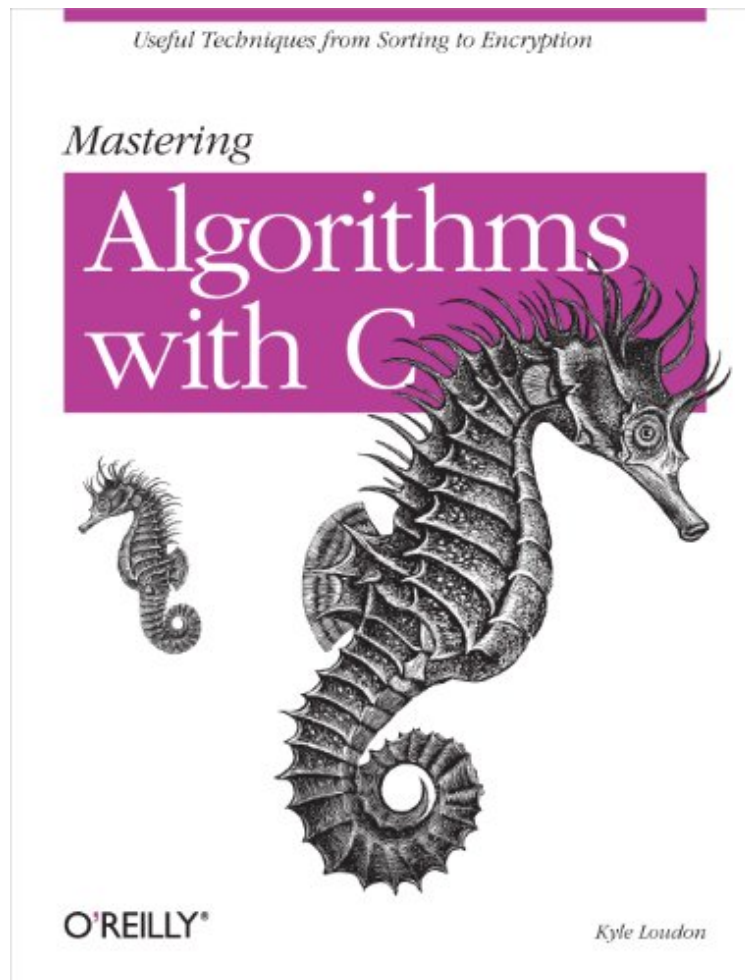


[Read download] Mastering Algorithms with C: Useful Techniques from Sorting to Encryption

Mastering Algorithms with C: Useful Techniques from Sorting to Encryption

Von Kyle Loudon

*Download PDF | ePub | DOC | audiobook | ebooks



DOWNLOAD



+

READ ONLINE

Produktinformation -Verkaufsrank: #448726 in eBooksVerffentlicht am: 1999-08-05Erscheinungsdatum: 2009-02-09File Name: B0043EWW5Q | File size: 19.Mb

Von Kyle Loudon : Mastering Algorithms with C: Useful Techniques from Sorting to Encryption before purchasing it in order to gage whether or not it would be worth my time, and all praised Mastering Algorithms with C: Useful Techniques from Sorting to Encryption:

KundenrezensionenHilfreichste Kundenrezensionen5 von 5 Kunden fanden die folgende Rezension hilfreich. A small collection of over-commented codeVon Cris LuengoI hoped to find a broad overview of existing algorithms. Instead I got a small collection of over-commented code. It is true that this book contains some algorithms not discussed in similar books, but it also doesn't contain many algorithms that I consider a part of a programmer's standard toolbox. For example, there is no mention of string matching and searching. The author discusses an implementation of the LZ77 compression algorithm, but doesn't even give a hint as to the workings of related (better) algorithms. There is an

implementation of the RSA public key encryption, but no mention on how the private encryption/public decryption counterpart works. The discussion on hash tables only covers hash tables with linked lists, and not the easier to implement double hashing. And I could go on and on. But the most annoying thing about the book is the author's commenting style, which render the code itself unreadable. Almost every instruction is commented by a 5-line high, 80-column wide comment block, exactly in the way that Rob Pike makes fun of in his "Notes on Programming in C" article, where he says: "Don't laugh now, wait until you see it in real life." In short, I would only recommend this book to people wanting to learn some basic algorithms, but already have a strong sense of correct programming style.

