
Data Structure Book

Codeless Data Structures and Algorithms
Mastering Algorithms with C
Advanced Algorithms and Data Structures
Purely Functional Data Structures
Data Structures and Algorithms in Java
The Design of Dynamic Data Structures
Data Structure Techniques
R Data Structures and Algorithms
Handbook of Data Structures and Applications
Programming Challenges
Advanced Data Structures
Data Structures and Algorithm Analysis in C+
CLASSIC DATA STRUCTURES, 2nd ed.
Compact Data Structures
Data Structures and Algorithms in C++
Data Structures and Algorithms in Python
Guide to Data Structures
Learning Functional Data Structures and Algorithms
Data Structures and Algorithms Using Java
Data Structures and Algorithms
Data Structure and Algorithms Using C++
Advanced Data Structures
Introduction to Data Structures in C
Introduction To Algorithms
Advanced Data Structures
An Introduction to Data Structures and Algorithms

Think Data Structures
Hands-On Data Structures and Algorithms with Rust
Open Data Structures
Data Structures and Algorithms with JavaScript
Algorithms in a Nutshell
The Algorithm Design Manual
A Common-Sense Guide to Data Structures and Algorithms
Data Structures & Algorithms in Python
JavaScript Data Structures and Algorithms
Expert Data Structure with C
Data Structures and Algorithms: A First Course
Data Structures & Their Algorithms
A Common-Sense Guide to Data Structures and Algorithms, Second Edition
Problem Solving with Algorithms and Data Structures Using Python

Data Structure Book

Downloaded from ftp.bonide.com by
guest

YOUNG SANTANA

Codeless Data Structures and Algorithms KHANNA PUBLISHING
HOUSE

Based on the authors' market leading data structures books in Java and C++, this textbook offers a comprehensive, definitive introduction to data structures in Python by authoritative authors. Data Structures and Algorithms in Python is the first authoritative object-oriented book available for the Python data structures course. Designed to provide a comprehensive introduction to data structures and algorithms, including their design, analysis, and implementation, the text will maintain the same general

structure as Data Structures and Algorithms in Java and Data Structures and Algorithms in C++.

Mastering Algorithms with C Pearson Education India

Data Structures & Theory of Computation

Advanced Algorithms and Data Structures Wiley Global
Education

An extensively revised edition of a mathematically rigorous yet accessible introduction to algorithms.

Purely Functional Data Structures Apress

In this second edition of his successful book, experienced teacher and author Mark Allen Weiss continues to refine and enhance his innovative approach to algorithms and data structures. Written for the advanced data structures course, this text highlights theoretical topics such as abstract data types and the efficiency

of algorithms, as well as performance and running time. Before covering algorithms and data structures, the author provides a brief introduction to C++ for programmers unfamiliar with the language. Dr Weiss's clear writing style, logical organization of topics, and extensive use of figures and examples to demonstrate the successive stages of an algorithm make this an accessible, valuable text. New to this Edition *An appendix on the Standard Template Library (STL) *C++ code, tested on multiple platforms, that conforms to the ANSI ISO final draft standard 0201361221B04062001

Data Structures and Algorithms in Java PHI Learning Pvt. Ltd. Everyone knows that programming plays a vital role as a solution to automate and execute a task in a proper manner. Irrespective of mathematical problems, the skills of programming are necessary to solve any type of problems that may be correlated to solve real life problems efficiently and effectively. This book is intended to flow from the basic concepts of C++ to technicalities of the programming language, its approach and debugging. The chapters of the book flow with the formulation of the problem, it's designing, finding the step-by-step solution procedure along with its compilation, debugging and execution with the output. Keeping in mind the learner's sentiments and requirements, the exemplary programs are narrated with a simple approach so that it can lead to creation of good programs that not only executes properly to give the output, but also enables the learners to incorporate programming skills in them. The style of writing a program using a programming language is also emphasized by introducing the inclusion of comments wherever necessary to encourage writing more readable and well commented programs.

As practice makes perfect, each chapter is also enriched with practice exercise questions so as to build the confidence of writing the programs for learners. The book is a complete and all-inclusive handbook of C++ that covers all that a learner as a beginner would expect, as well as complete enough to go ahead with advanced programming. This book will provide a fundamental idea about the concepts of data structures and associated algorithms. By going through the book, the reader will be able to understand about the different types of algorithms and at which situation and what type of algorithms will be applicable.

