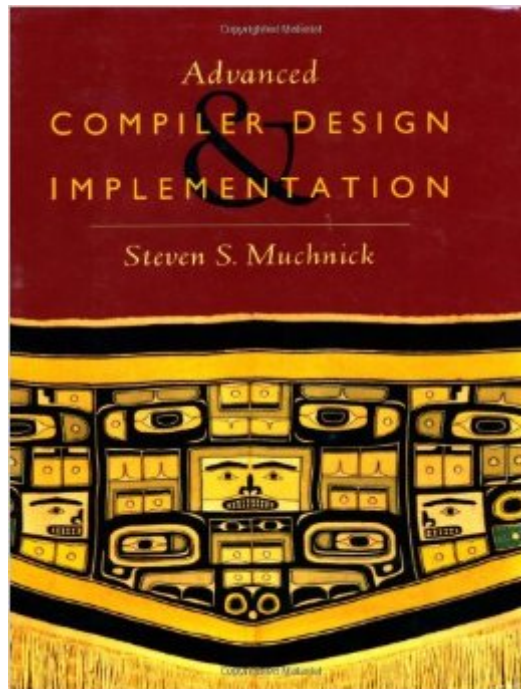


The book was found

# Advanced Compiler Design And Implementation



## Synopsis

From the Foreword by Susan L. Graham: This book takes on the challenges of contemporary languages and architectures, and prepares the reader for the new compiling problems that will inevitably arise in the future. The definitive book on advanced compiler design This comprehensive, up-to-date work examines advanced issues in the design and implementation of compilers for modern processors. Written for professionals and graduate students, the book guides readers in designing and implementing efficient structures for highly optimizing compilers for real-world languages. Covering advanced issues in fundamental areas of compiler design, this book discusses a wide array of possible code optimizations, determining the relative importance of optimizations, and selecting the most effective methods of implementation.\* Lays the foundation for understanding the major issues of advanced compiler design\* Treats optimization in-depth\* Uses four case studies of commercial compiling suites to illustrate different approaches to compiler structure, intermediate-code design, and optimization\* these include Sun Microsystems's compiler for SPARC, IBM's for POWER and PowerPC, DEC's for Alpha, and Intel's for Pentium and related processors\* Presents numerous clearly defined algorithms based on actual cases\* Introduces Informal Compiler Algorithm Notation (ICAN), a language devised by the author to communicate algorithms effectively to people

## Book Information

Hardcover: 856 pages

Publisher: Morgan Kaufmann; 1 edition (August 15, 1997)

Language: English

ISBN-10: 1558603204

ISBN-13: 978-1558603202

Product Dimensions: 9.6 x 7.6 x 1.7 inches

Shipping Weight: 3.7 pounds

Average Customer Review: 4.3 out of 5 stars Â Â See all reviews Â (19 customer reviews)

Best Sellers Rank: #945,411 in Books (See Top 100 in Books) #59 in Â Books > Computers & Technology > Programming > Languages & Tools > Compiler Design #159 in Â Books > Computers & Technology > Programming > Languages & Tools > Compilers #466 in Â Books > Computers & Technology > Hardware & DIY > Design & Architecture

## Customer Reviews

This book is the comprehensive text for anyone working on an optimizing compiler for uniprocessor

systems. It gives good detail on all major approaches and is up-to-date on important techniques like SSA form and partial redundancy information. As someone working directly in the field, it's saved me the effort of hunting up original research papers in many areas. One drawback for this book as a practical tool: the pseudocode used to illustrate examples is often pretty far from being suitable for real implementations. A warning: this is not an introductory book, and people who want to learn about the basics of building a compiler should look elsewhere; perhaps Andrew Appel's "Modern Compilers" series. Muchnick's book is for people who want to write compilers which generate high-performance code.

