

Programming Pearls

This book describes data structures and data structure design techniques for functional languages.

Implementing physical simulations for real-time games is a complex task that requires a solid understanding of a wide range of concepts from the fields of mathematics, physics, and software engineering. This book is a gems-like collection of practical articles in the area of game physics. Each provides hands-on detail that can be used in practical

Widely considered one of the best practical guides to programming, Steve McConnell's original `CODE COMPLETE` has been helping developers write better software for more than a decade. Now this classic book has been fully updated and revised with leading-edge practices—and hundreds of new code samples—illustrating the art and science of software construction. Capturing the body of knowledge available from research, academia, and everyday commercial practice, McConnell synthesizes the most effective techniques and must-know principles into clear, pragmatic guidance. No matter what your experience level, development environment, or project size, this book will inform and stimulate your thinking—and help you build the highest quality code. Discover the timeless techniques and strategies that help you: Design for minimum complexity and maximum creativity Reap the benefits of collaborative development

Download File PDF Programming Pearls

Apply defensive programming techniques to reduce and flush out errors Exploit opportunities to refactor—or evolve—code, and do it safely Use construction practices that are right-weight for your project Debug problems quickly and effectively Resolve critical construction issues early and correctly Build quality into the beginning, middle, and end of your project

Literate programming is a programming methodology that combines a programming language with a documentation language, making programs more easily maintained than programs written only in a high-level language. A literate programmer is an essayist who writes programs for humans to understand. When programs are written in the recommended style they can be transformed into documents by a document compiler and into efficient code by an algebraic compiler. This anthology of essays includes Knuth's early papers on related topics such as structured programming as well as the Computer Journal article that launched literate programming. Many examples are given, including excerpts from the programs for TeX and METAFONT. The final essay is an example of CWEB, a system for literate programming in C and related languages. Index included.

This collection of essays drawn from Plauger's popular "Programming on Purpose" column in the magazine Computer Language, focuses on the technology of writing computer software. Plauger's style is clear without being simplistic, reducing complex themes to bite-size chunks. KEY TOPICS: Covers a number of important technical

Download File PDF Programming Pearls

themes such as computer arithmetic, approximating math functions, human perception and artificial intelligence, encrypting data and clarifying documentation.

Learn how to take full advantage of Apache Kafka, the distributed, publish-subscribe queue for handling real-time data feeds. With this comprehensive book, you will understand how Kafka works and how it is designed. Authors Neha Narkhede, Gwen Shapira, and Todd Palino show you how to deploy production Kafka clusters; secure, tune, and monitor them; write rock-solid applications that use Kafka; and build scalable stream-processing applications. Learn how Kafka compares to other queues, and where it fits in the big data ecosystem. Dive into Kafka's internal design Pick up best practices for developing applications that use Kafka. Understand the best way to deploy Kafka in production monitoring, tuning, and maintenance tasks. Learn how to secure a Kafka cluster.

Write clean code that works with the help of this groundbreaking software method. Example-driven teaching is the basis of Beck's step-by-step instruction that will have readers using TDD to further their projects.

Have you ever... - Wanted to work at an exciting futuristic company? - Struggled with an interview problem that could have been solved in 15 minutes? - Wished you could study real-world computing problems? If so, you need to read Elements of Programming Interviews (EPI). EPI is your comprehensive guide to interviewing for software development roles. The core of EPI is a collection of over 250 problems with detailed

Download File PDF Programming Pearls

solutions. The problems are representative of interview questions asked at leading software companies. The problems are illustrated with 200 figures, 300 tested programs, and 150 additional variants. The book begins with a summary of the nontechnical aspects of interviewing, such as strategies for a great interview, common mistakes, perspectives from the other side of the table, tips on negotiating the best offer, and a guide to the best ways to use EPI. We also provide a summary of data structures, algorithms, and problem solving patterns. Coding problems are presented through a series of chapters on basic and advanced data structures, searching, sorting, algorithm design principles, and concurrency. Each chapter starts with a brief introduction, a case study, top tips, and a review of the most important library methods. This is followed by a broad and thought-provoking set of problems. A practical, fun approach to computer science fundamentals, as seen through the lens of common programming interview questions. Jeff Atwood/Co-founder, Stack Overflow and Discourse

