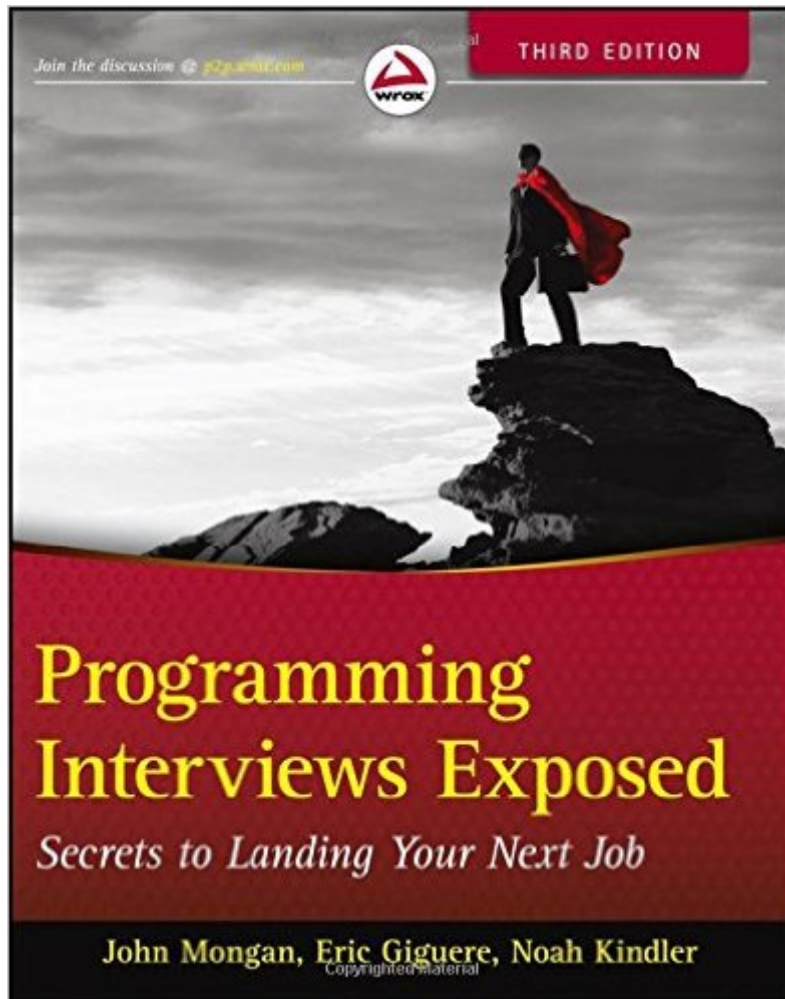The book was found

Programming Interviews Exposed: Secrets To Landing Your Next Job

Synopsis

Be prepared for your next job interview with this tried-and-true advice In today's tight job market, competition for programming jobs is hotter than ever. This third edition of a popular guide to programming interviews includes new code examples, information on the latest languages, new chapters on sorting and design patterns, tips on using LinkedIn, and a downloadable app to help prepare applicants for the interview. Like its earlier editions, this guide covers what software companies and IT departments want their programmers to know and includes plenty of helpful hints to boost your confidence. Looks at current job search and hiring processes, such as the rise of LinkedIn and other social networks as recruiting resources Addresses the most important languages for a programmer to know and features examples in multiple languages Includes new programming questions designed to sharpen your knowledge Features all-new chapters on design patterns and sorting, including how to deal with memory constraints and mobility issues Walk into your next job interview with confidence, knowing you have thoroughly studied this newest edition of Programming Interviews Exposed.