The Design of Dynamic Data Structures Pragmatic Bookshelf Explore data structures and algorithm concepts and their relation to everyday JavaScript development. A basic understanding of these ideas is essential to any JavaScript developer wishing to analyze and build great software solutions. You'll discover how to implement data structures such as hash tables, linked lists, stacks, queues, trees, and graphs. You'll also learn how a URL shortener, such as bit.ly, is developed and what is happening to the data as a PDF is uploaded to a webpage. This book covers the practical applications of data structures and algorithms to encryption, searching, sorting, and pattern matching. It is crucial for JavaScript developers to understand how data structures work and how to design algorithms. This book and the accompanying code provide that essential foundation for doing so. With JavaScript Data Structures and Algorithms you can start developing your knowledge and applying it to your JavaScript projects today. What You'll Learn Review core data structure fundamentals: arrays, linked-lists, trees, heaps, graphs, and hash-table Review core algorithm fundamentals: search, sort,

recursion, breadth/depth first search, dynamic programming, bitwise operators Examine how the core data structure and algorithms knowledge fits into context of JavaScript explained using prototypical inheritance and native JavaScript objects/data types Take a high-level look at commonly used design patterns in JavaScript Who This Book Is For Existing web developers and software engineers seeking to develop or revisit their fundamental data structures knowledge; beginners and students studying JavaScript independently or via a course or coding bootcamp.

Data Structure Techniques Franklin Beedle & Associates Increase speed and performance of your applications with efficient data structures and algorithms About This Book See how to use data structures such as arrays, stacks, trees, lists, and graphs through real-world examples Find out about important and advanced data structures such as searching and sorting algorithms Understand important concepts such as big-o notation, dynamic programming, and functional data structured Who This Book Is For This book is for R developers who want to use data structures efficiently. Basic knowledge of R is expected. What You Will Learn Understand the rationality behind data structures and algorithms Understand computation evaluation of a program featuring asymptotic and empirical algorithm analysis Get to know the fundamentals of arrays and linked-based data structures Analyze types of sorting algorithms Search algorithms along with hashing Understand linear and tree-based indexing Be able to implement a graph including topological sort, shortest path problem, and Prim's algorithm Understand dynamic programming (Knapsack) and randomized algorithms In Detail In

this book, we cover not only classical data structures, but also functional data structures. We begin by answering the fundamental question: why data structures? We then move on to cover the relationship between data structures and algorithms, followed by an analysis and evaluation of algorithms. We introduce the fundamentals of data structures, such as lists, stacks, queues, and dictionaries, using real-world examples. We also cover topics such as indexing, sorting, and searching in depth. Later on, you will be exposed to advanced topics such as graph data structures, dynamic programming, and randomized algorithms. You will come to appreciate the intricacies of high performance and scalable programming using R. We also cover special R data structures such as vectors, data frames, and atomic vectors. With this easy-to-read book, you will be able to understand the power of linked lists, double linked lists, and circular linked lists. We will also explore the application of binary search and will go in depth into sorting algorithms such as bubble sort, selection sort, insertion sort, and merge sort. Style and approach This easy-to-read book with its fast-paced nature will improve the productivity of an R programmer and improve the performance of R applications. It is packed with real-world examples.

R Data Structures and Algorithms Springer Science & Business Media

In the era of self-taught developers and programmers, essential topics in the industry are frequently learned without a formal academic foundation. A solid grasp of data structures and algorithms (DSA) is imperative for anyone looking to do professional software development and engineering, but classes

in the subject can be dry or spend too much time on theory and unnecessary readings. Regardless of your programming language background, *Codeless Data Structures and Algorithms* has you covered. In this book, author Armstrong Subero will help you learn DSAs without writing a single line of code. Straightforward explanations and diagrams give you a confident handle on the topic while ensuring you never have to open your code editor, use a compiler, or look at an integrated development environment. Subero introduces you to linear, tree, and hash data structures and gives you important insights behind the most common algorithms that you can directly apply to your own programs. *Codeless Data Structures and Algorithms* provides you with the knowledge about DSAs that you will need in the professional programming world, without using any complex mathematics or irrelevant information. Whether you are a new developer seeking a basic understanding of the subject or a decision-maker wanting a grasp of algorithms to apply to your projects, this book belongs on your shelf. Quite often, a new, refreshing, and unpretentious approach to a topic is all you need to get inspired. What You'll Learn Understand tree data structures without delving into unnecessary details or going into too much theory Get started learning linear data structures with a basic discussion on computer memory Study an overview of arrays, linked lists, stacks and queues Who This Book Is For This book is for beginners, self-taught developers and programmers, and anyone who wants to understand data structures and algorithms but don't want to wade through unnecessary details about quirks of a programming language or don't have time to sit and read a massive book on the subject. This book is also useful for non-

technical decision-makers who are curious about how algorithms work.