This revised, 13-week study goes back to the biblical basics of prayer, cleaning out myths about prayer to rev up a powerful, ongoing connection to God that can invigorate every aspect of a Christian's life. The author tackles the complex theological questions: If God is sovereign, why pray? If prayer is not a way to change God's mind, what is it? Prayer is a way, Jennifer illustrates, for God to send His power and provision into the world through His people. The study takes

a detailed look at prayer's purpose, process, promise, and practice.

Get ready for interview success Programming jobs are on the rise, and the field is predicted to keep growing, fast. Landing one of these lucrative and rewarding jobs requires more than just being a good programmer. Programming Interviews For Dummies explains the skills and knowledge you need to ace the programming interview. Interviews for software development jobs and other programming positions are unique. Not only must candidates demonstrate technical savvy, they must also show that they're equipped to be a productive member of programming teams and ready to start solving problems from day one. This book demystifies both sides of the process, offering tips and techniques to help candidates and interviewers alike. Prepare for the most common interview questions Understand what employers are looking for Develop the skills to impress non-technical interviewers Learn how to assess candidates for programming roles Prove that you (or your new hires) can be productive from day one Programming Interviews For Dummies gives readers a clear view of both sides of the process, so prospective coders and interviewers alike will learn to ace the interview.

Felix Klein, one of the great nineteenth-century geometers, rediscovered in mathematics an idea from Eastern philosophy: the heaven of Indra contained a

Download File PDF Programming Pearls

net of pearls, each of which was reflected in its neighbour, so that the whole Universe was mirrored in each pearl. Klein studied infinitely repeated reflections and was led to forms with multiple co-existing symmetries. For a century these ideas barely existed outside the imagination of mathematicians. However in the 1980s the authors embarked on the first computer exploration of Klein's vision, and in doing so found many further extraordinary images. Join the authors on the path from basic mathematical ideas to the simple algorithms that create the delicate fractal filigrees, most of which have never appeared in print before. Beginners can follow the step-by-step instructions for writing programs that generate the images. Others can see how the images relate to ideas at the forefront of research.

When programmers list their favorite books, Jon Bentley's collection of programming pearls is commonly included among the classics. Just as natural pearls grow from grains of sand that irritate oysters, programming pearls have grown from real problems that have irritated real programmers. With origins beyond solid engineering, in the realm of insight and creativity, Bentley's pearls offer unique and clever solutions to those nagging problems. Illustrated by programs designed as much for fun as for instruction, the book is filled with lucid and witty descriptions of practical programming techniques and fundamental

Download File PDF Programming Pearls

design principles. It is not at all surprising that Programming Pearls has been so highly valued by programmers at every level of experience. In this revision, the first in 14 years, Bentley has substantially updated his essays to reflect current programming methods and environments. In addition, there are three new essays on testing, debugging, and timing set representations string problems All the original programs have been rewritten, and an equal amount of new code has been generated. Implementations of all the programs, in C or C++, are now available on the Web. What remains the same in this new edition is Bentley's focus on the hard core of programming problems and his delivery of workable solutions to those problems. Whether you are new to Bentley's classic or are revisiting his work for some fresh insight, the book is sure to make your own list of favorites.

