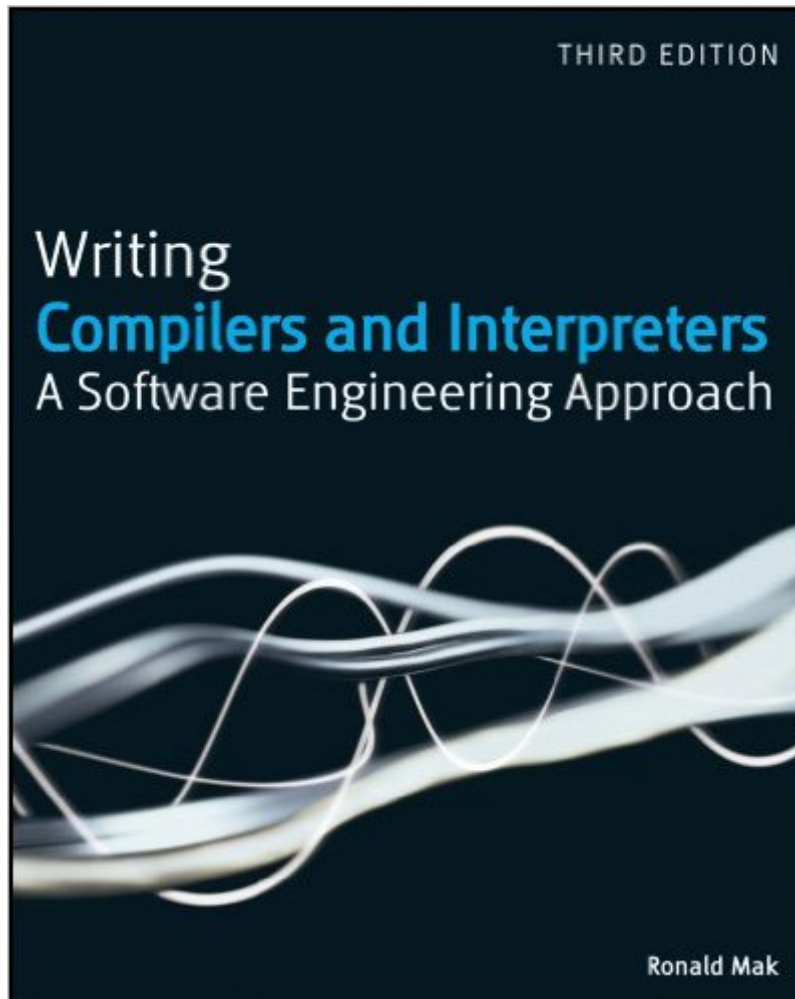# Writing Compilers And Interpreters: A Software Engineering Approach

Synopsis

Long-awaited revision to a unique guide that covers both compilers and interpreters Revised, updated, and now focusing on Java instead of C++, this long-awaited, latest edition of this popular book teaches programmers and software engineering students how to write compilers and interpreters using Java. You?ll write compilers and interpreters as case studies, generating general assembly code for a Java Virtual Machine that takes advantage of the Java Collections Framework to shorten and simplify the code. In addition, coverage includes Java Collections Framework, UML modeling, object-oriented programming with design patterns, working with XML intermediate code, and more.

Book Information

File Size: 12537 KB

Print Length: 864 pages

Publisher: Wiley; 3 edition (March 10, 2011)

Publication Date: March 10, 2011

Sold by:Â Digital Services LLC

Language: English

ASIN: B004S82O40

Text-to-Speech: Enabled

X-Ray: Not Enabled

Word Wise: Not Enabled

Lending: Enabled

Enhanced Typesetting: Not Enabled

Best Sellers Rank: #734,674 Paid in Kindle Store (See Top 100 Paid in Kindle Store)   #82 inÂ Books > Computers & Technology > Programming > Languages & Tools > Compiler Design #237 inÂ Books > Computers & Technology > Programming > Languages & Tools > Compilers #12792 inÂ Kindle Store > Kindle eBooks > Computers & Technology

Customer Reviews

The book of Ronald Mak is simply excellent.It develops very clearly the techniques to build a parser,an interpreter, a source-level debugger and a compiler for theJava Virtual Machine, all directed for the standard Pascal language.The design of Ronald Mak is very modular build upon robustsoftware engineering methodologies and the developed modules can be easilycustomized (e.g. by extending the Pascal language) and can be readilyutilized in other similar application

domains (e.g. implementing specialpurpose languages by replacing only the Pascal scanner and parser).The parser is a top-down one, easily understandable.Mak produces an intermediate code representation of the program, that is Pascalindependent, and thus both the presented source-level debugger, interpreter and JVMcompiler are kept independent of the Pascal language,since they operate on the AbstractSyntax Tree based intermediate code representation.The Java code of the book is very elegant and understandable.The design of the interactive source level debugger and of the compiler targeted at theJava Virtual Machine are also of outstanding quality.As the author also declares, the book is more for the one who wants toimplement compilers than for the compiler theorist. For the student or engineer whowants to realize interpreters/compilers, the book is magnificent.

This book was just what I needed to refresh my fading memory of compiler design skills. And this one is cross-platform, as the code is written in Java.The book goes step by step in developing a full Pascal interpreter and then compiler. Highly recommended!!

Not the best book on the subject. You can, however, learn something if you are willing to go through this book. I say "willing to go through this book" because the writing style is dull; reading the book becomes boring at times. To wrap it up: the book (along with the full source code online) contains valuable information, but the way this information is presented to the reader is not as good as it should.

It is filled with actual working code, so that a practical Software engineer can see what is happening as the chapters incrementally add to what has been developed in earlier chapters.There is not a large emphasis on theory, and while necessary at some point, it allows for a more grounded approach to learning from a working code perspective. Allowing theory to be learned after the fact, once code has been seen in action.Well written and well structured.

Writing Compilers and Interpreters: A Software Engineering Approach Writing Compilers and Interpreters Writing : Novel Writing Mastery, Proven And Simple Techniques To Outline-, Structure- And Write A Successful Novel ! - novel writing, writing fiction, writing skills - Algorithms, Languages, Automata, And Compilers: A Practical Approach Optimizing Compilers for Modern Architectures: A Dependence-based Approach Youdunit Whodunit!: How To Write Mystery, Thriller and Suspense Books (Writing Skills, Writing Fiction, Writing Instruction, Writing a Book) How to Write a Software

Patent Application: Your Guide to Quickly Writing Your US Software Patent Application Swift: Programming, Master's Handbook: A TRUE Beginner's Guide! Problem Solving, Code, Data Science, Data Structures & Algorithms (Code like a PRO in ... mining, software, software engineering,) The Good Guide: A Sourcebook for Interpreters, Docents, and Tour Guides Reading Between the Signs: Intercultural Communication for Sign Language Interpreters 3rd Edition Reading Between the Signs: Intercultural Communication for Sign Language Interpreters 2nd Edition The Bilingual Courtroom: Court Interpreters in the Judicial Process (With a New Chapter) Dictation: Dictate Your Writing - Write Over 1,000,000 Words A Year Without Breaking A Sweat! (Writing Habits, Write Faster, Productivity, Speech Recognition Software, Dragon Naturally Speaking) Compilers: Principles, Techniques, and Tools Compilers: Principles, Techniques, and Tools (2nd Edition) Compilers: Principles and Practice High-Performance Compilers for Parallel Computing Engineering Software as a Service: An Agile Approach Using Cloud Computing Engineering Software as a Service: An Agile Approach Using Cloud Computing + $10 AWS Credit A Software Engineering Approach to LabVIEW