I have been working on language processors, interpreters and compilers for almost twenty years. I try to order all the books that have something unique to say about compiler design and implementation. This is one of the best books I have seen on advanced compiler design. I have owned it since it was first published. Going back and rereading it I am reminded of what an excellent book it is, which is what motivated this review. Advanced compiler design deals with various forms of optimization, including local, global and loop optimization. This is a complex topic with thirty years of research behind it (it is interesting to note that the late Gary Kildall, of CP/M fame, did some early work on optimization in the early 1970s). No single book can provide complete coverage of all optimization issues. However, this book, along with Allen and Kennedy's equally excellent "Optimizing Compilers for Modern Architectures" covers almost everything you need to know. One of the problems with the academic literature on compiler optimization is that it can be unnecessarily obscure. Muchnick writes clearly, with the implementer in mind. He provides a wide range of techniques, allowing the implementer to choose the correct one for a given compiler. This approach is both useful and necessary: there is no single method for building a compiler, given the range of languages and design objectives. Muchnick covers everything you need to know about local and global scalar optimization, including scalar optimization in loops and optimization for modern processor architecture. The only thing missing is an in-depth coverage of loop dependence and optimization techniques, which is provided by Allen and Kennedy. If you are working on the design, implementation or extension of a modern compiler, this book should be part of your library.

This is a very good intermediate/advanced text for codegen & optimization. Having spent nearly 15 years working on compiler development, with most of it spent on codegen & optimization in compiler backends, I was very happy to see this book published ten years ago when I was relatively new to

the field and hungry for more information. The good: Coverage of many important topics, and a better discussion of the phase ordering issues than I've seen in other texts. The bad: The writing style is okay, but not great. Little insight is typically given for the problems discussed. A very annoying aspect of the book is that the author sometimes goes into great detail about one technique, and then later mentions another (often superior, IMO) way to solve the problem at hand, without going into much detail. As a working compiler writer, I would have preferred it be the other way around. The very very bad: The ICAN notation used throughout the book is verbose, unintuitive, and ultimately takes away from the presentation. Compare the complicated detail-filled ICAN listings in this book to the simple, elegant algorithms in the Dragon Book or Cooper/Torczon's text.

[Download to continue reading...](#)

Advanced Compiler Design and Implementation A Small C Compiler: Language, Usage, Theory, and Design The Compiler Design Handbook: Optimizations and Machine Code Generation, Second Edition Design of an Optimizing Compiler (Programming Languages) MCTS Self-Paced Training Kit (Exam 70-432): Microsoft® SQL Server® 2008 - Implementation and Maintenance: Microsoft SQL Server 2008--Implementation and Maintenance (Microsoft Press Training Kit) Asap Implementation at the Speed of Business: Implementation at the Speed of Business Advanced MPLS Design and Implementation (CCIE Professional Development) C Programming for Microcontrollers Featuring ATMEL's AVR Butterfly and the free WinAVR Compiler Engineering a Compiler: Vax-11 Code Generation and Optimization GNAT User's Guide - GNAT The GNU Ada Compiler: Manual For Gcc Version 4.3.3 GNAT Reference Manual - GNAT The GNU Ada Compiler: Manual For Gcc Version 4.3.3 Compiler Construction (International Computer Science Series) Operating Systems Design and Implementat: Design and Implementation Advanced Software Testing - Vol. 3, 2nd Edition: Guide to the ISTQB Advanced Certification as an Advanced Technical Test Analyst Advanced Software Testing - Vol. 2, 2nd Edition: Guide to the ISTQB Advanced Certification as an Advanced Test Manager Distributed Systems - Architecture and Implementation: An Advanced Course (Springer Study Edition) Design and Implementation of a Three-Tiered Web-Based Inventory Ordering and Tracking System Prototype Using CORBA and Java Database Systems: Design, Implementation, and Management (with Premium WebSite Printed Access Card and Essential Textbook Resources Printed Access Card) CommVault Storage Policies: An in depth guide to storage policy design and implementation The Design and Implementation of the 4.3 BSD UNIX Operating System

[Dmca](#)