Peter Seibel interviews 15 of the most interesting computer programmers alive today in *Coders at Work*, offering a companion volume to Apress's highly acclaimed best-seller *Founders at Work* by Jessica Livingston. As the words "at work" suggest, Peter Seibel focuses on how his interviewees tackle the day-to-day work of programming, while revealing much more, like how they became great programmers, how they recognize programming talent in others, and what kinds of problems they find most interesting. Hundreds of people have suggested

Download File PDF Programming Pearls

names of programmers to interview on the Coders at Work web site: www.codersatwork.com. The complete list was 284 names. Having digested everyone's feedback, we selected 15 folks who've been kind enough to agree to be interviewed: Frances Allen: Pioneer in optimizing compilers, first woman to win the Turing Award (2006) and first female IBM fellow Joe Armstrong: Inventor of Erlang Joshua Bloch: Author of the Java collections framework, now at Google Bernie Cosell: One of the main software guys behind the original ARPANET IMPs and a master debugger Douglas Crockford: JSON founder, JavaScript architect at Yahoo! L. Peter Deutsch: Author of Ghostscript, implementer of Smalltalk-80 at Xerox PARC and Lisp 1.5 on PDP-1 Brendan Eich: Inventor of JavaScript, CTO of the Mozilla Corporation Brad Fitzpatrick: Writer of LiveJournal, OpenID, memcached, and Perlbal Dan Ingalls: Smalltalk implementor and designer Simon Peyton Jones: Coinventor of Haskell and lead designer of Glasgow Haskell Compiler Donald Knuth: Author of The Art of Computer Programming and creator of TeX Peter Norvig: Director of Research at Google and author of the standard text on AI Guy Steele: Coinventor of Scheme and part of the Common Lisp Gang of Five, currently working on Fortress Ken Thompson: Inventor of UNIX Jamie Zawinski: Author of XEmacs and early Netscape/Mozilla hacker

Download File PDF Programming Pearls

The Software Insider's Guide to Getting Hired and Getting to the Top! Here's all the information you need to jumpstart your software career: the best ways to get hired, move up, and blaze your way to the top! The software business has radically changed, and this book reveals today's realities—everything your professors and corporate managers never told you. In his 20 years at IBM as a software architect, senior manager, and lead programmer, Sam Lightstone has briefed dozens of leading companies and universities on careers, new technology, and emerging areas of research. He currently works on one of the world's largest software development teams and spends a good part of his time recruiting and mentoring software engineers. This book shares all the lessons for success Sam has learned...plus powerful insights from 17 of the industry's biggest stars. Want to make it big in software? Start right here! Discover how to • Get your next job in software development • Master the nontechnical skills crucial to your success • “Work the org” to move up rapidly • Successfully manage your time, projects, and life • Avoid “killer” mistakes that could destroy your career • Move up to “medium-shot,” “big-shot,” and finally, “visionary” • Launch your own winning software company Exclusive interviews with Steve Wozniak, Inventor, Apple computer John Schwarz, CEO, Business Objects James Gosling, Inventor, Java programming language Marissa Mayer, Google VP, Search

Download File PDF Programming Pearls

Products and User Experience Jon Bentley, Author, Programming Pearls Marc Benioff, CEO and founder, Salesforce.com Grady Booch, IBM Fellow and co-founder Rational Software Bjarne Stroustrup, Inventor, C++ programming language David Vaskevitch, Microsoft CTO Linus Torvalds, Creator, Linux operating system kernel Richard Stallman, Founder, Free software movement Peter Norvig, Google's Director of Research Mark Russinovich, Microsoft Fellow and Windows Architect Tom Malloy, Adobe Chief Software Architect Diane Greene, Co-founder and past CEO of VMware Robert Kahn, Co-inventor, the Internet Ray Tomlinson, Inventor, email

This book constitutes the refereed proceedings of the ACM SIGPLAN/SIGSOFT Conference on Generative Programming and Component Engineering, GPCE 2002, held in Pittsburgh, PA, USA in October 2002. The 18 revised full papers presented were carefully reviewed and selected from 39 submissions. Among the topics covered are generative programming, meta-programming, program specialization, program analysis, program transformation, domain-specific languages, software architectures, aspect-oriented programming, and component-based systems.

