisites, basic knowledge about these concepts will definitely help.

4 Course Topics

4.1 Segment 1: Chatterjee

Goal. Probabilistic aspects play a crucial role in diverse disciplines, and in computer science in particular. The goal of this segment is to expose the students to the role of randomization in computation.

Topics. The possible topics for this segment are:

- Probabilistic complexity classes, their relations, and probabilistic complexity of problems such as polynomial equivalence testing.
- Decision making under uncertainty and weighted majority algorithm.
- Probabilistic inequalities (Markov inequality, Chernoff bounds) and analysis of stochastic process like Balls and Bins.
- Randomized graph algorithms (such as randomized min-cut algorithm).
- Lovasz local lemma and applications.

4.2 Segment 2: Kolmogorov

Goal. The goal of this segment is to expose the students (a) to the role of Markov chains in computation and (b) to algorithmic approaches to solve optimization problems.

Topics. The possible topics for this segment are:

- Basics of Markov chains.
- Coupling techniques and path congestion.
- Analysis of mixing times and their applications in computer science.
- Linear-programming and duality.
- Integrality of polyhedra.
- Ellipsoid method.

4.3 Segment 3: Pietrzak

Goal. The goal of this segment is to expose the students to parametrized algorithms and complexity classes.

Topics. The possible topics for this segment are:

- Algorithmic techniques: Treewidth, color coding, kernelization.
- Parameterized complexity classes and reductions.
- Lower bounds from the exponential time hypothesis.

5 Teaching Material

Any teachnig material (such as class notes) or background reading material will be provided by the instructor during the course.