Handbook of Data Structures and Applications Pragmatic Bookshelf

Learn Data Structures and Algorithms! This book is a collection of lectures notes on Data Structures and Algorithms. The content found in this book supplements the free video lecture series, of the same name, "Advanced Data Structures", by the author, Dr. Daniel Page. This video lecture series is available at <http://www.pagewizardgames.com/datastructures>. This book: - Contains Computer Science topics and materials comparable to those found among university courses at a similar level (second-year) at top Canadian universities. - Provides an accessible written companion and supplemental notes for those that wish to learn the subject of Data Structures and Algorithms from the video lecture series, but have difficulties taking notes, or would prefer having a written alternative to follow along. This book is ideal for those with already an introductory programming background, know a little bit about computing, and wish to learn more about Data Structures and Algorithms and begin a more formal study of Computer Science. The materials here are a great place to start for supplemental/additional learning materials on the subject for self-study, university students, or those that want to learn more about Computer Science. Dr. Daniel Page places great emphasis on the introductory mathematical aspects of Computer Science, a natural transition from a basic programming background to thinking a bit more like a computer scientist about Computer Science. This book is not a textbook. The author assumes the reader is familiar with algebra, functions, common finite and

infinite series such as arithmetic series and geometric series, and basic control structures in programming or logic. All the algorithms in this book are described in English, or using Java-like pseudocode. Chapters -Chapter 1 - Introduction: Data Structures, Problems, Input Size, Algorithms, The Search Problem. -Chapter 2 - Intro to Analysis of Algorithms I: Complexity Analysis, Comparing Algorithms, Growth Rate of Functions (Asymptotics), Showing f is $O(g)$, Showing f is not $O(g)$. -Chapter 3 - Intro to Analysis of Algorithms II: Some Properties of O , An Iterative Example, Back to our "Easy" Search Problem. -Chapter 4 - Dictionaries: The Dictionary Problem, Simple Implementations of a Dictionary. -Chapter 5 - Hashing: Hash Function, Hash Code, Separate Chaining, Open Addressing, Revisiting the Load Factor. -Chapter 6 - Trees: Tree ADT, Linked Tree Representation, Tree Property, Computing Height of a Tree, Tree Traversals -Chapter 7 - Priority Queues & Heaps: Priority Queues, Heaps, Array-Based Implementation, Building a Heap, Application: Sorting, Introduction to Amortized Analysis -Chapter 8 - Binary Search Trees: Ordered Dictionary ADT, BST Implementations, Inorder Traversal, Smallest, Get, Put, Remove, Successor. -Chapter 9 - AVL Trees: Height, AVL Trees, Re-Balancing AVL Trees, putAVL, removeAVL, AVL Tree Performance. -Chapter 10 - Graphs: Degrees and the Handshaking Lemma, Complete Graphs, Paths and Cycles, Trees, Forests, Subgraphs, and Connectivity, Graph Representations. -Chapter 11 - Graph Traversals: Depth-First Search (DFS), Path-Finding, Cycle Detection, Counting Vertices, DFS Tree, Breadth-First Search (BFS), Summary. -Chapter 12 - Minimum Spanning Trees: Weighted Graphs, Minimum Spanning Trees & Algorithms, Prim's Algorithm, Heap-Based

Implementation of Prim's Algorithm and More! -Chapter 13 - Shortest Paths: Single-Source Shortest Path Problem, Dijkstra's Algorithm. -Chapter 14 - Multiway Search Trees: Beyond Binary Search Trees, Get, Put, Successor and Remove, (2,4)-Trees, B-Trees.