Tap into the wisdom of experts to learn what every programmer should know, no matter what language you use. With the 97 short and extremely useful tips for

Download File PDF Programming Pearls

programmers in this book, you'll expand your skills by adopting new approaches to old problems, learning appropriate best practices, and honing your craft through sound advice. With contributions from some of the most experienced and respected practitioners in the industry--including Michael Feathers, Pete Goodliffe, Diomidis Spinellis, Cay Horstmann, Verity Stob, and many more--this book contains practical knowledge and principles that you can apply to all kinds of projects. A few of the 97 things you should know: "Code in the Language of the Domain" by Dan North "Write Tests for People" by Gerard Meszaros "Convenience Is Not an -ility" by Gregor Hohpe "Know Your IDE" by Heinz Kabutz "A Message to the Future" by Linda Rising "The Boy Scout Rule" by Robert C. Martin (Uncle Bob) "Beware the Share" by Udi Dahan

The pressure is on during the interview process but with the right preparation, you can walk away with your dream job. This classic book uncovers what interviews are really like at America's top software and computer companies and provides you with the tools to succeed in any situation. The authors take you step-by-step through new problems and complex brainteasers they were asked during recent technical interviews. 50 interview scenarios are presented along with in-depth analysis of the possible solutions. The problem-solving process is clearly illustrated so you'll be able to easily apply what you've learned during crunch time. You'll also find expert tips on what questions to ask,

Download File PDF Programming Pearls

how to approach a problem, and how to recover if you become stuck. All of this will help you ace the interview and get the job you want. What you will learn from this book

- Tips for effectively completing the job application
- Ways to prepare for the entire programming interview process
- How to find the kind of programming job that fits you best
- Strategies for choosing a solution and what your approach says about you
- How to improve your interviewing skills so that you can respond to any question or situation
- Techniques for solving knowledge-based problems, logic puzzles, and programming problems

Who this book is for: This book is for programmers and developers applying for jobs in the software industry or in IT departments of major corporations.

Wrox Beginning guides are crafted to make learning programming languages and technologies easier than you think, providing a structured, tutorial format that will guide you through all the techniques involved.

Classic on practical methods of optimizing programs: This book gives practical advice on improving the efficiency (optimizing) programs and the limits there of. While showing how to trade off speed for space or vice-versa, the author points out the limits that can be expected to gain. His list of techniques is a collection of practical approaches rather than theoretical possibilities. At 158 pages (not counting index) this book is eminently readable, accessible and useful. Clearly written and well organized this is a book to keep on your shelf for when a program needs improving. It is also a book to read before a program as a reminder not to make things complicated with optimization that aren't

Download File PDF Programming Pearls

needed.

Presents the pinnacle of writing on C++ by renowned experts in the field, and is a must-read for today's C++ programmer.

High Performance Parallelism Pearls Volume 2 offers another set of examples that demonstrate how to leverage parallelism. Similar to Volume 1, the techniques included here explain how to use processors and coprocessors with the same programming - illustrating the most effective ways to combine Xeon Phi coprocessors with Xeon and other multicore processors. The book includes examples of successful programming efforts, drawn from across industries and domains such as biomed, genetics, finance, manufacturing, imaging, and more. Each chapter in this edited work includes detailed explanations of the programming techniques used, while showing high performance results on both Intel Xeon Phi coprocessors and multicore processors. Learn from dozens of new examples and case studies illustrating "success stories" demonstrating not just the features of Xeon-powered systems, but also how to leverage parallelism across these heterogeneous systems. Promotes write-once, run-anywhere coding, showing how to code for high performance on multicore processors and Xeon Phi. Examples from multiple vertical domains illustrating real-world use of Xeon Phi coprocessors. Source code available for download to facilitate further exploration. A guide to practical programming techniques and design principles, with information on such topics as testing, debugging and timing, set representations, and string problems.

