

An Introduction To Mathematics For Computing And It Practitioners

[#mathematics for computing](#) [#math for IT practitioners](#) [#foundational math for tech](#) [#applied math for IT](#) [#computer science mathematics](#)

This comprehensive introduction explores essential mathematical concepts specifically tailored for professionals in the computing and IT fields. Gain the foundational knowledge and analytical skills crucial for understanding algorithms, data structures, cybersecurity, and more, empowering you to excel in your technical career.

We collaborate with global institutions to share verified journal publications.

We would like to thank you for your visit.

This website provides the document Mathematics For Tech Professionals you have been searching for.

All visitors are welcome to download it completely free.

The authenticity of the document is guaranteed.

We only provide original content that can be trusted.

This is our way of ensuring visitor satisfaction.

Use this document to support your needs.

We are always ready to offer more useful resources in the future.

Thank you for making our website your choice.

Across digital archives and online libraries, this document is highly demanded.

You are lucky to access it directly from our collection.

Enjoy the full version Mathematics For Tech Professionals, available at no cost.

An Introduction to Mathematics for Computing and IT Practitioners

There are a number of topics within mathematics which have a direct relevance to computing and IT. Some of these topics form the basis of how a computer processes data, some are used at a higher level to enable a program to perform a required function, and others are algorithmic and can be easily implemented within a computer program. This book attempts to bring together many of these mathematical concepts and present them in a way that is relevant to those studying ICT and computing qualifications, and to those who would like to start to explore this subject for themselves. This book not only demonstrates how to perform the relevant calculations, but puts the topics into the context of computing. Explanations include simple diagrams, tables of data, worked examples and questions with worked answers to allow the reader to check their understanding of a topic. Some of the explanations include the use of algorithms and flowcharts as well as coded examples using JavaScript. Additionally, throughout this book there are examples to illustrate how the mathematical concepts are used within computing and IT. Please note that this book is purely an introduction to this subject and many of the concepts are described in their simplest form. The reader may then wish to go and explore a specific topic in more depth. The following is a list of the topics which are covered by the book:- Algorithms: Steps and Flowcharts.- Matrices: Operations, Transformations, Simultaneous Equations, Maps and Graphs.- Sequences and Series: Arithmetic, Geometric, Recursive Algorithms, Fibonacci Sequence, Golden Ratio.- Searching and Sorting Algorithms: Bubble Sort, Quicksort and Binary Search.- Probability: Tree Diagrams, Space Diagrams, Venn Diagrams and Simulation.- Number Systems: Binary, Octal, Hexadecimal, Conversions between number systems, Operations on Binary, Gray Codes.- Boolean Algebra: Logical Operators, Venn Diagrams, Boolean Expressions and Logic Gates.- Character Codes: ASCII, Unicode, UTF-8, Hamming Codes, MIME Base 64.- IP Addresses: IPv4, Network and Host IDs, CIDR Notation and Subnetting.

C++ for Mathematicians

For problems that require extensive computation, a C++ program can race through billions of examples faster than most other computing choices. C++ enables mathematicians of virtually any discipline to create programs to meet their needs quickly, and is available on most computer systems at no cost. C++ for Mathematicians: An Introduction for Students and Professionals accentuates C++ concepts that are most valuable for pure and applied mathematical research. This is the first book available on C++ programming that is written specifically for a mathematical audience; it omits the language's more obscure features in favor of the aspects of greatest utility for mathematical work. The author explains how to use C++ to formulate conjectures, create images and diagrams, verify proofs, build mathematical structures, and explore myriad examples. Emphasizing the essential role of practice as part of the learning process, the book is ideally designed for undergraduate coursework as well as self-study. Each chapter provides many problems and solutions which complement the text and enable you to learn quickly how to apply them to your own problems. Accompanying downloadable resources provide all numbered programs so that readers can easily use or adapt the code as needed. Presenting clear explanations and examples from the world of mathematics that develop concepts from the ground up, C++ for Mathematicians can be used again and again as a resource for applying C++ to problems that range from the basic to the complex.

Introduction to the Tools of Scientific Computing

The book provides an introduction to common programming tools and methods in numerical mathematics and scientific computing. Unlike widely used standard approaches, it does not focus on any particular language but aims to explain the key underlying concepts. In general, new concepts are first introduced in the particularly user-friendly Python language and then transferred and expanded in various scientific programming environments from C / C ++, Julia and MATLAB to Maple. This includes different approaches to distributed computing. The fact that different languages are studied and compared also makes the book useful for mathematicians and practitioners trying to decide which programming language to use for which purposes.

Mathematics for Computer Science

This book covers elementary discrete mathematics for computer science and engineering. It emphasizes mathematical definitions and proofs as well as applicable methods. Topics include formal logic notation, proof methods; induction, well-ordering; sets, relations; elementary graph theory; integer congruences; asymptotic notation and growth of functions; permutations and combinations, counting principles; discrete probability. Further selected topics may also be covered, such as recursive definition and structural induction; state machines and invariants; recurrences; generating functions.