Programming Challenges Cambridge University Press

Algorithms and data structures are much more than abstract concepts. Mastering them enables you to write code that runs faster and more efficiently, which is particularly important for today's web and mobile apps. Take a practical approach to data structures and algorithms, with techniques and real-world scenarios that you can use in your daily production code, with examples in JavaScript, Python, and Ruby. This new and revised second edition features new chapters on recursion, dynamic programming, and using Big O in your daily work. Use Big O notation to measure and articulate the efficiency of your code, and modify your algorithm to make it faster. Find out how your choice of arrays, linked lists, and hash tables can dramatically affect the code you write. Use recursion to solve tricky problems and create algorithms that run exponentially faster than the alternatives. Dig into advanced data structures such as binary trees and graphs to help scale specialized applications such as social networks and mapping software. You'll even encounter a single keyword that can give your code a turbo boost. Practice your new skills with exercises in every chapter, along with detailed solutions. Use these techniques today to make your code faster and more scalable.

Advanced Data Structures PageWizard Games, Learning & Entertainment

This newly expanded and updated second edition of the best-selling classic continues to take the "mystery" out of designing algorithms, and analyzing their efficacy and efficiency. Expanding on the first edition, the book now serves as the primary textbook of choice for algorithm design courses while maintaining its status as the premier practical reference guide to algorithms for programmers, researchers, and students. The reader-friendly *Algorithm Design Manual* provides straightforward access to combinatorial algorithms technology, stressing design over analysis. The first part, *Techniques*, provides accessible instruction on methods for designing and analyzing computer algorithms. The second part, *Resources*, is intended for browsing and reference, and comprises the catalog of algorithmic resources, implementations and an extensive bibliography. NEW to the second edition:

- Doubles the tutorial material and exercises over the first edition
- Provides full online support for lecturers, and a completely updated and improved website component with lecture slides, audio and video
- Contains a unique catalog identifying the 75 algorithmic problems that arise most often in practice, leading the reader down the right path to solve them
- Includes several NEW "war stories" relating experiences from real-world applications
- Provides up-to-date links leading to the very best algorithm implementations available in C, C++, and Java

Data Structures and Algorithm Analysis in C+ Springer

Learn functional data structures and algorithms for your applications and bring their benefits to your work now About This Book Moving from object-oriented programming to functional programming? This book will help you get started with functional

programming. Easy-to-understand explanations of practical topics will help you get started with functional data structures. Illustrative diagrams to explain the algorithms in detail. Get hands-on practice of Scala to get the most out of functional programming. Who This Book Is For This book is for those who have some experience in functional programming languages. The data structures in this book are primarily written in Scala, however implementing the algorithms in other functional languages should be straight forward. What You Will Learn Learn to think in the functional paradigm Understand common data structures and the associated algorithms, as well as the context in which they are commonly used Take a look at the runtime and space complexities with the O notation See how ADTs are implemented in a functional setting Explore the basic theme of immutability and persistent data structures Find out how the internal algorithms are redesigned to exploit structural sharing, so that the persistent data structures perform well, avoiding needless copying. Get to know functional features like lazy evaluation and recursion used to implement efficient algorithms Gain Scala best practices and idioms In Detail Functional data structures have the power to improve the codebase of an application and improve efficiency. With the advent of functional programming and with powerful functional languages such as Scala, Clojure and Elixir becoming part of important enterprise applications, functional data structures have gained an important place in the developer toolkit. Immutability is a cornerstone of functional programming. Immutable and persistent data structures are thread safe by definition and hence very appealing for writing robust concurrent programs. How do we express

traditional algorithms in functional setting? Won't we end up copying too much? Do we trade performance for versioned data structures? This book attempts to answer these questions by looking at functional implementations of traditional algorithms. It begins with a refresher and consolidation of what functional programming is all about. Next, you'll get to know about Lists, the work horse data type for most functional languages. We show what structural sharing means and how it helps to make immutable data structures efficient and practical. Scala is the primary implementation languages for most of the examples. At times, we also present Clojure snippets to illustrate the underlying fundamental theme. While writing code, we use ADTs (abstract data types). Stacks, Queues, Trees and Graphs are all familiar ADTs. You will see how these ADTs are implemented in a functional setting. We look at implementation techniques like amortization and lazy evaluation to ensure efficiency. By the end of the book, you will be able to write efficient functional data structures and algorithms for your applications. Style and approach Step-by-step topics will help you get started with functional programming. Learn by doing with hands-on code snippets that give you practical experience of the subject. CLASSIC DATA STRUCTURES, 2nd ed. Packt Publishing Ltd