Download File PDF Programming Pearls

Ace technical interviews with smart preparation *Programming Interviews Exposed* is the programmer's ideal first choice for technical interview preparation. Updated to reflect changing techniques and trends, this new fourth edition provides insider guidance on the unique interview process that today's programmers face. Online coding contests are being used to screen candidate pools of thousands, take-home projects have become commonplace, and employers are even evaluating a candidate's public code repositories at GitHub—and with competition becoming increasingly fierce, programmers need to shape themselves into the ideal candidate well in advance of the interview. This book doesn't just give you a collection of questions and answers, it walks you through the process of coming up with the solution so you learn the skills and techniques to shine on whatever problems you're given. This edition combines a thoroughly revised basis in classic questions involving fundamental data structures and algorithms with problems and step-by-step procedures for new topics including probability, data science, statistics, and machine learning which will help you fully prepare for whatever comes your way. Learn what the interviewer needs to hear to move you forward in the process Adopt an effective approach to phone screens with non-technical recruiters Examine common interview problems and tests with expert explanations Be ready to demonstrate your skills verbally, in contests, on GitHub, and more Technical jobs require the skillset, but you won't get hired unless you are able to effectively and efficiently demonstrate that skillset under pressure, in competition with hundreds of

Download File PDF Programming Pearls

others with the same background. Programming Interviews Exposed teaches you the interview skills you need to stand out as the best applicant to help you get the job you want.

Maira Kalman, the author of the bestsellers *The Principles of Uncertainty* and *The Elements of Style*, and Alex Kalman, the designer, curator, writer, and founder of Mmuseumm, combine their talents in this captivating family memoir, a creative blend of narrative and striking visuals that is a paean to an exceptional woman and a celebration of individuality, personal expression, and the art of living authentically. In the early 1950s, Jewish émigré Sara Berman arrived in the Bronx with her husband and two young daughters. When the children were grown, she and her husband returned to Israel, but Sara did not stay for long. In the late 1960s, at age sixty, she left her husband after thirty-eight years of marriage. One night, she packed a single suitcase and returned alone to New York City, moving into a studio apartment in Greenwich Village near her family. In her new home, Sara began discovering new things and establishing new rituals, from watching *Jeopardy* each night at 7:00 to eating pizza at the Museum of Modern Art's cafeteria every Wednesday. She also began discarding the unnecessary, according to the Kalmans: "in a burst of personal expression, she decided to wear only white." Sara kept her belongings in an extraordinarily clean and organized closet. Filled with elegant, minimalist, heavily starched, impeccably pressed and folded all-white clothing, including socks and undergarments, as well as carefully

Download File PDF Programming Pearls

selected objects—from a potato grater to her signature perfume, Chanel No.19—the space was sublime. Upon her death in 2004, her family decided to preserve its pristine contents, hoping to find a way to exhibit them one day. In 2015, the Mmuseumm, a new type of museum located in a series of unexpected locations founded and curated by Sara's grandson, Alex Kalman, recreated the space in a popular exhibit—Sara Berman's Closet—in Tribeca. The installation eventually moved to the Metropolitan Museum of Art, and in spring 2019, will become an outdoor monument to independence at the National Museum of American Jewish History on Independence Mall in Philadelphia. Inspired by the exhibit, this spectacular illustrated memoir, packed with family photographs, exclusive images, and Maira Kalman's distinctive paintings, is an ode to Sara's life, freedom, and re-invention. Sara Berman's Closet is an indelible portrait of the human experience—overcoming hardship, taking risks, experiencing joy, enduring loss. It is also a reminder of the significance of the seemingly insignificant moments in our lives—the moments we take for granted that may turn out to be the sweetest. Filled with a daughter and grandson's wry and touching observations conveyed in Maira's signature script, Sara Berman's Closet is a beautiful, loving tribute to one woman's indomitable spirit.

An industry insider explains why there is so much bad software—and why academia doesn't teach programmers what industry wants them to know. Why is software so prone to bugs? So vulnerable to viruses? Why are software products so often delayed,