Mathematical Tools for Real-World Applications

Techniques for applying mathematical concepts in the real world: six rarely taught but crucial tools for analysis, research, and problem-solving. Many young graduates leave school with a solid knowledge of mathematical concepts but struggle to apply these concepts in practice. Real scientific and engineering problems are different from those found in textbooks: they are messier, take longer to solve, and standard solution recipes might not apply. This book fills the gap between what is taught in the typical college curriculum and what a practicing engineer or scientist needs to know. It presents six powerful tools for analysis, research, and problem-solving in the real world: dimensional analysis, limiting cases, symmetry, scaling, making order of magnitude estimates, and the method of successive approximations. The book does not focus on formulaic manipulations of equations, but emphasizes analysis and explores connections between the equations and the application. Each chapter introduces a set of ideas and techniques and then shows how these techniques apply to a series of problems. (Knowledge of algebra and trigonometry, but not calculus, is required.) The final two chapters tie all six techniques together and apply them to two real-world problems: computing the probability of a rare, catastrophic event, and tracking a satellite with a GPS receiver. Readers will learn how to analyze, dissect, and gain insight into the results by using all the techniques presented in earlier chapters—and discover how analysis tools work on problems not concocted for a textbook. The appendix provides solutions to many of the problems found throughout the book. Alexandr Draganov was born and raised in Kyiv, Ukraine; in light of the current war in Ukraine he will donate 100% of his royalties for the first year to support medical and humanitarian efforts there.

Modeling Time in Computing

Models that include a notion of time are ubiquitous in disciplines such as the natural sciences, engineering, philosophy, and linguistics, but in computing the abstractions provided by the traditional models are problematic and the discipline has spawned many novel models. This book is a systematic thorough presentation of the results of several decades of research on developing, analyzing, and applying time models to computing and engineering. After an opening motivation introducing the topics, structure and goals, the authors introduce the notions of formalism and model in general terms along with some of their fundamental classification criteria. In doing so they present the fundamentals of propositional and predicate logic, and essential issues that arise when modeling time across all types of system. Part I is a summary of the models that are traditional in engineering and the natural sciences, including fundamental computer science: dynamical systems and control theory; hardware design; and software algorithmic and complexity analysis. Part II covers advanced and specialized formalisms dealing with time modeling in heterogeneous software-intensive systems: formalisms that share finite state machines as common “ancestors”; Petri nets in many variants; notations based on mathematical logic, such as temporal logic; process algebras; and “dual-language approaches” combining two notations with different characteristics to model and verify complex systems, e.g., model-checking frameworks. Finally, the book concludes with summarizing remarks and hints towards future developments and open challenges. The presentation uses a rigorous, yet not overly technical, style, appropriate for readers with heterogeneous backgrounds, and each chapter is supplemented with detailed bibliographic remarks and carefully chosen exercises of varying difficulty and scope. The book is aimed at graduate students and researchers in computer science, while researchers and practitioners in other scientific and engineering disciplines interested in time modeling with a computational flavor will also find the book of value, and the comparative and conceptual approach makes this a valuable introduction for non-experts. The authors assume a basic knowledge of calculus, probability theory, algorithms, and programming, while a more advanced knowledge of automata, formal languages, and mathematical logic is useful.

Understand Mathematics, Understand Computing

In this book the authors aim to endow the reader with an operational, conceptual, and methodological understanding of the discrete mathematics that can be used to study, understand, and perform computing. They want the reader to understand the elements of computing, rather than just know them. The basic topics are presented in a way that encourages readers to develop their personal way of thinking about mathematics. Many topics are developed at several levels, in a single voice, with sample applications from within the world of computing. Extensive historical and cultural asides emphasize the human side of mathematics and mathematicians. By means of lessons and exercises on “doing” mathematics, the book prepares interested readers to develop new concepts and invent new techniques and technologies that will enhance all aspects of computing. The book will be of value to students, scientists, and engineers engaged in the design and use of computing systems, and to scholars and practitioners beyond these technical fields who want to learn and apply novel computational ideas.

Mathematical Analysis and the Mathematics of Computation

This book is a comprehensive, unifying introduction to the field of mathematical analysis and the mathematics of computing. It develops the relevant theory at a modern level and it directly relates modern mathematical ideas to their diverse applications. The authors develop the whole theory. Starting with a simple axiom system for the real numbers, they then lay the foundations, developing the theory, exemplifying where it's applicable, in turn motivating further development of the theory. They progress from sets, structures, and numbers to metric spaces, continuous functions in metric spaces, linear normed spaces and linear mappings; and then differential calculus and its applications, the integral calculus, the gamma function, and linear integral operators. They then present important aspects of approximation theory, including numerical integration. The remaining parts of the book are devoted to ordinary differential equations, the discretization of operator equations, and numerical solutions of ordinary differential equations. This textbook contains many exercises of varying degrees of difficulty, suitable for self-study, and at the end of each chapter the authors present more advanced problems that shed light on interesting features, suitable for classroom seminars or study groups. It will be valuable for undergraduate and graduate students in mathematics, computer science, and related fields such as engineering. This is a rich field that has experienced enormous development in recent decades,

and the book will also act as a reference for graduate students and practitioners who require a deeper understanding of the methodologies, techniques, and foundations.

