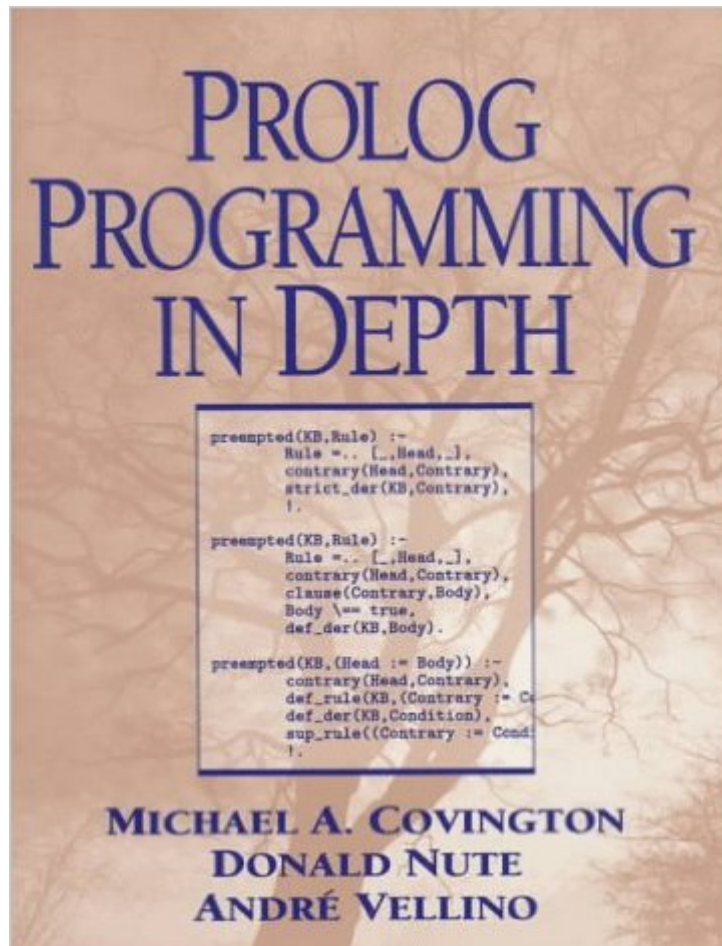The book was found

Prolog Programming In Depth

Synopsis

This book covers the Prolog programming language thoroughly with an emphasis on building practical application software, not just theory. Working through this book, readers build several types of expert systems, as well as natural language processing software and utilities to read foreign file formats. This is the first book to cover ISO Standard Prolog, but the programs are compatible with earlier dialects of the language. Program files are available by FTP from The University of Georgia.

Book Information

Paperback: 516 pages

Publisher: Prentice Hall; 1 edition (May 31, 1996)

Language: English

ISBN-10: 013138645X

ISBN-13: 978-0131386457

Product Dimensions: 7 x 0.8 x 9.2 inches

Shipping Weight: 1.9 pounds

Average Customer Review: 4.7 out of 5 stars See all reviews (7 customer reviews)

Best Sellers Rank: #1,924,128 in Books (See Top 100 in Books)   #13 in Books > Computers & Technology > Programming > Languages & Tools > Prolog   #5837 in Books > Textbooks > Computer Science > Programming Languages   #13843 in Books > Computers & Technology > Software

Customer Reviews

One of the most underrated schools of thought in AI and NLP is the the school which was lead by Michael Covington and Donald Nute at University of Georgia. It is the opinion of this reviewer that this school produced, in its papers, thesis of grad students, and books, the most impressive results in discourse representation theory produced in the western hemisphere, and are comparable with those produced anywhere else in the world.This book--a product of that school--is such an unexpected little gem of a prolog book. I wish I would have stumbled across it much earlier in my quest for prolog mastery, because it would have saved me at least 18 months of struggling to learn prolog tools and techniques--techniques which were not only scattered among dozens of books, but were not nearly as clearly described.One area which this book covers which virtually no other prolog book covers is nondefault reasoning. Really a remarkable discussion. Unfortunately, I think that the techniques of nondefault reasoning which are covered in this book have been superceeded by

newer techniques, such as probabilistic horn abduction and Baysian methods. But just because these techniques are of historical interest doesn't mean they are of no interest, and the treatment they get here is clear, honest, and more complete than anywhere else I've looked.