Book Information

Paperback: 336 pages

Publisher: Wrox; 3 edition (November 13, 2012)

Language: English

ISBN-10: 1118261364

ISBN-13: 978-1118261361

Product Dimensions: 7.4 x 0.6 x 9.3 inches

Shipping Weight: 1.2 pounds (View shipping rates and policies)

Average Customer Review: 4.4 out of 5 starsÂ Â See all reviewsÂ (94 customer reviews)

Best Sellers Rank: #9,132 in Books (See Top 100 in Books)   #10 inÂ Books > Textbooks > Computer Science > Software Design & Engineering   #10 inÂ Books > Business & Money > Job Hunting & Careers > Interviewing   #16 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Development

Customer Reviews

I worked with the data science association on their new standards for "Data Scientist" interviews (entry salary of $125,000), and both real questions and after interview polls were included for the biggest names in data today, from the web to corporate and government IT. "Data Scientist" is one of the hottest new jobs out there today, and some companies are even forming CDSO jobs--Chief

Data Science Officer!To begin, ALL FOUR of the books in this review are 5 star "superstars" for IT interviews. The two problems are, my library customers want to know the top two, and our shoppers want to know if they can get away with one, two, three, or if they have to buy all four! Of course the answer depends both on the focus of your resume, and the overlap/focus in the four books.First, the summary, by author, title/ link, year published/edition, number of pages, trim and cost, problems included, main language(s) foci. These four are the most frequently purchased by the over 100,000 libraries (including corporate technical libraries and schools as well as private and public) in our database. (Note: page counts are via visual inspection at the time of this writing, not stats. Pages can vary with on-demand books.).Aziz,Â Elements of Programming Interviews: 300 Questions and Solutions by Aziz, Adnan, Prakash, Amit, Lee, Tsung-Hsien 1st (first) Edition (10/11/2012), 2012, 481 pages, 6 x 9, $25, 300 problems (mostly C++, concurrency in Java, discrete math in formulas and English)McDowell,Â Cracking the Coding Interview: 150 Programming Questions and Solutions, 2011 (5th edition), 500 pages, 6 x 9, $23, 150 problems, (mostly all Java except of course the C, C++ question sections!)Guiness,Â Ace the Programming Interview: 160 Questions and Answers for Success, 2013, 419 pages, 6 x 9, $20, 160 problems, (mostly Java and C# but some unusual JavaScript, SQL, Ruby and Perl examples too)Mongan,Â Programming Interviews Exposed: Secrets to Landing Your Next Job, 2013 (ed. 3), 301 pages, 7.4 x 9, $18, 150+ problems (C, C++, C#, Java)All four of these fine prep texts cover the usual suspects in Algorithms and Data structures, including a focus on "scalable" problems of most concern to the s, Googles, Facebooks, etc. of the world. These include recursion, arrays, lists, hash tables, binary searches and trees, and other foundation coding subjects.All also cover the usual tricks, brain teasers, presentation problems, prep, process, etc. issues, and in the case of Cracking, specifics on many different company processes.The divergence is in the "extras." Aziz jumps into parallel computing and covers discrete math (in grad school joke terms, all the computer oriented math that has been taken out of high school courses). McDowell has an unusually well written probability section. Guiness is very up to date with cross platform apps and concurrent programming nightmares, and goes into both more depth and detail on individual topics like big O notation. Mongan is published by wrox, and has not only technical editors, but outstanding web resources. His database section is the most robust of the group.Aziz and McDowell are print on demand, which usually means there are many more errors in early going, but much faster correction of them via almost weekly files to the printer. Guiness is Wiley and bulletproof. We've tested the code extensively in all four (my payroy sister programmers, not me!) and ALL of them are outstanding, with very few errors at this writing, which can only get better fast in the two PODs, and wasn't a