Aggregation Functions: A Guide for Practitioners

A broad introduction to the topic of aggregation functions is to be found in this book. It also provides a concise account of the properties and the main classes of such functions. Some state-of-the-art techniques are presented, along with many graphical illustrations and new interpolatory aggregation functions. Particular attention is paid to identification and construction of aggregation functions from application specific requirements and empirical data.

An Introduction to Modern Mathematical Computing

Thirty years ago mathematical, as opposed to applied numerical, computation was difficult to perform and so relatively little used. Three threads changed that: the emergence of the personal computer; the discovery of fiber-optics and the consequent development of the modern internet; and the building of the Three "M's" Maple, Mathematica and Matlab. We intend to persuade that Maple and other like tools are worth knowing assuming only that one wishes to be a mathematician, a mathematics educator, a computer scientist, an engineer or scientist, or anyone else who wishes/needs to use mathematics better. We also hope to explain how to become an 'experimental mathematician' while learning to be better at proving things. To accomplish this our material is divided into three main chapters followed by a postscript. These cover elementary number theory, calculus of one and several variables, introductory linear algebra, and visualization and interactive geometric computation.

Probability with R

Provides a comprehensive introduction to probability with an emphasis on computing-related applications. This self-contained new and extended edition outlines a first course in probability applied to computer-related disciplines. As in the first edition, experimentation and simulation are favoured over mathematical proofs. The freely down-loadable statistical programming language R is used throughout the text, not only as a tool for calculation and data analysis, but also to illustrate concepts of probability and to simulate distributions. The examples in *Probability with R: An Introduction with Computer Science Applications, Second Edition* cover a wide range of computer science applications, including: testing program performance; measuring response time and CPU time; estimating the reliability of components and systems; evaluating algorithms and queuing systems. Chapters cover: The R language; summarizing statistical data; graphical displays; the fundamentals of probability; reliability; discrete and continuous distributions; and more. This second edition includes: improved R code throughout the text, as well as new procedures, packages and interfaces; updated and additional examples, exercises and projects covering recent developments of computing; an introduction to bivariate discrete distributions together with the R functions used to handle large matrices of conditional probabilities, which are often needed in machine translation; an introduction to linear regression with particular emphasis on its application to machine learning using testing and training data; a new section on spam filtering using Bayes theorem to develop the filters; an extended range of Poisson applications such as network failures, website hits, virus attacks and accessing the cloud; use of new allocation functions in R to deal with hash table collision, server overload and the general allocation problem. The book is supplemented with a Wiley Book Companion Site featuring data and solutions to exercises within the book. Primarily addressed to students of computer science and related areas, *Probability with R: An Introduction with Computer Science Applications, Second Edition* is also an excellent text for students of engineering and the general sciences. Computing professionals who need to understand the relevance of probability in their areas of practice will find it useful.

Introduction to Evolutionary Computing

The overall structure of this new edition is three-tier: Part I presents the basics, Part II is concerned with methodological issues, and Part III discusses advanced topics. In the second edition the authors have reorganized the material to focus on problems, how to represent them, and then how to choose and design algorithms for different representations. They also added a chapter on problems, reflecting the overall book focus on problem-solvers, a chapter on parameter tuning, which they combined with the parameter control and "how-to" chapters into a methodological part, and finally a chapter on evolutionary robotics with an outlook on possible exciting developments in this field. The book is suitable for undergraduate and graduate courses in artificial intelligence and computational intelligence, and

for self-study by practitioners and researchers engaged with all aspects of bioinspired design and optimization.

Introduction to Computation and Modeling for Differential Equations

Uses mathematical, numerical, and programming tools to solve differential equations for physical phenomena and engineering problems. Introduction to Computation and Modeling for Differential Equations, Second Edition features the essential principles and applications of problem solving across disciplines such as engineering, physics, and chemistry. The Second Edition integrates the science of solving differential equations with mathematical, numerical, and programming tools, specifically with methods involving ordinary differential equations; numerical methods for initial value problems (IVPs); numerical methods for boundary value problems (BVPs); partial differential equations (PDEs); numerical methods for parabolic, elliptic, and hyperbolic PDEs; mathematical modeling with differential equations; numerical solutions; and finite difference and finite element methods. The author features a unique “Five-M” approach: Modeling, Mathematics, Methods, MATLAB®, and Multiphysics, which facilitates a thorough understanding of how models are created and preprocessed mathematically with scaling, classification, and approximation and also demonstrates how a problem is solved numerically using the appropriate mathematical methods. With numerous real-world examples to aid in the visualization of the solutions, Introduction to Computation and Modeling for Differential Equations, Second Edition includes: New sections on topics including variational formulation, the finite element method, examples of discretization, ansatz methods such as Galerkin’s method for BVPs, parabolic and elliptic PDEs, and finite volume methods. Numerous practical examples with applications in mechanics, fluid dynamics, solid mechanics, chemical engineering, heat conduction, electromagnetic field theory, and control theory, some of which are solved with computer programs MATLAB and COMSOL Multiphysics®. Additional exercises that introduce new methods, projects, and problems to further illustrate possible applications. A related website with select solutions to the exercises, as well as the MATLAB data sets for ordinary differential equations (ODEs) and PDEs. Introduction to Computation and Modeling for Differential Equations, Second Edition is a useful textbook for upper-undergraduate and graduate-level courses in scientific computing, differential equations, ordinary differential equations, partial differential equations, and numerical methods. The book is also an excellent self-study guide for mathematics, science, computer science, physics, and engineering students, as well as an excellent reference for practitioners and consultants who use differential equations and numerical methods in everyday situations.

