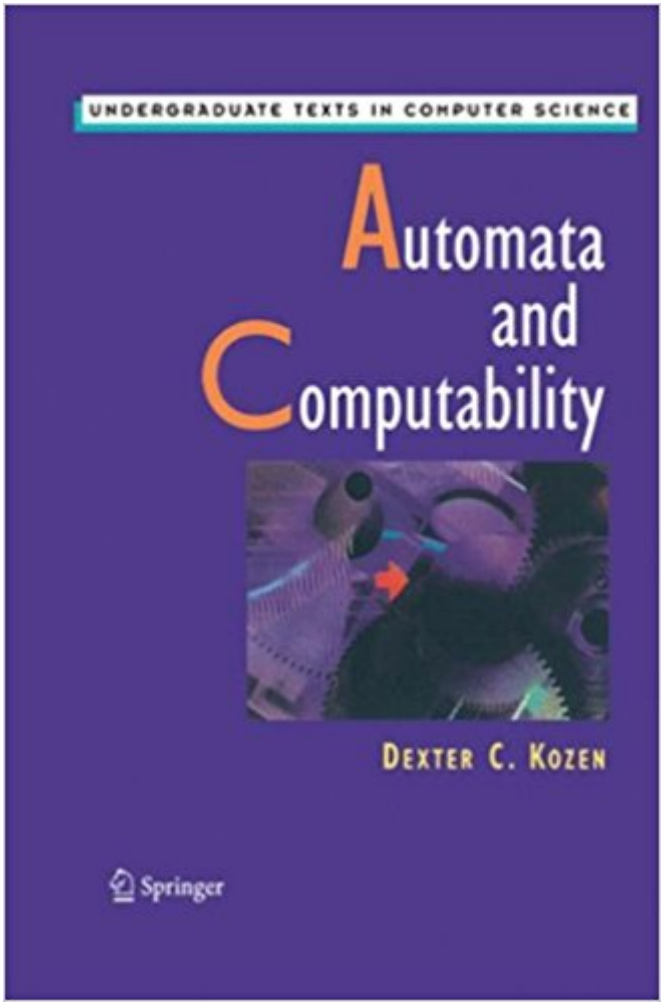
Automata And Computability (Undergraduate Texts In Computer Science)



Synopsis

This textbook provides undergraduate students with an introduction to the basic theoretical models of computability, and develops some of the model's rich and varied structure. The first part of the book is devoted to finite automata and their properties. Pushdown automata provide a broader class of models and enable the analysis of context-free languages. In the remaining chapters, Turing machines are introduced and the book culminates in analyses of effective computability, decidability, and GÃƒÂ¶del's incompleteness theorems. Students who already have some experience with elementary discrete mathematics will find this a well-paced first course, and a number of supplementary chapters introduce more advanced concepts.

Book Information

Series: Undergraduate Texts in Computer Science
Hardcover: 400 pages
Publisher: Springer (August 1997)
Language: English
ISBN-10: 0387949070
ISBN-13: 978-0387949079
Product Dimensions: 7 x 0.9 x 10 inches
Shipping Weight: 1.6 pounds (View shipping rates and policies)
Average Customer Review: 4.2 out of 5 stars 13 customer reviews
Best Sellers Rank: #84,848 in Books (See Top 100 in Books) #17 inÃ Â Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Structured Design #19 inÃ Â Books > Computers & Technology > Computer Science > AI & Machine Learning > Machine Theory #27 inÃ Â Books > Textbooks > Computer Science > Algorithms

Customer Reviews

I signed up for a grad course and needed a refresher on this stuff. I bought Sipser's book as an undergrad and have been going through it as well. The two sync up beautifully. The things I just wasn't getting from Sipser's book just kind of clicked when I read the descriptions in this book (and the other way around). If you're having trouble with the subject I highly recommend you go get both of them.

This book consists of lecture notes - in the old fashioned understanding of the word - that could be taken straight from the blackboard with a few expositions in between. The format has helped to

keep the contents to a reasonable minimum, without the depth of the "Automata Theory, Languages, and Computation" book by Hopcroft and Ullman, but it also makes it exceptionally well suited to a course at the undergraduate / lower graduate level. A good student should be able to go through the text by her/him-self and get a good understanding of a philosophically important field within Computer Science.