problem to begin with via the technically edited wrox and wiley teams.Surprisingly, there is NOT a lot of overlap in solutions in these four texts, just as there IS a lot of overlap in the questions (strings, arrays, binaries, hashes... structures are structures and algos are algos). The difference in ALL these books (as opposed to a Cormen) is that the algorithm examples are not academic--they give you many options to "cheat" - and most of the cheats are more real world than techniques given in the 1,300 page algo function texts.McDowell is the industry standard, but she teaches very much to Google, as does Aziz, meaning web focus, and even a little forgiveness on php, but NO forgiveness on memory or scalability. If you're a library client and have to pick two, we advise one from the McDowell/ Aziz dyad and one from the Guiness/ Mongan dyad. If you're applying for a job with a specific language requirement, these self sort, although of course all are object oriented today.For shoppers preparing for a real interview: buy all four. I mean, come on. This is your future! You can get all four for the price you'd pay for a larger (way less useful) algo + data structure or individual language text, and maybe less. Some points about interview technique are common, but all four offer different and important examples in approaches to solutions, even though they share common algorithmic and data structure challenges.IRONY: The only programming area growing faster than data scientist today is at the other end of the big scale spectrum: embedded systems. I kid you not, specialize in embedded, and you're GUARANTEED a dream job, both due to the explosion of these systems, and the rarity of programmers here (but yes, you have to get into circuits!). Our sister Payroy group shows job stats, demand and salaries that are to die for if you go there-- way better than Google. NONE of these books cover it (because other than mobile and server embeds, embedded was traditionally automotive and industrial, but even "Google and Microsoft TV" type ventures are now hungering for it).There is NO good interview book out on embedded yet, but these two are the best of breed in the field itself: 1. Samek (Practical UML Statecharts in C/C++: Event-Driven Programming for Embedded Systems) and 2. White (Making Embedded Systems: Design Patterns for Great Software). Why C and C++? Because that's where the majority of electronics still reside, and "object" programmers in the field often just use the C subset of ++ and don't really get into sexy classes/methods/parents/kids, etc.! 6 months brushing up on this, specializing, and going for an embedded job will be worth years of competing with the interviews in these texts!!!Now, a simple tip. I was part of a team that interviewed for a high level, very high paying digital art programming position at shader joes dot com. One candidate stood out as really technically challenged--she even confused a call with a register in one of her answers! She called herself an "autodidact" - meaning, unlike Yahoo, we can't be recruiting only from the 18 top schools.At the end of her interview, she asked us to check out a disc she'd brought. She had

programmed her own video game with movie-real characters, explosions, storyline, etc. using Unity, Maya, blastcode, Python, Lua and C#, with web distributions in Java, HTML 5 and php. She proceeded to explain her entire process, from idea to distribution. She was hired before she could reach the elevator. In olden-days, old timer parlance, don't forget your "portfolio" if you have one! It can trump a LOT of the bureaucratic hurdles!EMAILERS ANSWER: IF you are a manager, rusty at coding, a data scientist, etc. and are in an interview where you have to "understand" coding basics, but not necessarily code, see our review of Karumanchi (Coding Interview Questions).Library Picks reviews only for the benefit of shoppers and has nothing to do with , the authors, manufacturers or publishers of the items we review. We always buy the items we review for the sake of objectivity, and although we search for gems, are not shy about trashing an item if it's a waste of time or money for shoppers. If the reviewer identifies herself, her job or her field, it is only as a point of reference to help you gauge the background and any biases.

I was a Software Engineer Intern at Facebook before. This book provides an introduction level of text about how those job interview processes go. As a student who never worked at industry before, it is really useful to get know all those details before hand. However, for the problems part, I felt there is a miss in there since there are not enough problems (although those problems are well-chosen for sure).After reading this book and having some initial shaky phone interviews, I have to find some other materials to sharpen my problem solving skills. Since personally I know all the basic algorithms and data structures, what I need is practicing lots of problems to apply those skills. I borrowed one copy of Elements of Programming Interviews: 300 Questions and Solutions from my friend which provides rich problems (a little too much to me at that time but definitely a great resource from practice perspective).In conclusion, this book is good for you to know the interview process (but not enough for others), and you should definitely look for some other materials for practice.

I bought this book after finishing (or giving up on) all the problems in Cracking the Coding Interview. This is not more of the same. The problems here are much easier and fewer and often duplicates. The bulk of this book consists of overly detailed explanations of the problems and more than a review of basic comp sci concepts.If somehow you're going to be appearing for a technical interview and are just starting to program, this may be helpful (though there still aren't enough problems). Or maybe you found the solutions in Cracking to be too brief.There are very few problems that really twist your brain up and make you scratch your head for a whileÃ¢Â Â"like the kind you'll actually get

from companies. For those see Cracking or CareerCup.com. I'd love more suggestions for better questions.