Advanced Data Structures presents a comprehensive look at the ideas, analysis, and implementation details of data structures as a specialized topic in applied algorithms. Data structures are how data is stored within a computer, and how one can go about searching for data within. This text examines efficient ways to search and update sets of numbers, intervals, or strings by various data structures, such as search trees, structures for sets

of intervals or piece-wise constant functions, orthogonal range search structures, heaps, union-find structures, dynamization and persistence of structures, structures for strings, and hash tables. This is the first volume to show data structures as a crucial algorithmic topic, rather than relegating them as trivial material used to illustrate object-oriented programming methodology, filling a void in the ever-increasing computer science market. Numerous code examples in C and more than 500 references make Advanced Data Structures an indispensable text. Numerous code examples in C and more than 500 references make Advanced Data Structures an indispensable text. *Compact Data Structures* Cambridge University Press

" Algorithms and data structures are much more than abstract concepts. Mastering them enables you to write code that runs faster and more efficiently, which is particularly important for today's web and mobile apps. This book takes a practical approach to data structures and algorithms, with techniques and real-world scenarios that you can use in your daily production code. Graphics and examples make these computer science concepts understandable and relevant. You can use these techniques with any language; examples in the book are in JavaScript, Python, and Ruby. Use Big O notation, the primary tool for evaluating algorithms, to measure and articulate the efficiency of your code, and modify your algorithm to make it faster. Find out how your choice of arrays, linked lists, and hash tables can dramatically affect the code you write. Use recursion to solve tricky problems and create algorithms that run exponentially faster than the alternatives. Dig into advanced data structures such as binary trees and graphs to help scale

specialized applications such as social networks and mapping software. You'll even encounter a single keyword that can give your code a turbo boost. Jay Wengrow brings to this book the key teaching practices he developed as a web development bootcamp founder and educator. Use these techniques today to make your code faster and more scalable. "

[Data Structures and Algorithms in C++](#) "O'Reilly Media, Inc."

This book has three key features : fundamental data structures and algorithms; algorithm analysis in terms of Big-O running time in introduced early and applied through; python is used to facilitates the success in using and mastering data structures and algorithms.

Data Structures and Algorithms in Python Cambridge University Press

Advanced Algorithms and Data Structures introduces a collection of algorithms for complex programming challenges in data analysis, machine learning, and graph computing. Summary As a software engineer, you'll encounter countless programming challenges that initially seem confusing, difficult, or even impossible. Don't despair! Many of these "new" problems already have well-established solutions. Advanced Algorithms and Data Structures teaches you powerful approaches to a wide range of tricky coding challenges that you can adapt and apply to your own applications. Providing a balanced blend of classic, advanced, and new algorithms, this practical guide upgrades your programming toolbox with new perspectives and hands-on techniques. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the technology Can you improve the speed and efficiency of your

applications without investing in new hardware? Well, yes, you can: Innovations in algorithms and data structures have led to huge advances in application performance. Pick up this book to discover a collection of advanced algorithms that will make you a more effective developer. About the book Advanced Algorithms and Data Structures introduces a collection of algorithms for complex programming challenges in data analysis, machine learning, and graph computing. You'll discover cutting-edge approaches to a variety of tricky scenarios. You'll even learn to design your own data structures for projects that require a custom solution. What's inside Build on basic data structures you already know Profile your algorithms to speed up application Store and query strings efficiently Distribute clustering algorithms with MapReduce Solve logistics problems using graphs and optimization algorithms About the reader For intermediate programmers. About the author Marcello La Rocca is a research scientist and a full-stack engineer. His focus is on optimization algorithms, genetic algorithms, machine learning, and quantum computing. Table of Contents 1 Introducing data structures PART 1 IMPROVING OVER BASIC DATA STRUCTURES 2 Improving priority queues: d-way heaps 3 Treaps: Using randomization to balance binary search trees 4 Bloom filters: Reducing the memory for tracking content 5 Disjoint sets: Sub-linear time processing 6 Trie, radix trie: Efficient string search 7 Use case: LRU cache PART 2 MULTIDIMENSIONAL QUERIES 8 Nearest neighbors search 9 K-d trees: Multidimensional data indexing 10 Similarity Search Trees: Approximate nearest neighbors search for image retrieval 11 Applications of nearest neighbor search 12 Clustering 13 Parallel clustering: MapReduce and canopy

clustering PART 3 PLANAR GRAPHS AND MINIMUM CROSSING NUMBER 14 An introduction to graphs: Finding paths of minimum distance 15 Graph embeddings and planarity: Drawing graphs with minimal edge intersections 16 Gradient descent: Optimization problems (not just) on graphs 17 Simulated annealing: Optimization beyond local minima 18 Genetic algorithms: Biologically inspired, fast-converging optimization Guide to Data Structures "O'Reilly Media, Inc."