This textbook has been chosen as our undergraduate textbook for Foundations of Computer Science since 2000. It is a perfect book for students to review what was been taught during the lectures. The contents are divided into small sections that are easy for students to read -- unlike a big book in which a single chapter can be 100 pages long. You never get frustrated reading this book!

Reading this textbook is a pleasure.

This book simply can’t stand on it’s own as a text. A textbook should supplement the lectures of a course with further explanations and examples. This book essentially echoes exactly what you would expect in a lecture. This leaves the student hanging in the breeze when it comes to further understanding the material. Recommended as a supplement to another text on the subject but not as a sole source of learning.

This book has been a great surprise to me. Initially I thought that in about 300 pages (excluding homeworks and exercises) I could not find all I could need for an Automata, Languages and Computation course. I was wrong, definitely. The book is coincise, but also rich and precise.The material is very well chosen, and the writing stile is directly thought with students in mind. Kozen has a pluri-annual experience in teaching at Cornell University, and it seems he has developed an effective style of communication with students, that’s perfectly reflected in his books.Some important topics are present in this book and not in both Sipser and Hopcroft-Ullman. If you need (as I did) to learn about Myhill-Nerode Relations and Theorem, this book features the best account I’ve seen (the other, much shorter, reference can be found in the first editon of Hopcroft-Ullman but not in the second one !).A nice shot of the Lambda-calculus is also featured, and this too lacks in the other two books.The organization in lectures is a very good idea when studying. Lectures are carefully cut and self-contained, so that you can organize your time using this unit, and wherever you choose to stop a study session, you always stop at correct boundary of a topics.As a further (and important)

note, the notation used is very clear and elegant. As soon as you get used with it (very soon since its clarity) it becomes very stimulating. Don't understimate this value, since many books feature too-hard-to-follow notations, or no notation at all. Both of which cases are to be avoided, INMH.I have used other books for my course, starting from both the editions of the Hopcroft and Ullman, but one way or the other I found myself always with this book (and Sipser's) in my hands.

This is the textbook I used for my Honors Introduction to Theory of Computing course which was taught by Kozen. This book is very well organized, each chapter corresponds exactly to one lecture, so it's almost like a collection of lecture notes in a sense. This book (and the course it's based on) provides a very good introduction to general theoretical aspects of computing. It's divided mainly into 3 sections, each covering a third of the course. First Finite Automata, then Context Free Languages and Pushdown Automata, finally Turing machines and general computability. It covers the basics very well, sprinkled with some optional lectures on more advanced topics such as Kleene Algebra (which is a favorite of Kozen)This course mainly deals with notions and models of computation, a previous reviewer noted that it doesn't include NP-completeness. There is a reason for this, because at Cornell University, this course is the first in a sequence, the second of which covers algorithms and complexity issues. That course covers NP-completeness and all the basic algorithm techniques.For those readers in a similar situation as the previous reviewer, it's difficult to find a more simple introduction to computer theory. I thought DFAs were the easiest part of the book/course, DFAs are the simplest models of computation, you can think of counting fingers as a form of DFA. I'm confident that anyone that can count will be able to understand the explanations of DFA in this book.

Student’s Guide to Picking the Best College Major and Career) Discrete Structures, Logic, And Computability The Annotated Turing: A Guided Tour Through Alan Turing’s Historic Paper on Computability and the Turing Machine Computability: Computable Functions, Logic, and the Foundations of Mathematics Computability: Computable Functions Logic and the Foundations of Math (Wadsworth & Brooks/Cole Mathematics Series) The Data Science Design Manual (Texts in Computer Science) An Introduction to Mathematical Finance with Applications: Understanding and Building Financial Intuition (Springer Undergraduate Texts in Mathematics and Technology) Mathematics and Technology (Springer Undergraduate Texts in Mathematics and Technology) Combinatorics and Graph Theory (Springer Undergraduate Texts in Mathematics and Technology) Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra (Undergraduate Texts in Mathematics) Fourier Analysis and Its Applications (Pure and Applied Undergraduate Texts) Combinatorics and Graph Theory (Undergraduate Texts in Mathematics)