Programming Interviews Exposed (PIE) should be the very first one book seriously talking about how to prepare for programming interviews. Its release clearly sets a standard for all following similar books, like Cracking the Coding Interviews and Elements of Programming Interviews. PIE use interesting and real interview problems as a unit to discuss about the solution, and possible pitfalls you may face during real interviews. However, since the content of this book has not being updated for a long time, my friends and I always joke about that people should not start their interview problems if they cannot solve PIE's problems; those problems are too easy from the point of view nowadays. Therefore, please don't treat this book as an elixir for your programming interviews; you should find some others if you are serious.

Love this book for interview prep. I have both this one and the 'Cracking the Coding Interview' and I think I slightly lean in favor of this one. The other book has more problems, with in depth solutions, but I like the topic coverage of this book. I think all concepts are explained very well. I hate coding interviews, and have issues with nerves that make it very challenging for me to think in that kind of setting. The only way to combat that is to practice and be confident in your knowledge. This book certainly will help in the later. I highly recommend having both books, but if you are only going to go with one I would make this one it.

I was able to read - and understand - the entire book over a single weekend. It provides a thorough walk-trough for the base problems, with hints and common mistakes. I did not have to memorize anything since I gained a solid understanding and could repeat it myself.The books then builds up on the dissected base problems with common derivations and twists.

Programming Interviews Exposed: Secrets to Landing Your Next Job INTERVIEW: 12 Steps To Successful Job Interviews To End Your Job Search, Get Hired (Finding A Job, Google Interview, Interview Skills, Interview Questions, Career Change, Job Interview, Negotiation) Adobe Certified Expert (ACE) Secrets To Acing The Exam and Successful Finding And Landing Your Next Adobe Certified Expert (ACE) Certified Job Adobe Certified Instructor (ACI) Secrets To Acing The Exam and Successful Finding And Landing Your Next Adobe Certified Instructor (ACI) Certified Job Knock 'em Dead Job Interview: How to Turn Job Interviews Into Job Offers Landing a Federal Legal Job:

Finding Success in the U.S. Government Job Market Interview: How To Best Prepare For An Interview And Land Your Dream Job In 2016! (Interview, Interviewing, Successful Interview, Interview Tips, Job Interview, ... Job Offer, Interview Questions, Dream Job) Interview: Job Interview: HOW TO PREPARE FOR A JOB INTERVIEW AND MAKE SURE YOU GET THE JOB YOU DESIRE!: (+2nd FREE BOOK) 50+ Most Essential Questions (Interview,Search,Hunting,Job Interview) Job Escape Plan: The 7 Steps to Build a Home Business, Quit your Job and Enjoy the Freedom: Includes Interviews of John Lee Dumas, Nick Loper, Rob Cubbon, Steve Scott, Stefan Pylarinos & others! Knock 'em Dead Job Interview: How to Turn Job Interviews into Paychecks Can I Wear My Nose Ring to the Interview?: A Crash Course in Finding, Landing, and Keeping Your First Real Job C++: A Smart Way to Learn C++ Programming and Javascript (c plus plus, C++ for beginners, JAVA, programming computer, hacking, hacking exposed) (C ... Coding, CSS, Java, PHP) (Volume 1) How to Become a Video Game Artist: The Insider's Guide to Landing a Job in the Gaming World Cracking the Tech Career: Insider Advice on Landing a Job at Google, Microsoft, Apple, or any Top Tech Company What Color Is Your Parachute? Guide to Rethinking Interviews: Ace the Interview and Land Your Dream Job (What Color Is Your Parachute Guide to Rethinking..) Minecraft: Minecraft Secrets Handbook : Master these Ultimate Minecraft game tips, tricks, secrets and hints to take your game to the next level. (Unofficial Minecraft Secrets Guide) How to Write Powerful College Student Resumes and Cover Letters: Secrets That Get Job Interviews Like Magic Interviewing: Interview Questions - Job Interview ! Learn How to Job Interview and Master the Key Interview Skills! BONUS INCLUDED! 37 Ways to Have Unstoppable ... Interview! GET THE JOB YOU DESERVE! Book 1) GET THAT BODYGUARD JOB NOW: HOW TO LAND A LUCRATIVE BODYGUARD JOB IN TODAY'S JOB MARKET Sharkproof: Get the Job You Want, Keep the Job You Love... in Today's Frenzied Job Market

Dmca