C++ for Mathematicians

A thorough exposition of quantum computing and the underlying concepts of quantum physics, with explanations of the relevant mathematics and numerous examples. The combination of two of the twentieth century’s most influential and revolutionary scientific theories, information theory and quantum mechanics, gave rise to a radically new view of computing and information. Quantum information processing explores the implications of using quantum mechanics instead of classical mechanics to model information and its processing. Quantum computing is not about changing the physical substrate on which computation is done from classical to quantum but about changing the notion of computation itself, at the most basic level. The fundamental unit of computation is no longer the bit but the quantum bit or qubit. This comprehensive introduction to the field offers a thorough exposition of quantum computing and the underlying concepts of quantum physics, explaining all the relevant mathematics and offering numerous examples. With its careful development of concepts and thorough explanations, the book makes quantum computing accessible to students and professionals in mathematics, computer science, and engineering. A reader with no prior knowledge of quantum physics (but with sufficient knowledge of linear algebra) will be able to gain a fluent understanding by working through the book.

Quantum Computing

Computer Graphics - First Mathematical Steps will help students to master basic Computer Graphics and the mathematical concepts which underlie this subject. They will be led to develop their own skills, and appreciate Computer Graphics techniques in both two and three dimensions. The presentation of the text is methodical, systematic and gently paced - everything translates into numbers and simple ideas. Sometimes students experience difficulty in understanding some of the mathematics in standard Computer Graphics books; this book can serve as a good introduction to more advanced texts. It starts

from first principles and is sympathetically written for those with a limited mathematical background. Computer Graphics - First Mathematical Steps is suitable for supporting undergraduate programmes in Computers and also the newer areas of Computer Graphics and Visualization. It is appropriate for post-graduate conversion courses which develop expertise in Computer Graphics and CAD. It can also be used for enrichment topics for high-flying pre-college students, and for refresher/enhancement courses for computer graphics technicians.

Computer Graphics

This book teaches introductory computer programming using Maple, offering more mathematically oriented exercises and problems than those found in traditional programming courses, while reinforcing and applying concepts and techniques of calculus. Includes case studies.

Mathematical Computing

Computer technology is pervasive in the modern world, its role ever more important as it becomes embedded in a myriad of physical systems and disciplinary ways of thinking. The late Michael Sean Mahoney was a pioneer scholar of the history of computing, one of the first established historians of science to take seriously the challenges and opportunities posed by information technology to our understanding of the twentieth century. Mahoney's work ranged widely, from logic and the theory of computation to the development of software and applications as craft-work. But it was always informed by a unique perspective derived from his distinguished work on the history of medieval mathematics and experimental practice during the Scientific Revolution. His writings offered a new angle on very recent events and ideas and bridged the gaps between academic historians and computer scientists. Indeed, he came to believe that the field was irreducibly pluralistic and that there could be only histories of computing. In this collection, Thomas Haigh presents thirteen of Mahoney's essays and papers organized across three categories: historiography, software engineering, and theoretical computer science. His introduction surveys Mahoney's work to trace the development of key themes, illuminate connections among different areas of his research, and put his contributions into context. The volume also includes an essay on Mahoney by his former students Jed Z. Buchwald and D. Graham Burnett. The result is a landmark work, of interest to computer professionals as well as historians of technology and science.

Histories of Computing

This book investigates some of the difficulties related to scientific computing, describing how these can be overcome.

Accuracy and Reliability in Scientific Computing

"To design future networks that are worthy of society's trust, we must put the 'discipline' of computer networking on a much stronger foundation. This book rises above the considerable minutiae of today's networking technologies to emphasize the long-standing mathematical underpinnings of the field." –Professor Jennifer Rexford, Department of Computer Science, Princeton University "This book is exactly the one I have been waiting for the last couple of years. Recently, I decided most students were already very familiar with the way the net works but were not being taught the fundamentals—the math. This book contains the knowledge for people who will create and understand future communications systems." –Professor Jon Crowcroft, The Computer Laboratory, University of Cambridge

The Essential Mathematical Principles Required to Design, Implement, or Evaluate Advanced Computer Networks

Students, researchers, and professionals in computer networking require a firm conceptual understanding of its foundations. Mathematical Foundations of Computer Networking provides an intuitive yet rigorous introduction to these essential mathematical principles and techniques. Assuming a basic grasp of calculus, this book offers sufficient detail to serve as the only reference many readers will need. Each concept is described in four ways: intuitively; using appropriate mathematical notation; with a numerical example carefully chosen for its relevance to networking; and with a numerical exercise for the reader. The first part of the text presents basic concepts, and the second part introduces four theories in a progression that has been designed to gradually deepen readers' understanding. Within each part, chapters are as self-contained as possible. The first part covers probability; statistics; linear algebra; optimization; and signals, systems, and transforms. Topics range from Bayesian networks to hypothesis testing, and eigenvalue computation to Fourier transforms. These preliminary chapters establish a basis for the four theories covered in the second part of the book: queueing theory, game

theory, control theory, and information theory. The second part also demonstrates how mathematical concepts can be applied to issues such as contention for limited resources, and the optimization of network responsiveness, stability, and throughput.

