LISP, which stands for LISt Processing, was developed in the late 1950s by John McCarthy as a language for manipulating symbols. This book presents the Common LISP Language, a version of LISP, and details a range of applications for it, including data structures, computer systems, and compiler design. This book presents a tutorial introduction to Common LISP, covering lists and trees, recursion, local vs. global reference, characters and strings, streams, LAMDA and LABELS, control, debugging, macros, structures, classes and objects, vectors and arrays, and efficiency and compilation. For programmers interested in a language with simple syntax, extensive adaptability and advanced memory management.

Book Information

Paperback: 800 pages
Publisher: Prentice Hall (August 9, 1997)
Language: English
ISBN-10: 0136059406
Product Dimensions: 6.9 x 1.7 x 9 inches
Shipping Weight: 3 pounds (View shipping rates and policies)
Average Customer Review: 3.5 out of 5 stars See all reviews (6 customer reviews)
Best Sellers Rank: #1,602,064 in Books (See Top 100 in Books) #92 in Books > Computers & Technology > Programming > Languages & Tools > Lisp #1835 in Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Object-Oriented Design #5133 in Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

I'd bought a copy of Keene, because I'd heard it was good, but ended up being a bit disappointed with it. So when I saw Slade's book, I wondered if this was the CLOS book I'd been looking for. Short answer: no, it's worse. It's not even clear what the book is trying to teach: programming, or Lisp, or object-oriented programming, or CLOS. He seems to be trying to do all of the above, but due to the huge scope, being unable to go deep into anything. He starts chapters with irrelevant quotes, which works if you're Knuth, but he's no Knuth. He jumps around a lot, and the order is bizarre: several important concepts are held off until relatively late. Other big concepts are mentioned only in passing, which an experienced Lisper will understand, but then, if you already know what he means, why would you read this book? No surprise that I'd not heard of this book: in a
field of classics, there’s just no point. If you want to learn programming, read SICP. If you want to learn Common Lisp (with lots of sample code), read PCL, or PAIP.

*Download to continue reading...*