2 von 2 Kunden fanden die folgende Rezension hilfreich. Fantastic addition to any personal library. Von B. S. Clewett Just what I needed. One of those books I had to ring in sick to work, sit at home and read. Writing commercial c applications for the Internet, I was looking for a book to give me assistance with my coding. What I found was something which caused me to go back and rewrite a lot of my work! Starting by explaining exactly how memory is managed when calling functions, going on to explain, in english, how many commonly used structures in c work. Everything from hash tables to encryption. A good read for anybody interested in how code works. Essential for anybody wishing to write quality programs. The best bit. This comes with c libraries uncompiled with examples, fully explained in the text. Many of which I have used. Reliable, easy to implement and dramatically increase the quality of my code. Want a easy to use binary search tree? A few minutes work and it's there. I also emaild the author, who was more than happy to help with a question. Personally speaking, one of the best computer books I have ever bought.

0 von 0 Kunden fanden die folgende Rezension hilfreich. Exceptional writing, elegant code, great examples Von Ein Kunde Mastering Algorithms in C is the most readable algorithms book I've ever encountered. Not only does the author have a tremendous command of English, he has a writing style that is simply a pleasure to read. The author also deserves mention as having one of the cleanest coding styles I've come across. Having taught and worked with computers for over 15 years, I've seen many. It is no easy feat to present the subject of algorithms using real C code in a consistently elegant manner. This book does it wonderfully. Another feature of the book that works exceptionally well is its detailed presentation of interesting (and I emphasize interesting) real-world examples of how various data structures and algorithms in the book are actually applied. I'm a computer science type, so I especially enjoyed the examples about virtual memory managers, lexical analyzers, and packet-switching over the Internet. But the book includes many other examples of more general interest. Students will find all of the examples particularly insightful. Although most of the code in the book does make use of many of the more advanced features of C, an inordinate number of comments have been included which should help even the feeblest of programmer carry on. In addition, there are two great chapters on pointers and recursion. Exceptional writing, elegant code, great examples, not to mention a lot of entertainment value -- O'Reilly has another winner here. I highly recommend it.

Kurzbeschreibung There are many books on data structures and algorithms, including some with useful libraries of C functions. Mastering Algorithms with C offers you a unique combination of theoretical background and working code. With robust solutions for everyday programming tasks, this book avoids the abstract style of most classic data structures and algorithms texts, but still provides all of the information you need to understand the purpose and use of common programming techniques. Implementations, as well as interesting, real-world examples of each data structure and algorithm, are included. Using both a programming style and a writing style that are exceptionally clean, Kyle Loudon shows you how to use such essential data structures as lists, stacks, queues, sets, trees, heaps, priority queues, and graphs. He explains how to use algorithms for sorting, searching, numerical analysis, data compression, data encryption, common graph problems, and computational geometry. And he describes the relative efficiency of all implementations. The compression and encryption chapters not only give you working code for reasonably efficient solutions, they offer explanations of concepts in an approachable manner for people who never have had the time or expertise to study them in depth. Anyone with a basic understanding of the C language can use this book. In order to provide maintainable and extendible code, an extra level of abstraction (such as pointers to functions) is used in examples where appropriate. Understanding that these techniques may be unfamiliar to some programmers, Loudon explains them clearly in the introductory chapters. Contents include: Pointers Recursion Analysis of algorithms Data structures (lists, stacks, queues, sets, hash tables, trees, heaps, priority queues, graphs) Sorting and searching Numerical methods Data compression Data encryption Graph algorithms Geometric algorithms