Mathematical Foundations of Computer Networking

This textbook presents an introduction to the mathematical foundations of software engineering. It presents the rich applications of mathematics in areas such as error-correcting codes, cryptography, the safety and security critical fields, the banking and insurance fields, as well as traditional engineering applications. Topics and features: Addresses core mathematics for critical thinking and problem solving Discusses propositional and predicate logic and various proof techniques to demonstrate the correctness of a logical argument. Examines number theory and its applications to cryptography Considers the underlying mathematics of error-correcting codes Discusses graph theory and its applications to modelling networks Reviews tools to support software engineering mathematics, including automated and interactive theorem provers and model checking Discusses financial software engineering, including simple and compound interest, probability and statistics, and operations research Discusses software reliability and dependability and explains formal methods used to derive a program from its specification Discusses calculus, matrices, vectors, complex numbers, and quaternions, as well as applications to graphics and robotics Includes key learning topics, summaries, and review questions in each chapter, together with a useful glossary This practical and easy-to-follow textbook/reference is ideal for computer science students seeking to learn how mathematics can assist them in building high-quality and reliable software on time and on budget. The text also serves as an excellent self-study primer for software engineers, quality professionals, and software managers.

Mathematical Foundations of Software Engineering

Despite growing interest, basic information on methods and models for mathematically analyzing algorithms has rarely been directly accessible to practitioners, researchers, or students. An Introduction to the Analysis of Algorithms, Second Edition, organizes and presents that knowledge, fully introducing primary techniques and results in the field. Robert Sedgewick and the late Philippe Flajolet have drawn from both classical mathematics and computer science, integrating discrete mathematics, elementary real analysis, combinatorics, algorithms, and data structures. They emphasize the mathematics needed to support scientific studies that can serve as the basis for predicting algorithm performance and for comparing different algorithms on the basis of performance. Techniques covered in the first half of the book include recurrences, generating functions, asymptotics, and analytic combinatorics. Structures studied in the second half of the book include permutations, trees, strings, tries, and mappings. Numerous examples are included throughout to illustrate applications to the analysis of algorithms that are playing a critical role in the evolution of our modern computational infrastructure. Improvements and additions in this new edition include Upgraded figures and code An all-new chapter introducing analytic combinatorics Simplified derivations via analytic combinatorics throughout The book's thorough, self-contained coverage will help readers appreciate the field's challenges, prepare them for advanced results—covered in their monograph Analytic Combinatorics and in Donald Knuth's The Art of Computer Programming books—and provide the background they need to keep abreast of new research. "[Sedgewick and Flajolet] are not only worldwide leaders of the field, they also are masters of exposition. I am sure that every serious computer scientist will find this book rewarding in many ways." —From the Foreword by Donald E. Knuth

An Introduction to the Analysis of Algorithms

In the last few years, courses on parallel computation have been developed and offered in many institutions in the UK, Europe and US as a recognition of the growing significance of this topic in mathematics and computer science. There is a clear need for texts that meet the needs of students and lecturers and this book, based on the author's lecture at ETH Zurich, is an ideal practical student guide to scientific computing on parallel computers working up from a hardware instruction level, to shared memory machines, and finally to distributed memory machines. Aimed at advanced undergraduate and graduate students in applied mathematics, computer science, and engineering, subjects covered include linear algebra, fast Fourier transform, and Monte-Carlo simulations, including examples in C and, in some cases, Fortran. This book is also ideal for practitioners and programmers.

Introduction to Parallel Computing

Introduces professionals and scientists to statistics and machine learning using the programming language R Written by and for practitioners, this book provides an overall introduction to R, focusing on tools and methods commonly used in data science, and placing emphasis on practice and business use. It covers a wide range of topics in a single volume, including big data, databases, statistical machine learning, data wrangling, data visualization, and the reporting of results. The topics covered are all important for someone with a science/math background that is looking to quickly learn several practical technologies to enter or transition to the growing field of data science. The Big R-Book for Professionals: From Data Science to Learning Machines and Reporting with R includes nine parts, starting with an introduction to the subject and followed by an overview of R and elements of statistics. The third part revolves around data, while the fourth focuses on data wrangling. Part 5 teaches readers about exploring data. In Part 6 we learn to build models, Part 7 introduces the reader to the reality in companies, Part 8 covers reports and interactive applications and finally Part 9 introduces the reader to big data and performance computing. It also includes some helpful appendices. Provides a practical guide for non-experts with a focus on business users Contains a unique combination of topics including an introduction to R, machine learning, mathematical models, data wrangling, and reporting Uses a practical tone and integrates multiple topics in a coherent framework Demystifies the hype around machine learning and AI by enabling readers to understand the provided models and program them in R Shows readers how to visualize results in static and interactive reports Supplementary materials includes PDF slides based on the book's content, as well as all the extracted R-code and is available to everyone on a Wiley Book Companion Site The Big R-Book is an excellent guide for science technology, engineering, or mathematics students who wish to make a successful transition from the academic world to the professional. It will also appeal to all young data scientists, quantitative analysts, and analytics professionals, as well as those who make mathematical models.