Download File PDF Programming Pearls

or even canceled? Is software development really hard, or are software developers just not that good at it? In *The Problem with Software*, Adam Barr examines the proliferation of bad software, explains what causes it, and offers some suggestions on how to improve the situation. For one thing, Barr points out, academia doesn't teach programmers what they actually need to know to do their jobs: how to work in a team to create code that works reliably and can be maintained by somebody other than the original authors. As the size and complexity of commercial software have grown, the gap between academic computer science and industry has widened. It's an open secret that there is little engineering in software engineering, which continues to rely not on codified scientific knowledge but on intuition and experience. Barr, who worked as a programmer for more than twenty years, describes how the industry has evolved, from the era of mainframes and Fortran to today's embrace of the cloud. He explains bugs and why software has so many of them, and why today's interconnected computers offer fertile ground for viruses and worms. The difference between good and bad software can be a single line of code, and Barr includes code to illustrate the consequences of seemingly inconsequential choices by programmers. Looking to the future, Barr writes that the best prospect for improving software engineering is the move to the cloud. When software is a service and not a product, companies will have more incentive to make it good rather than "good enough to ship."

When programmers list their favorite books, Jon Bentley's collection of programming pearls is

Download File PDF Programming Pearls

commonly included among the classics. Just as natural pearls grow from grains of sand that irritate oysters, programming pearls have grown from real problems that have irritated real programmers. With origins beyond solid engineering, in the realm of insight and creativity, Bentley's pearls offer unique and clever solutions to those nagging problems. Illustrated by programs designed as much for fun as for instruction, the book is filled with lucid and witty descriptions of practical programming techniques and fundamental design principles. It is not at all surprising that Programming Pearls has been so highly valued by programmers at every level of experience. In this revision, the first in 14 years, Bentley has substantially updated his essays to reflect current programming methods and environments. In addition, there are three new essays on testing, debugging, and timing set representations string problems All the original programs have been rewritten, and an equal amount of new code has been generated. Implementations of all the programs, in C or C++, are now available on the Web. What remains the same in this new edition is Bentley's focus on the hard core of programming problems and his delivery of workable solutions to those problems. Whether you are new to Bentley's classic or are revisiting his work for some fresh insight, the book is sure to make your own list of favorites.

Authors Jim Jeffers and James Reinders spent two years helping educate customers about the prototype and pre-production hardware before Intel introduced the first Intel Xeon Phi coprocessor. They have distilled their own experiences coupled with insights from many expert customers, Intel Field Engineers, Application Engineers and Technical Consulting Engineers, to create this authoritative first book on the essentials of programming for this new architecture and these new products. This book is useful even before you ever touch a system with an Intel

Download File PDF Programming Pearls

Xeon Phi coprocessor. To ensure that your applications run at maximum efficiency, the authors emphasize key techniques for programming any modern parallel computing system whether based on Intel Xeon processors, Intel Xeon Phi coprocessors, or other high performance microprocessors. Applying these techniques will generally increase your program performance on any system, and better prepare you for Intel Xeon Phi coprocessors and the Intel MIC architecture. A practical guide to the essentials of the Intel Xeon Phi coprocessor Presents best practices for portable, high-performance computing and a familiar and proven threaded, scalar-vector programming model Includes simple but informative code examples that explain the unique aspects of this new highly parallel and high performance computational product Covers wide vectors, many cores, many threads and high bandwidth cache/memory architecture

Offering advice for programmers at entry level, each chapter in this work discusses problems the programmer might face on any day, and suggests methods to get around them. Introduces fundamental techniques for reasoning mathematically about functional programs. Ideal for a first- or second-year undergraduate course.

The core concepts and technologies of Windows networking Networking can be a complex topic, especially for those new to the field of IT. This focused, full-color book takes a unique approach to teaching Windows networking to beginners by stripping down a network to its bare basics, thereby making each topic clear and easy to understand. Focusing on the new Microsoft Technology Associate (MTA) program, this book pares down to just the essentials, showing beginners how to gain a solid foundation for understanding networking concepts upon which more advanced topics and technologies can be built. This straightforward guide begins

Download File PDF Programming Pearls

each chapter by laying out a list of topics to be discussed, followed by a concise discussion of the core networking skills you need to have to gain a strong handle on the subject matter. Chapters conclude with review questions and suggested labs so you can measure your level of understanding of the chapter's content. Serves as an ideal resource for gaining a solid understanding of fundamental networking concepts and skills Offers a straightforward and direct approach to networking basics and covers network management tools, TCP/IP, the name resolution process, and network protocols and topologies Reviews all the topics you need to know for taking the MTA 98-366 exam Provides an overview of networking components, discusses connecting computers to a network, and looks at connecting networks with routers If you're new to IT and interested in entering the IT workforce, then Microsoft Windows Networking Essentials is essential reading.

