The book was found

The Little Prover (MIT Press)

The Little Prover

Daniel P. Friedman and Carl Eastlund

Drawings by Duane Bibby
Foreword by J Strother Moore
Afterword by Matthias Felleisen

PDF Adobe · DOWNLOAD EBOOK
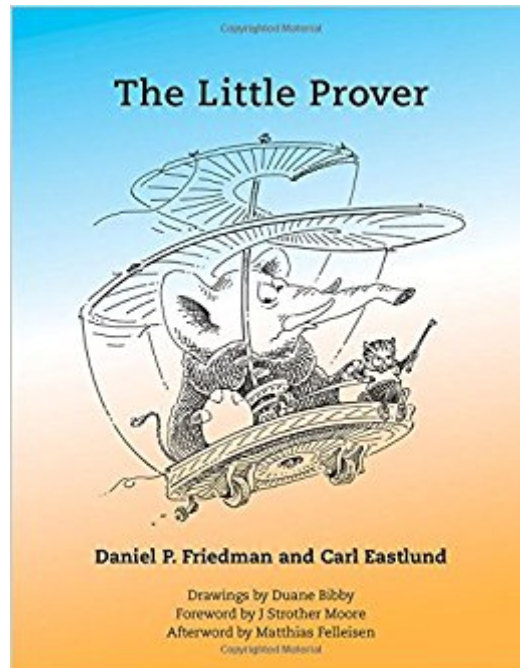
Synopsis

The Little Prover introduces inductive proofs as a way to determine facts about computer programs. It is written in an approachable, engaging style of question-and-answer, with the characteristic humor of The Little Schemer (fourth edition, MIT Press). Sometimes the best way to learn something is to sit down and do it; the book takes readers through step-by-step examples showing how to write inductive proofs. The Little Prover assumes only knowledge of recursive programs and lists (as presented in the first three chapters of The Little Schemer) and uses only a few terms beyond what novice programmers already know. The book comes with a simple proof assistant to help readers work through the book and complete solutions to every example.

Book Information

Series: MIT Press

Paperback: 248 pages

Publisher: The MIT Press; 1 edition (July 10, 2015)

Language: English

ISBN-10: 0262527952

ISBN-13: 978-0262527958

Product Dimensions: 7 x 0.6 x 9 inches

Shipping Weight: 1 pounds (View shipping rates and policies)

Average Customer Review: 4.6 out of 5 stars   6 customer reviews

Best Sellers Rank: #160,972 in Books (See Top 100 in Books)   #57 inÃ Â Books > Computers & Technology > Mobile Phones, Tablets & E-Readers > Programming & App Development   #81 inÃ Â Books > Science & Math > Mathematics > Pure Mathematics > Logic   #2177 inÃ Â Books > Computers & Technology > Programming

Customer Reviews

Computational theorem proving is so useful, effective, and important that its advocates present it in economic terms: about preventing costly errors in software and protocols. What gets lost is just how much fun it can be. Friedman and Eastlund, two jolly characters, eschew talk of bugs and bombs, and strip it to its essence as only a Little book can. Want proof? Look inside! (Shriram Krishnamurthi, Professor of Computer Science, Brown University)What can you learn about a program without actually running it? What can you know by reading the code? In the grand tradition of The Little LISPer and The Little Schemer, Dan Friedman and Carl Eastlund elegantly present small programs that contain big ideas, but now also show how to prove claims about their behavior.

The question-and-answer format makes it surprisingly easy to learn and master inductive proof techniques -- it's like eating peanuts! Don't overlook the concise and elegant code for the J-Bob proof assistant itself, tucked in the back. The Little Prover is a marvelous introduction to the program proof techniques used in such tools as ACL2, Isabelle, and Coq. (Guy L. Steele Jr., Software Architect, Oracle Labs, coinventor of the Scheme language, and coauthor of Common LISP: The Language)Friedman and Eastlund's The Little Prover is a gentle introduction to the nuts and bolts of formal proofs about programs. Following on from The Little Schemer, it is an excellent guide for both thoughtful functional programmers wondering what it really means to know that a program is correct and do-it-yourselfers who want a taste of how proof assistants like ACL2 do their work. Bring your sense of humor and your thinking cap! (Benjamin C. Pierce, Henry Salvatori Professor of Computer and Information Science, University of Pennsylvania)

Daniel P. Friedman is Professor of Computer Science at Indiana University and coauthor of The Little Schemer (fourth edition), The Reasoned Schemer, The Seasoned Schemer, and Essentials of Programming Languages (third edition), all published by the MIT Press. Carl Eastlund is a software engineer at Jane Street Capital in New York City.

Okay book.

The Little Prover is a great introduction to computation logic. I really enjoy it because of its format, its pacing, and its ability to explain core concepts in mathematics and computer programming.I like The Little Prover's question-and-answer format. It encourages me to write and understand proofs for myself, because the format explains the steps when I get stuck, and it helps me understand the significance of the steps when I succeed.The Little Prover was very well paced. More complicated proofs are introduced at a rate that keeps them challenging and interesting but never impossible.The way The Little Prover covers computational logic means I am able to relate it to my A-Level standard knowledge of proof in mathematics (A-Levels are the English equivalent of American high school). The book doesn't just show you how to use a computer program to do certain things, but relates what you're doing to concepts like inductive proof and recursion.The format of The Little Prover is very similar to the other books in the Little series such as The Little Schemer. I highly recommend those too. The Little Prover also contains many ideas found in How to Design Programs (HtDP). For example the programs in The Little Prover clearly all use design recipes from HtDP and concepts such as natural recursion are mentioned in both HtDP and The

Little Prover.