Classical and Quantum Computing

An Introduction to Parallel Programming is the first undergraduate text to directly address compiling and running parallel programs on the new multi-core and cluster architecture. It explains how to design, debug, and evaluate the performance of distributed and shared-memory programs. The author Peter Pacheco uses a tutorial approach to show students how to develop effective parallel programs with MPI, Pthreads, and OpenMP, starting with small programming examples and building progressively to more challenging ones. The text is written for students in undergraduate parallel programming or parallel computing courses designed for the computer science major or as a service course to other departments; professionals with no background in parallel computing. Takes a tutorial approach, starting with small programming examples and building progressively to more challenging examples Focuses on designing, debugging and evaluating the performance of distributed and shared-memory programs Explains how to develop parallel programs using MPI, Pthreads, and OpenMP programming models

The Big R-Book

Ecological research is becoming increasingly quantitative, yet students often opt out of courses in mathematics and statistics, unwittingly limiting their ability to carry out research in the future. This textbook provides a practical introduction to quantitative ecology for students and practitioners who have realised that they need this opportunity. The text is addressed to readers who haven't used mathematics since school, who were perhaps more confused than enlightened by their undergraduate lectures in statistics and who have never used a computer for much more than word processing and data entry. From this starting point, it slowly but surely instils an understanding of mathematics, statistics and programming, sufficient for initiating research in ecology. The book's practical value is enhanced by extensive use of biological examples and the computer language R for graphics, programming and data analysis. Key Features: Provides a complete introduction to mathematics statistics and computing for ecologists. Presents a wealth of ecological examples demonstrating the applied relevance of abstract mathematical concepts, showing how a little technique can go a long way in answering interesting ecological questions. Covers elementary topics, including the rules of algebra, logarithms, geometry, calculus, descriptive statistics, probability, hypothesis testing and linear regression. Explores more advanced topics including fractals, non-linear dynamical systems, likelihood and Bayesian estimation, generalised linear, mixed and additive models, and multivariate statistics. R boxes provide step-by-step recipes for implementing the graphical and numerical techniques outlined in each section. How to be a Quantitative Ecologist provides a comprehensive introduction to mathematics, statistics and computing and is the ideal textbook for late undergraduate and postgraduate courses in environmental biology.

"With a book like this, there is no excuse for people to be afraid of maths, and to be ignorant of what it can do." —Professor Tim Benton, Faculty of Biological Sciences, University of Leeds, UK

An Introduction to Parallel Programming

An innovative introduction to chemical engineering computing As chemical engineering technology advances, so does the complexity of the problems that arise. The problems that chemical engineers and chemical engineering students face today can no longer be answered with programs written on a case-by-case basis. Introduction to Chemical Engineering Computing teaches professionals and students the kinds of problems they will have to solve, the types of computer programs needed to solve these problems, and how to ensure that the problems have been solved correctly. Each chapter in Introduction to Chemical Engineering Computing contains a description of the physical problem in general terms and in a mathematical context, thorough step-by-step instructions, numerous examples, and comprehensive explanations for each problem and program. This indispensable text features Excel, MATLAB(r), Aspen PlusTM, and FEMLAB programs and acquaints readers with the advantages of each. Perfect for students and professionals, Introduction to Chemical Engineering Computing gives readers the professional tools they need to solve real-world problems involving:

- * Equations of state
- * Vapor-liquid and chemical reaction equilibria
- * Mass balances with recycle streams
- * Mass transfer equipment
- * Process simulation
- * Chemical reactors
- * Transfer processes in 1D
- * Fluid flow in 2D and 3D
- * Convective diffusion equations in 2D and 3D

How to be a Quantitative Ecologist

This book keeps an eye in the direction of applications of advanced and high performance scientific computing in describing the behavior of natural and constructed systems, e.g. chaos, bifurcation, fractal, Lyapunov exponent, period doubling, Poincaré map, strange attractor etc. With the aid of powerful computers the modern theory of chaos and its geometry, the fractals, and attractors are developed. The concepts of object oriented computing are introduced early in the text and steadily expanded as one progresses through the chapters. The beginning of each chapter is of an introductory nature, followed by practical applications, the discussion of numerical results, theoretical investigations on nonlinear stability and convergence. This is the first complete introduction to process modelling and computing that fully integrates software tools — enabling professionals and students to master critical techniques hands on through computer simulations based on the popular MATLAB environment. The book offers a simple tool for all those oscillations that are travelling through the world, helping them discover its hidden beauty. Many applications as well as results of computer simulations are presented. The center of concern is set on existing as well as emerging continuous methods of investigations useful for researchers, engineers and practitioners active in many and often interdisciplinary fields, where physics, electrochemistry, biology and medicine play a key role. Coverage includes:

- Dynamic behavior of nonlinear systems,
- Fundamental descriptions of processes exhibiting nonlinear oscillations,
- Mechanism and function of structures of nonlinear oscillations' patterns,
- Analysis of dynamical oscillations in electric circuits and systems,
- Artificial intelligence models of natural systems,
- Nonlinear oscillations in chemistry, biology and medicine,
- Oscillations in mechanics and transport systems,
- Oscillations in fractional-order systems,
- Energy harvesting systems from the surrounding environment.