Anyone who develops software for a living needs a proven way to produce it better, faster, and cheaper. The Productive Programmer offers critical timesaving and productivity tools that you can adopt right away, no matter what platform you use. Master developer Neal Ford not only offers advice on the mechanics of productivity-how to work smarter, spurn interruptions, get the most out your computer, and avoid repetition-he also details valuable practices that will help you elude common traps, improve your code, and become more valuable to your team. You'll learn to: Write the test before you write the code Manage the lifecycle of your objects fastidiously Build only what you need now, not what you might need later Apply ancient philosophies to software development Question authority, rather than blindly adhere to standards Make hard things easier and impossible things possible through meta-programming Be sure all code within a method is at the same level of abstraction Pick the right editor and

Download File PDF Programming Pearls

assemble the best tools for the job This isn't theory, but the fruits of Ford's real-world experience as an Application Architect at the global IT consultancy ThoughtWorks. Whether you're a beginner or a pro with years of experience, you'll improve your work and your career with the simple and straightforward principles in *The Productive Programmer*.

Software -- Software Engineering.

Covers Expression, Structure, Common Blunders, Documentation, & Structured Programming Techniques

When looking for ways to improve your website, how do you decide which changes to make? And which changes to keep? This concise book shows you how to use Multiarmed Bandit algorithms to measure the real-world value of any modifications you make to your site. Author John Myles White shows you how this powerful class of algorithms can help you boost website traffic, convert visitors to customers, and increase many other measures of success. This is the first developer-focused book on bandit algorithms, which were previously described only in research papers. You'll quickly learn the benefits of several simple algorithms—including the epsilon-Greedy, Softmax, and Upper Confidence Bound (UCB) algorithms—by working through code examples written in Python, which you can easily adapt for deployment on your own website. Learn the basics of A/B testing—and recognize when it's better to use bandit algorithms

Develop a unit testing framework for debugging bandit algorithms Get additional

Download File PDF Programming Pearls

code examples written in Julia, Ruby, and JavaScript with supplemental online materials

Programming Pearls Addison-Wesley Professional

Brian Kernighan and Rob Pike have written *The Practice of Programming* to help make individual programmers more effective and productive. The practice of programming is more than just writing code. Programmers must also assess tradeoffs, choose among design alternatives, debug and test, improve performance, and maintain software written by themselves and others. At the same time, they must be concerned with issues like compatibility, robustness, and reliability, while meeting specifications. *The Practice of Programming* covers all these topics, and more. This book is full of practical advice and real-world examples in C, C++, Java, and a variety of special-purpose languages.

Richard Bird takes a radical approach to algorithm design, namely, design by calculation. These 30 short chapters each deal with a particular programming problem drawn from sources as diverse as games and puzzles, intriguing combinatorial tasks, and more familiar areas such as data compression and string matching. Each pearl starts with the statement of the problem expressed using the functional programming language Haskell, a powerful yet succinct language for capturing algorithmic ideas clearly and simply. The novel aspect of

Download File PDF Programming Pearls

the book is that each solution is calculated from an initial formulation of the problem in Haskell by appealing to the laws of functional programming. Pearls of Functional Algorithm Design will appeal to the aspiring functional programmer, students and teachers interested in the principles of algorithm design, and anyone seeking to master the techniques of reasoning about programs in an equational style.

More C++ Gems picks up where the first book left off, presenting tips, tricks, proven strategies, easy-to-follow techniques, and usable source code.

[Copyright: 17571201d81fa0fee26e2a2154e1587b](https://www.pdfdrive.com/programming-pearls-pdf/ebook/17571201d81fa0fee26e2a2154e1587b.html)