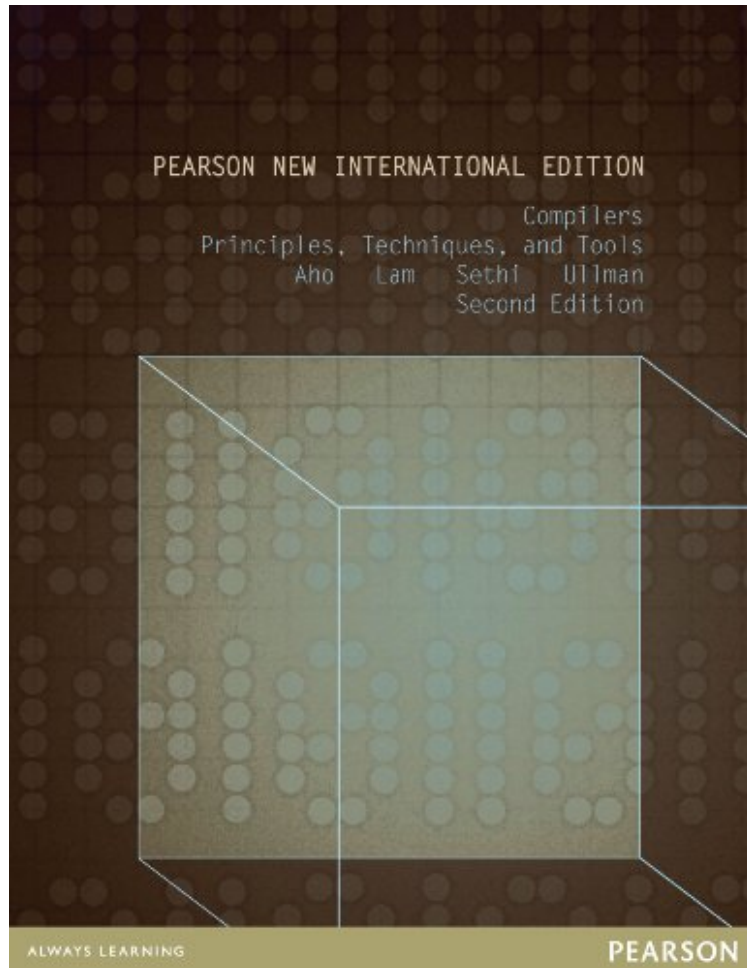


Compilers: Pearson New International Edition: Principles, Techniques, and Tools

Von A.V. Aho, Monica S Lam, R. Sethi, Jeffrey D. Ullman
DOC | *audiobook | ebooks | Download PDF | ePub



Produktinformation -Verkaufsrank: #102916 in eBooksVerffentlicht am: 2013-08-29Erscheinungsdatum: 2013-08-29File Name: B00IZ0FXX6 | File size: 41.Mb

Von A.V. Aho, Monica S Lam, R. Sethi, Jeffrey D. Ullman : Compilers: Pearson New International Edition: Principles, Techniques, and Tools before purchasing it in order to gage whether or not it would be worth my time, and all praised Compilers: Pearson New International Edition: Principles, Techniques, and Tools:

KundenrezensionenHilfreichste Kundenrezensionen6 von 8 Kunden fanden die folgende Rezension hilfreich. The classical dragon bookVon Daniel PlatteI just recently bought the new edition of the "dragon book" and it seems to be a really high quality book (being worth the spent money). Although I studied electrical engineering and did only have a brief understanding of compilers, the book provides a very good opportunity to learn more about the (partially quite theoretical) backgrounds of compilers. The writing style is very good to read. Furthermore, I like the mixture of theory and practical examples as provided by the book very much.15 von 25 Kunden fanden die folgende Rezension hilfreich. Deep disappointmentVon Vadim KataevThis book doesn't help to get systematic, general, and consistent

knowledge about how to design a new programming language or how to develop a new compiler. Authors tend to explain even the most simple things using a lot of words, making an illusion it's a complicated stuff. After reading first chapters of this book, I have decided to stop and to search for another sources of explanation, and I found one very helpful, useful book "Engineering a Compiler" by Keith D. Cooper and Linda Torcson. Some passages below should give you a feeling of how different they are. On Context-Free Grammars¹) from "Compilers. Principles, Techniques, and Tools":[...]In this section, we introduce a notation - the "context-free grammar", or "grammar" for short - that is used to specify the syntax of a language. Grammars will be used throughout this book to organize compiler front ends.[...]2) from "Engineering a Compiler":[...]The traditional notation for expressing syntax is a grammar - a collection of rules that define, mathematically, when a string of symbols is actually a sentence in the language. Computer scientists usually describe the syntactic structure of a language using an abstraction called a context-free grammar (cfg). [...]

Kurzbeschreibung Compilers: Principles, Techniques and Tools, known to professors, students, and developers worldwide as the "Dragon Book," is available in a new edition. Every chapter has been completely revised to reflect developments in software engineering, programming languages, and computer architecture that have occurred since 1986, when the last edition published. The authors, recognizing that few readers will ever go on to construct a compiler, retain their focus on the broader set of problems faced in software design and software development. New chapters include: Chapter 10 Instruction-Level Parallelism Chapter 11 Optimizing for Parallelism and Locality Chapter 12 Interprocedural Analysis Kurzbeschreibung Compilers: Principles, Techniques and Tools, known to professors, students, and developers worldwide as the "Dragon Book," is available in a new edition. Every chapter has been completely revised to reflect developments in software engineering, programming languages, and computer architecture that have occurred since 1986, when the last edition published. The authors, recognizing that few readers will ever go on to construct a compiler, retain their focus on the broader set of problems faced in software design and software development. New chapters include: Chapter 10 Instruction-Level Parallelism Chapter 11 Optimizing for Parallelism and Locality Chapter 12 Interprocedural Analysis Synopsis Compilers: Principles, Techniques and Tools, known to professors, students, and developers worldwide as the "Dragon Book," is available in a new edition. Every chapter has been completely revised to reflect developments in software engineering, programming languages, and computer architecture that have occurred since 1986, when the last edition published. The authors, recognizing that few readers will ever go on to construct a compiler, retain their focus on the broader set of problems faced in software design and software development. New chapters include: Chapter 10 Instruction-Level Parallelism Chapter 11 Optimizing for Parallelism and Locality Chapter 12 Interprocedural Analysis