With an insatiable appetite for exploring the surrounding world and doing research, this book can help readers quickly find ways to use new computers and facilitate the quest for greater knowledge and understanding of reality. The reach of novelty of the book ranges from new mathematical ideas to motivating questions and science issues in many subject areas.

Introduction to Chemical Engineering Computing

An introduction to the methods of designing algorithms for hard computing tasks, concentrating mainly on approximate, randomized, and heuristic algorithms, and on the theoretical and experimental comparison of these approaches according to the requirements of the practice. This is the first book to systematically explain and compare all the main possibilities of attacking hard computing problems. It also closes the gap between theory and practice by providing at once a graduate textbook and a handbook for practitioners dealing with hard computing problems.

Mathematical Modelling and Computing in Physics, Chemistry and Biology

Thirty years ago mathematical, as opposed to applied numerical, computation was difficult to perform and so relatively little used. Three threads changed that: the emergence of the personal computer; the

discovery of fiber-optics and the consequent development of the modern internet; and the building of the Three "M's" Maple, Mathematica and Matlab. We intend to persuade that Mathematica and other similar tools are worth knowing, assuming only that one wishes to be a mathematician, a mathematics educator, a computer scientist, an engineer or scientist, or anyone else who wishes/needs to use mathematics better. We also hope to explain how to become an "experimental mathematician" while learning to be better at proving things. To accomplish this our material is divided into three main chapters followed by a postscript. These cover elementary number theory, calculus of one and several variables, introductory linear algebra, and visualization and interactive geometric computation.

Algorithmics for Hard Problems

Designed for a one-semester course, Introduction to Numerical Analysis and Scientific Computing presents fundamental concepts of numerical mathematics and explains how to implement and program numerical methods. The classroom-tested text helps students understand floating point number representations, particularly those pertaining to IEEE simple and double-precision standards as used in scientific computer environments such as MATLAB® version 7. Drawing on their years of teaching students in mathematics, engineering, and the sciences, the authors discuss computer arithmetic as a source for generating round-off errors and how to avoid the use of algebraic expression that may lead to loss of significant figures. They cover nonlinear equations, linear algebra concepts, the Lagrange interpolation theorem, numerical differentiation and integration, and ODEs. They also focus on the implementation of the algorithms using MATLAB®. Each chapter ends with a large number of exercises, with answers to odd-numbered exercises provided at the end of the book. Throughout the seven chapters, several computer projects are proposed. These test the students' understanding of both the mathematics of numerical methods and the art of computer programming.

An Introduction to Modern Mathematical Computing

This book provides a practical and comprehensive introduction to computational problem solving from the viewpoints of practitioners in both academic and industrial worlds. The authors present scientific problem-solving using computation and aim to increase computational thinking, which is the mindset and skillset required to solve scientific problems with computational methodologies via model building, simulation, data analysis, and visualization using the Python programming language. Topics and examples span fundamental areas of physical science as well as contemporary topics including quantum computing, neural networks, machine learning, global warming, and energy balance. The book features unique and innovative techniques and practices including: intentional scaffolding to help beginners learn computational problem solving; multimodal computing environments including cloud-based platforms and just-in-time computing; emphasis and connection between both numerical and symbolic computations; and extensive exercise sets carefully designed for further exploration as project assignments or self-paced study. The book is suitable for introductory level readers in physical sciences, engineering, and related STEM disciplines. Specifically, the book is appropriate for use in either a standalone course on computation and modeling and as a resource for readers interested in learning about proven techniques in interactive computing.

Introduction to Numerical Analysis and Scientific Computing

This book touches on an area seldom explored: the mathematical underpinnings of the relational database. The topic is important, but far too often ignored. This is the first book to explain the underlying math in a way that's accessible to database professionals. Just as importantly, if not more so, this book goes beyond the abstract by showing readers how to apply that math in ways that will make them more productive in their jobs. What's in this book will "open the eyes" of most readers to the great power, elegance, and simplicity inherent in relational database technology.