Lots of practical tips early in the book, particularly on I/O, get the reader off to a fast start in Prolog. This reviewer felt lost in Prolog until he read this text. The breadth of examples is impressive. The only flaw on this book is the author's tendency to fight some ideological battles that the reader does not care about. Their tone is unnecessarily hostile, particularly in the introduction to Chapter 4. But that's only one paragraph in an otherwise wonderful book. It is a great book for getting started, and getting a feel for Prolog, but it is no substitute for a thorough text that includes some theory. Theory is not a bad word. A working understanding of how the logic interpreter works is important for debugging. In my opinion, Chapter 3 is denser than it appears, and should be studied carefully. cf. the discussion of append. This reviewer has done the exercises using SWI prolog with only minor adaptations. All told, a solid introduction. A good book to read before (but not instead of) a more theoretical introduction such as The Art of Prolog. Even though freeware prologs exist on the internet, the appeal of this "practical" book would be greatly increased if the authors arranged to have a CD of some freeware prologs included with the text.

This book has been an excellent material for me to develop my Logic Programming skills. That is because it provides not only theory of Prolog programming but also very useful artificial intelligence applications and expert system shells even with uncertainty. Another useful feature I liked much is; the book makes comparisons (where needed) between how to do the same thing in Prolog and in a conventional programming language, Pascal.

I have been in the field of software development and architecture for quite a while now. While I value the rigor and comprehensive nature of academic texts on specific topics in this field, I frequently find them to be overly abstract to easily apply to a particular problem domain. This text provides an excellent, pragmatic, and easy to apply introduction to both the Prolog semantic model and language structure, as well as its differences from traditional procedural or assignment-based programming languages and metaphors. I also have "The Art of Prolog" and several other books on the topic, but this is the book that I find myself repeatedly using as a reference on how to most clearly and easily represent and solve a particular problem using this extremely flexible and expressive language. Kudos to the authors for an excellent presentation on the subject!

This is an excellent reference and great examples in a time that it is difficult to find anything to help with Prolog.Great for the person with no experience in Prolog that has inherited an application written in Prolog.

Extremely solid introduction to Prolog. Does not leave you wondering what is going onas I have seen MANY other books on Prolog do. Wish I had read this as my first Prolog book.

The book is well organized with fluent exposition especially, in comparing the traditional and logic programming. The efford spend to create a bridge between these programming paradigmas is interesting and remarkable. It contains the examples that were prepared intelligently about how to think writing a program and AI examples. It also contains some hints about elegant programming techniques in prolog and for those who want to write a prolog interpreter. Moreover, how a programmer makes mistakes in programming were determined. In this mean, it has a spectrum from novice to expert. Reader can find differencies between different prolog implementation and prolog standards and can find good explanation about some special topics such as cut operator, recursive definition, file usage, reading data in foreign formats, tree structures and operator definitions special to user.It would have been better to write a whole chapter which includes how to write a prolog interpreter. There is no knowledge about prolog can interact with what languages and how it is done this. For example, how we can use C in prolog and vice versa.Mehmet Fatih Hocaoglu

Download to continue reading...

Prolog Programming Success in a Day: Beginners Guide to Fast, Easy and Efficient Learning of Prolog Programming Prolog Programming Success in a Day: Beginner's Guide to Fast, Easy, and Efficient Learning of Prolog Programming Prolog Programming in Depth Prolog: Reproductive Endocrinology and Infertility / Critique Book / Assessment Book (ACOG, PROLOG) The Art of Prolog: Advanced Programming Techniques (Mit Press Series in Logic Programming) The Art of Prolog, Second Edition: Advanced Programming Techniques (Logic Programming) Prolog ++: The Power of Object-Oriented and Logic Programming (International Series in Logic Programming) The Art of Prolog: Programming Examples - Macintosh (Logic Programming) The Art of Prolog: Programming Examples - PC (Logic Programming) Java: The Simple Guide to Learn Java Programming In No Time (Programming,Database, Java for dummies, coding books, java programming) (HTML,Javascript,Programming,Developers,Coding,CSS,PHP) (Volume 2) Prolog Programming for Artificial Intelligence Logic, Programming and Prolog Prolog Programming

Success In A Day Techniques of Prolog Programming with Implementation of Logical Negation and Quantified Goals Logic Programming with Prolog Prolog & Expert Systems Programming An Introduction to Logic Programming Through Prolog (Prentice Hall International Series in Computer Science) Programming in Prolog Micro-Prolog: Programming in Logic PROLOG Programming