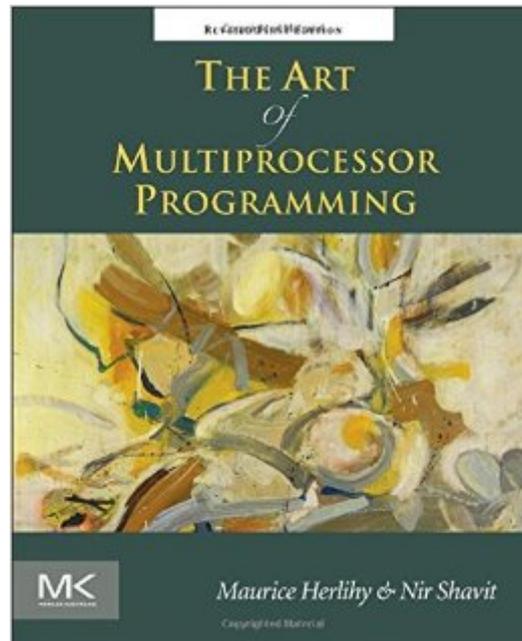


The book was found

# The Art Of Multiprocessor Programming, Revised Reprint



## Synopsis

Revised and updated with improvements conceived in parallel programming courses, *The Art of Multiprocessor Programming* is an authoritative guide to multicore programming. It introduces a higher level set of software development skills than that needed for efficient single-core programming. This book provides comprehensive coverage of the new principles, algorithms, and tools necessary for effective multiprocessor programming. Students and professionals alike will benefit from thorough coverage of key multiprocessor programming issues. This revised edition incorporates much-demanded updates throughout the book, based on feedback and corrections reported from classrooms since 2008. Learn the fundamentals of programming multiple threads, accessing shared memory, explore mainstream concurrent data structures and the key elements of their design, as well as synchronization techniques from simple locks to transactional memory systems. Visit the companion site and download source code, example Java programs, and materials to support and enhance the learning experience.

## Book Information

Paperback: 536 pages

Publisher: Morgan Kaufmann; 1 edition (June 5, 2012)

Language: English

ISBN-10: 0123973376

ISBN-13: 978-0123973375

Product Dimensions: 7.5 x 1.2 x 9.2 inches

Shipping Weight: 2.3 pounds (View shipping rates and policies)

Average Customer Review: 4.1 out of 5 stars [See all reviews](#) (30 customer reviews)

Best Sellers Rank: #258,856 in Books (See Top 100 in Books) #28 in [Books > Computers & Technology > Programming > Parallel Programming](#) #297 in [Books > Computers & Technology > Programming > Languages & Tools > Java](#) #374 in [Books > Textbooks > Computer Science > Software Design & Engineering](#)

## Customer Reviews

*The Art of Multiprocessor Programming* is an outstanding text that will soon become a classic. I give a chapter by chapter review of it below. Practitioners that are already well versed in parallel programming can jump directly to Chapter 7, however, I would suggest at least skimming Chapters 2, 3 and 4. Even those programmers who understand shared memory and locking may be shocked at how relaxed memory models or compiler optimizations can reorder operations causing innocent

looking code to break.-----Chapter 1 - IntroductionWhy is this book called "The Art of Multiprocessor Programming" and not "The Art of Parallel Programming?" It is not by accident. There is a directed effort to explain parallel programming concepts as they relate to multi-core (or many-core) architectures. In particular, shared-memory multiprocessors have specific implementation details, such as cache coherence policies, that directly affect parallel software run on such architectures. The introduction gives a brief overview of the direction of the text: principles and practice.-----Part 1 - PrinciplesChapter 2 - Mutual ExclusionMutual exclusion is a key concept to multi-threaded programming, and this chapter is rightly placed at the beginning of the text. This chapter presents some of the foundational concepts in parallel computing, such as, understanding time related to operation interleavings, pessimistic critical sections, forward progress, deadlocks and fairness. In addition, some of the classic algorithms are presented here, such as Lamport's Ticket Locking and Peterson's 2-Threaded Lock.