Advanced data structures is a core course in Computer Science which most graduate program in Computer Science, Computer Science and Engineering, and other allied engineering disciplines, offer during the first year or first semester of the curriculum. The objective of this course is to enable students to have the much-needed foundation for advanced technical skill, leading to better problem-solving in their respective disciplines. Although the course is running in almost all the technical universities for decades, major changes in the syllabus have been observed due to the recent paradigm shift of computation which is more focused on huge data and internet-based technologies. Majority of the institute has been redefined their course content of advanced data structure to fit the current need and course material heavily relies on research papers because of nonavailability of the redefined text book advanced data structure. To the best of our knowledge well-known textbook on advanced data structure provides only partial coverage of the syllabus. The book offers comprehensive coverage of the most essential topics, including: Part I details advancements on basic data structures, viz., cuckoo hashing, skip list, tango tree and Fibonacci heaps and index files. Part II details data structures of

different evolving data domains like special data structures, temporal data structures, external memory data structures, distributed and streaming data structures. Part III elucidates the applications of these data structures on different areas of computer science viz, network, www, DBMS, cryptography, graphics to name a few. The concepts and techniques behind each data structure and their applications have been explained. Every chapter includes a variety of Illustrative Problems pertaining to the data structure(s) detailed, a summary of the technical content of the chapter and a list of Review Questions, to reinforce the comprehension of the concepts. The book could be used both as an introductory or an advanced-level textbook for the advanced undergraduate, graduate and research programmes which offer advanced data structures as a core or an elective course. While the book is primarily meant to serve as a course material for use in the classroom, it could be used as a starting point for the beginner researcher of a specific domain. *Learning Functional Data Structures and Algorithms* Springer Science & Business Media

Using only practically useful techniques, this book teaches methods for organizing, reorganizing, exploring, and retrieving data in digital computers, and the mathematical analysis of those techniques. The authors present analyses that are relatively brief and non-technical but illuminate the important performance characteristics of the algorithms. *Data Structures and Their Algorithms* covers algorithms, not the expression of algorithms in the syntax of particular programming languages. The authors have adopted a pseudocode notation that is readily understandable to programmers but has a simple syntax.

Data Structures and Algorithms Using Java John Wiley & Sons

This accessible and engaging textbook/guide provides a concise introduction to data structures and associated algorithms. Emphasis is placed on the fundamentals of data structures, enabling the reader to quickly learn the key concepts, and providing a strong foundation for later studies of more complex topics. The coverage includes discussions on stacks, queues, lists, (using both arrays and links), sorting, and elementary binary trees, heaps, and hashing. This content is also a natural continuation from the material provided in the separate Springer title Guide to Java by the same authors. Topics and features: reviews the preliminary concepts, and introduces stacks and queues using arrays, along with a discussion of array-based lists; examines linked lists, the implementation of stacks and queues using references, binary trees, a range of varied sorting techniques, heaps, and hashing; presents both primitive and generic data types in each chapter, and makes use of contour diagrams to illustrate object-oriented concepts; includes chapter summaries, and asks the reader questions to help them interact with the material; contains numerous examples and illustrations, and one or more complete program in every chapter; provides

exercises at the end of each chapter, as well as solutions to selected exercises, and a glossary of important terms. This clearly-written work is an ideal classroom text for a second semester course in programming using the Java programming language, in preparation for a subsequent advanced course in data structures and algorithms. The book is also eminently suitable as a self-study guide in either academe or industry.

Data Structures and Algorithms John Wiley & Sons

This is an excellent, up-to-date and easy-to-use text on data structures and algorithms that is intended for undergraduates in computer science and information science. The thirteen chapters, written by an international group of experienced teachers, cover the fundamental concepts of algorithms and most of the important data structures as well as the concept of interface design. The book contains many examples and diagrams. Whenever appropriate, program codes are included to facilitate learning. This book is supported by an international group of authors who are experts on data structures and algorithms, through its website at <http://www.cs.pitt.edu/jung/GrowingBook/>, so that both teachers and students can benefit from their expertise