Disclaimer: I was a tech reviewer for this book.Ever since computer programming was invented, bugs have been the bane of every programmer's existence. But does it have to be that way? What if we could prove our programs correct? Computer-aided theorem proving applied to proving programs correct has a long and rich history, and its use has accelerated greatly over the past decade. I believe that this will be a foundational subject for the next generation of programmers, as the type systems for existing languages reach their limits and are replaced by richer type systems which require theorem provers as an integral part of the programming process. (Languages/environments like Agda, Coq, and Idris are exploring this space already.) The Little Prover is a very accessible introduction to theorem proving; if you understand the first few chapters of The Little Schemer (and are willing to work!) you will be able to follow The Little Prover. In addition, the material is fascinating in its own right and will be worthwhile reading as brain food for programmers looking for a new challenge (much like the other "Little" books that Friedman et al have written). However, don't confuse "accessible" with "easy". Compared to a lot of modern proof assistant software like Coq, the proofs are quite manual and require a lot of user guidance. This is actually advantageous in a book like this, because you see every step and never have to wonder how we got from point A to point B (plus the authors are doing all the hard work; you just have to follow along!). The authors also provide a downloadable proof assistant which is extremely helpful when the proofs get sufficiently complex, and they list all the proofs in their final form in the back of the book.I suspect that for most readers, the biggest conceptual hurdle will not be the material presented but just wrapping your head around the very notion that programs are something that can be reasoned about rigorously and that it is in fact possible to prove things about them. This is not the way we have been taught to think about programming, and it's not the way that programming is currently done. But it is the way that programming will have to be done in the future if we ever want to dig ourselves out of the swamp of perpetually-buggy software, and this book is an excellent starting point for learning about theorem proving in software. If you make it all the way to the end, I think you will be "proof-infected" and this will not be the last book on theorem proving that you read!

This is an excellent introduction to assisted theorem proving. For a longer-form review of this book, see the post on my site about it located at mattjquinn.com entitled "Assisted Theorem Proving with J-Bob," dated May 2016. I cover my motivations and provide a few code extracts to show my experience with working through the book's chapters and exercises. If you're interested in learning

about Coq, TLA+, and other theorem provers, you'll have to do a lot more work above and beyond working through this book, but it is an excellent resource for getting you into the right frame of mind.

The Little Prover was published just a few months ago and when I learned it's getting published I pre-ordered a bunch of copies not just for myself but for all my friends. I was that excited!The Little Prover teaches the readers how to determine facts about recursive functions using induction. The book starts with programming concepts such as recursive functions and lists, and leads the reader along the shortest path to inductive proofs. It assumes knowledge of neither logic nor mathematics beyond arithmetic. Just like all other books in the series, it's written as a dialog between the authors and you in a super fun style with a lot of jokes and insider references. This book will make you think from page one and it will stretch your mind quite a bit.You'll learn about axioms, theorems, what it means for something to be true in computing, conjectures, claims, counterexamples, total functions, partial functions, conjunctions, induction, natural recursion, and more.While working through this book, you'll need j-bob, the proof assistant:[...](Google for j-bob if this link gets removed.)You'll truly appreciate this book only after having read the other books in the series (The Little Schemer, The Seasoned Schemer, The Reasoned Schemer, The Little MLer). Get those first, and then read this book. And then read all the books again for full effect.I've placed this book #24 in my Top 100 Programming, Computer and Science books list:[...](If this link gets removed google for >>catonmat top 100 programming computer science books

I haven't read it but I'm sure it's good. Friedman's other books are.

Download to continue reading...

The Little Prover (MIT Press) Little Bear Audio CD Collection: Little Bear, Father Bear Comes Home, Little Bear's Friend, Little Bear's Visit, and A Kiss for Little Bear A Little Books Boxed Set Featuring Little Pea, Little Hoot, Little Oink My Little Bible Box: Little Words of Wisdom from the Bible; Little Blessings from the Bible; Little Psalms from the Bible Lerne FranzÃƒÂ¶sisch mit Mimi: Mimi und die Ausstellung. Ein Bilderbuch auf FranzÃƒÂ¶sisch/Deutsch mit Vokabeln (Mimi de-fr 2) (German Edition) Lies Mit Mir! Intermediate Reader 2 (Komm Mit) Komm mit!: Beginner Reader Lies mit mir Level 1 Komm mit!: Advanced Reader Lies mit mir Level 3 Komm mit! German: PRAC & ACT BK KOMM MIT! HOLT GERMAN 2 95 Level 2 Westafrika mit dem Fahrrad: Mit dem Rad durch Marokko, Mauretanien, Senegal, Mali, Burkina Faso und Togo (German Edition) 101 Things I Learned in Architecture School (MIT Press) Collage City (MIT Press) Sentient City: Ubiquitous Computing, Architecture, and the Future of Urban Space (MIT Press) You Have to Pay for the

Public Life: Selected Essays of Charles W. Moore (MIT Press) Architect?: A Candid Guide to the Profession (MIT Press) Down Detour Road: An Architect in Search of Practice (MIT Press) Architecture Depends (MIT Press) Sustainability (The MIT Press Essential Knowledge series) Planning Ideas That Matter: Livability, Territoriality, Governance, and Reflective Practice (MIT Press) The Strip: Las Vegas and the Architecture of the American Dream (MIT Press)