Introduction to Computation in Physical Sciences

This monograph uses the Julia language to guide the reader through an exploration of the fundamental concepts of probability and statistics, all with a view of mastering machine learning, data science, and artificial intelligence. The text does not require any prior statistical knowledge and only assumes a basic understanding of programming and mathematical notation. It is accessible to practitioners and researchers in data science, machine learning, bio-statistics, finance, or engineering who may wish to solidify their knowledge of probability and statistics. The book progresses through ten independent chapters starting with an introduction of Julia, and moving through basic probability, distributions, statis-

tical inference, regression analysis, machine learning methods, and the use of Monte Carlo simulation for dynamic stochastic models. Ultimately this text introduces the Julia programming language as a computational tool, uniquely addressing end-users rather than developers. It makes heavy use of over 200 code examples to illustrate dozens of key statistical concepts. The Julia code, written in a simple format with parameters that can be easily modified, is also available for download from the book's associated GitHub repository online. See what co-creators of the Julia language are saying about the book: Professor Alan Edelman, MIT: With "Statistics with Julia", Yoni and Hayden have written an easy to read, well organized, modern introduction to statistics. The code may be looked at, and understood on the static pages of a book, or even better, when running live on a computer. Everything you need is here in one nicely written self-contained reference. Dr. Viral Shah, CEO of Julia Computing: Yoni and Hayden provide a modern way to learn statistics with the Julia programming language. This book has been perfected through iteration over several semesters in the classroom. It prepares the reader with two complementary skills - statistical reasoning with hands on experience and working with large datasets through training in Julia.

Applied Mathematics for Database Professionals

This textbook presents a concise introduction to the fundamental principles of software engineering, together with practical guidance on how to apply the theory in a real-world, industrial environment. The wide-ranging coverage encompasses all areas of software design, management, and quality. Topics and features: presents a broad overview of software engineering, including software lifecycles and phases in software development, and project management for software engineering; examines the areas of requirements engineering, software configuration management, software inspections, software testing, software quality assurance, and process quality; covers topics on software metrics and problem solving, software reliability and dependability, and software design and development, including Agile approaches; explains formal methods, a set of mathematical techniques to specify and derive a program from its specification, introducing the Z specification language; discusses software process improvement, describing the CMMI model, and introduces UML, a visual modelling language for software systems; reviews a range of tools to support various activities in software engineering, and offers advice on the selection and management of a software supplier; describes such innovations in the field of software as distributed systems, service-oriented architecture, software as a service, cloud computing, and embedded systems; includes key learning topics, summaries and review questions in each chapter, together with a useful glossary. This practical and easy-to-follow textbook/reference is ideal for computer science students seeking to learn how to build high quality and reliable software on time and on budget. The text also serves as a self-study primer for software engineers, quality professionals, and software managers.

Statistics with Julia

This is the second volume in a series of innovative proceedings entirely devoted to the connections between mathematics and computer science. Here mathematics and computer science are directly confronted and joined to tackle intricate problems in computer science with deep and innovative mathematical approaches. The book serves as an outstanding tool and a main information source for a large public in applied mathematics, discrete mathematics and computer science, including researchers, teachers, graduate students and engineers. It provides an overview of the current questions in computer science and the related modern and powerful mathematical methods. The range of applications is very wide and reaches beyond computer science.

Concise Guide to Software Engineering

Distributed computing is at the heart of many applications. It arises as soon as one has to solve a problem in terms of entities -- such as processes, peers, processors, nodes, or agents -- that individually have only a partial knowledge of the many input parameters associated with the problem. In particular each entity cooperating towards the common goal cannot have an instantaneous knowledge of the current state of the other entities. Whereas parallel computing is mainly concerned with 'efficiency', and real-time computing is mainly concerned with 'on-time computing', distributed computing is mainly concerned with 'mastering uncertainty' created by issues such as the multiplicity of control flows, asynchronous communication, unstable behaviors, mobility, and dynamicity. While some distributed algorithms consist of a few lines only, their behavior can be difficult to understand and their properties hard to state and prove. The aim of this book is to present in a comprehensive way the basic

notions, concepts, and algorithms of distributed computing when the distributed entities cooperate by sending and receiving messages on top of an asynchronous network. The book is composed of seventeen chapters structured into six parts: distributed graph algorithms, in particular what makes them different from sequential or parallel algorithms; logical time and global states, the core of the book; mutual exclusion and resource allocation; high-level communication abstractions; distributed detection of properties; and distributed shared memory. The author establishes clear objectives per chapter and the content is supported throughout with illustrative examples, summaries, exercises, and annotated bibliographies. This book constitutes an introduction to distributed computing and is suitable for advanced undergraduate students or graduate students in computer science and computer engineering, graduate students in mathematics interested in distributed computing, and practitioners and engineers involved in the design and implementation of distributed applications. The reader should have a basic knowledge of algorithms and operating systems.

Mathematics and Computer Science II

This book provides a good introduction to the classical elementary number theory and the modern algorithmic number theory, and their applications in computing and information technology, including computer systems design, cryptography and network security. In this second edition proofs of many theorems have been provided, further additions and corrections were made.

Distributed Algorithms for Message-Passing Systems

This is a concise and informal introductory book on the mathematical concepts that underpin computer graphics. The author, John Vince, makes the concepts easy to understand, enabling non-experts to come to terms with computer animation work. The book complements the author's other works and is written in the same accessible and easy-to-read style. It is also a useful reference book for programmers working in the field of computer graphics, virtual reality, computer animation, as well as students on digital media courses, and even mathematics courses.

Number Theory for Computing

Mathematics for Computer Graphics