This book gives programmers the practical and theoretical tools they need to adapt to the proliferation of multi-core machines. It opens with six chapters on theoretical subjects. These chapters are fascinating in their own right as well as directly applicable to my daily work. I thought the most important subjects were wait-free synchronization (every method completes in a finite number of steps), lock-free synchronization (some method completes in a finite number of steps), and some computability proofs. The authors use computability to demonstrate the equivalence of several types of synchronization primitives. They also present some impossibility proofs that show you how to avoid trying to solve unsolvable problems. The computability results and synchronization guarantees combine to give you the tools to determine whether one concurrent algorithm is "better" than another. The remainder of the book is devoted to practical subjects. These chapters cover locks, a variety of data structures, work scheduling, and some miscellaneous topics. Java's `java.util.concurrent` package provides production-quality implementations of most of these data structures. The authors know this, and they use the data structures chapters to demonstrate generally applicable techniques while avoiding unnecessary implementation details. The work scheduling chapter is a sobering reminder of the difficulty inherent in fully exploiting highly parallel architectures. The authors show how to use recurrences to analyze the relative speedup an algorithm gains by running on  $P$  processors instead of a single processor. Combining this with the discussion of Amdal's Law earlier in the book we see that the essential math behind parallelism severely penalizes you for seemingly small sequential portions of your code.

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

The Art of Multiprocessor Programming, Revised Reprint Java: The Simple Guide to Learn Java Programming In No Time (Programming, Database, Java for dummies, coding books, java programming) (HTML, Javascript, Programming, Developers, Coding, CSS, PHP) (Volume 2) Citroen DS: Revised and updated edition (Classic Reprint) Phase Transformations in Metals and Alloys, Third Edition (Revised Reprint) Mosby's PDQ for Respiratory Care - Revised Reprint, 2e A Little Book of Dental Hygienists' Rules - Revised Reprint, 1e Rapid Review Pathology Revised Reprint: With STUDENT CONSULT Online Access, 3e Essentials of Psychiatric Mental Health Nursing - Revised Reprint, 2e Policy and Politics in Nursing and Healthcare - Revised Reprint, 6e (Mason, Policy and Politics in Nursing and Health Care) Python: Python Programming For Beginners - The Comprehensive Guide To Python Programming: Computer Programming, Computer Language, Computer Science Python: Python Programming Course: Learn the Crash Course to Learning the Basics of Python (Python Programming, Python Programming Course, Python Beginners Course) Swift Programming Artificial Intelligence: Made Easy, w/ Essential Programming Learn to Create your \* Problem Solving \* Algorithms! TODAY! w/ Machine ... engineering, r programming, iOS development) Delphi Programming with COM and ActiveX (Programming Series) (Charles River Media Programming) Java: The Ultimate Guide to Learn Java and Python Programming (Programming, Java, Database, Java for dummies, coding books, java programming) (HTML, ... Developers, Coding, CSS, PHP) (Volume 3) Programming #8:C Programming Success in a Day & Android Programming in a Day! PowerShell: For Beginners! Master The PowerShell Command Line In 24 Hours (Python Programming, Javascript, Computer Programming, C++, SQL, Computer Hacking, Programming) Excel VBA Programming: Learn Excel VBA Programming FAST and EASY! (Programming is Easy) (Volume 9) Python: Python Programming For Beginners - The Comprehensive Guide To Python Programming: Computer Programming, Computer Language, Computer Science (Machine Language) IEC 61131-3: Programming Industrial Automation Systems: Concepts and Programming Languages, Requirements for Programming Systems, Decision-Making Aids Art: Painting For Artists - Styles: Acrylic And Oil Painting (art history, art books, art theory, art techniques Book 2)

[Dmca](#)