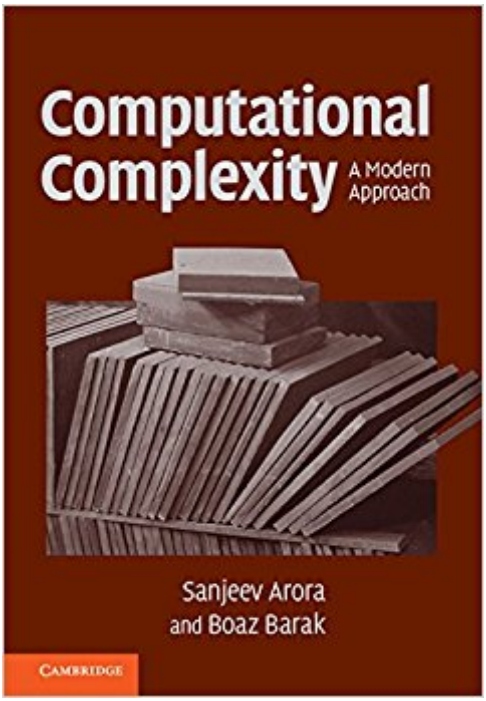
Computational Complexity: A Modern Approach



Synopsis

This beginning graduate textbook describes both recent achievements and classical results of computational complexity theory. Requiring essentially no background apart from mathematical maturity, the book can be used as a reference for self-study for anyone interested in complexity, including physicists, mathematicians, and other scientists, as well as a textbook for a variety of courses and seminars. More than 300 exercises are included with a selected hint set.

Book Information

Hardcover: 594 pages

Publisher: Cambridge University Press; 1 edition (April 20, 2009)

Language: English

ISBN-10: 0521424267

ISBN-13: 978-0521424264

Product Dimensions: 8.5 x 1.4 x 10 inches

Shipping Weight: 2.6 pounds (View shipping rates and policies)

Average Customer Review: 4.2 out of 5 starsÂ Â See all reviewsÂ (25 customer reviews)

Best Sellers Rank: #115,397 in Books (See Top 100 in Books) #64 inÂ Books > Science & Math > Mathematics > Pure Mathematics > Logic #1110 inÂ Books > Textbooks > Science & Mathematics > Mathematics #1546 inÂ Books > Textbooks > Computer Science

Customer Reviews

I almost didn’t order this book. I had visions of opening the box and pulling out some incomprehensible tome with one coma-inducing proof after another. I have a BS in math, but that was an embarrassingly long time ago, so I wasn’t sure I was up for a real test of my sanity. What a relief it was to see that this book is relatively approachable.OK, wait -- quick sanity check. This *is* a textbook about the mathematical analysis of computation; hopefully you wouldn’t even be reading this review if you couldn’t wade through a jungle of mathematical proofs, if you didn’t know some discrete math, graph theory, etc., or if you didn’t have some programming experience. There are formal notations everywhere. The subject matter of this book sets a pretty high bar, regardless of how the book is written.So, back to how the book is written. Very well! Yes, there are proofs and lemmas everywhere, but the authors do several things to focus on getting the point across without being tiresome. First, they are great about motivating what they are talking about. Why is this issue important? Why are we going to approach the problem this way? Second, they are generous with well thought out diagrams that depict what is being described in words. A few good diagrams go a

long way with me, personally. Third, in some cases they just give "hand-wavy proofs." By not getting hung up the formality of the proofs, they can choose interesting statements to prove and get the idea across in a paragraph. May I just say "Hallelujah!" -- I wish more books took this approach, concentrating more on understanding than on formality.

Vine ReviewThis is a 500 page textbook for a graduate course written by two Princeton professors who are experts in the field. As a non-expert I am not qualified to review it on technical grounds, but I was intrigued by the authors' claim to require of readers only minimal computational and mathematical background. In their introduction they state:"This book aims to describe such recent achievements of complexity theory in the context of more classical results. It is intended to serve both as a textbook and as a reference for self-study. This means it must simultaneously cater to many audiences, and it is carefully designed with that goal in mind. We assume essentially no computational background and very minimal mathematical background, which we review in Appendix A."I thought it would be an interesting experiment to see just how much I could learn about the topic just from the book itself. I did succeed in getting quite a bit more than I expected from dipping into it at various points, but in the end I was reminded of P.F. Strawson's remark that "There is no shallow end to the philosophical pool." The computational complexity pool is similarly configured.The fundamental background for this field was set up in the 1930's by logicians such as Alan Turing and Alonzo Church whose work gave the answer to what constitutes a computable function. Turing described a simple type of abstract machine and showed that such a machine could compute any computable function.With the question of what is theoretically computable decisively settled, the question turns to what is computable efficiently--that is, within an amount of time determined by the size of the input raised to some power.

Necessary Unity (Advances in Systems Theory, Complexity, and the Human Sciences) Elegant Complexity: A Study of David Foster Wallace's Infinite Jest Exploring Complexity: An Introduction Small Worlds: The Dynamics of Networks between Order and Randomness (Princeton Studies in Complexity) Computability, Complexity, and Languages, Second Edition: Fundamentals of Theoretical Computer Science (Computer Science and Scientific Computing) Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals (MIT Press) Complexity: Life at the Edge of Chaos Systems Thinking, Third Edition: Managing Chaos and Complexity: A Platform for Designing Business Architecture Statistical Mechanics: Entropy, Order Parameters and Complexity (Oxford Master Series in Physics) Quantum Transport in Mesoscopic Systems: Complexity and Statistical Fluctuations (Mesoscopic Physics and Nanotechnology)