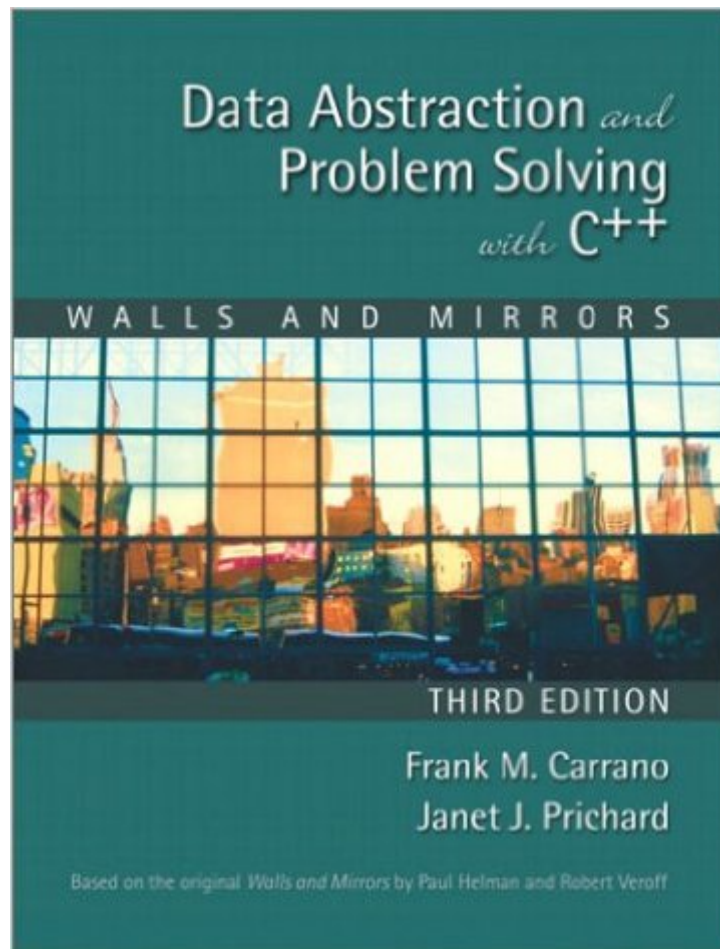


The book was found

# Data Abstraction And Problem Solving With C++: Walls And Mirrors (3rd Edition)



## Synopsis

This classic book has been revised to further enhance its focus on data abstraction and data structures using C++. The book continues to provide a firm foundation in data abstraction, emphasizing the distinction between specification and implementation as the foundation for an object-oriented approach. The authors cover key object-oriented concepts, including encapsulation, inheritance and polymorphism. However, the focus remains on data abstraction instead of simply C++ syntax. The authors also illustrate the role of classes and ADTs in the problem-solving process, and includes major applications of ADTs, such as searching a flight map and event-driven simulation. The book offers early, extensive coverage of recursion and uses this technique in many examples and exercises. It also introduces analysis of algorithms and the Big "O" notation. In addition, this text reviews, in an appendix, basic C++ syntax for those who either have studied the language previously or are making the transition from another language to C++.

## Book Information

Hardcover: 944 pages

Publisher: Addison Wesley; 3 edition (August 9, 2001)

Language: English

ISBN-10: 0201741199

ISBN-13: 978-0201741193

Product Dimensions: 8.2 x 1.5 x 10.2 inches

Shipping Weight: 4.1 pounds

Average Customer Review: 3.5 out of 5 stars [See all reviews](#) (58 customer reviews)

Best Sellers Rank: #1,148,314 in Books (See Top 100 in Books) #112 in [Books > Computers & Technology > Programming > Algorithms > Data Structures](#) #576 in [Books > Computers & Technology > Programming > Languages & Tools > C & C++ > C++](#) #797 in [Books > Computers & Technology > Programming > Microsoft Programming > C & C++ Windows Programming](#)

## Customer Reviews

This book was used for a 200 level computer science course at my school and it multiplied my level of frustration by a factor of 2. There is way too much pseudocode in this book. While I see the importance of pseudocode when actually developing your application, I don't think it's quite practical for the purposes of teaching. About half of the book's chapters will create entire methods and classes in pseudocode only to have that same code rewritten in actual Java syntax one or two pages later. In my opinion it just seems a waste of page space and time since you could easily write

the proper code and just comment what concept or idea should be going on in real Java syntax. Often times, the authors will start a sentence with phrases such as, "Clearly...", "While this implementation is pretty straightforward...", and "Obviously..." Then when I start reading the actual paragraph that's supposed to be obvious, confusion would set in, and other choice words come to my head that would describe the clarity of the writing. I have a feeling the only reason why the computer science department at my school chose this book is because someone knew one of the publishing rep's and that's why it's been used for so long. If you want good books on how to actually program and work with data structures on a theoretical level, then I suggest Data Structures & Algorithms in Java by Robert LaFore. His writing is way more concise and I found it solidified the material in about half the time as when I was trying to read this book. In fact, I would say almost any programming books published by Sams' Publishing are really quite good. I picked up Teach Yourself C in 21 days a few years ago, and it was super easy to understand what it is I was typing. I am not the only one who has the belief. One of my friends in the graduate program pretty much said the same thing and said he didn't even read the book when he took the same class as an undergraduate.

I picked this up because my school requires this data structure book. Well, this book is fine overall. I took a look at a few chapters, and I did not have many huge problems. So let's talk about the pros and cons of this book. Pros: The author explains everything in a very detailed way, which is good for people who don't really have good background in programming and problem solving. Cons: You won't find any complete code in this book since the author only gives the main part of the program, which means you need to develop the program yourself. Even in the source code the author provides, there is not any complete code that you can run in IDE. So, if you don't have a strong Java background, this book might be a little tough for you.

This book does an excellent job of introducing the mechanics of Data structures. A very useful book to refresh one's knowledge about data structures and get a rigorous insight in the subject in preparation for advanced studies in the area of Data Structures. Good book for an introductory University course in Data Structures. This book has been successfully used (and is still being used) as a standard textbook in an intro course in Data Structures at UT Austin. Prerequisites: At least 1 introductory programming course in any high level language (preferably C++). A decent knowledge of C++. (no need of OOP knowledge). Reader should be prepared to seriously study this book. This is a full blown ACADEMIC book, not a tutorial

This book is one of the best of its kind that I have read. It is very descriptive and contains a lot of good examples on the subjects. It describes the construction of a lot of the collection classes like lists, trees, queues etc. and how this is most efficiently sorted and structured. Other subjects are graphs, the Big "O" Notation for evaluation of algorithm performance and a very good description on how and when to use recursion (The mirrors). All subjects are described in detail with great examples. To further test if the subjects have been understood a self-test section is at the end of each chapter (and the answers are in the back of book). The reader of the book should have some knowledge of object-oriented design, but besides that the code is fairly easy to read. In short it's a buy.

"Algorithms and Data Structures" is a huge field. Lot of algorithms and data structures are used in today's computer software of various types. Not all data structures or algorithms on them are adequate for solving a particular problem, so you must have some skills to say which one is "better" than another in your particular situation. Despite its name, this book is an almost complete reference to achieve these skills. "This is great book!!!!" I like it. ( ^.' )

This book has some errors. So long as you or your instructor is aware of that, it's not bad. I like how they present things, and it's clear on most subjects it presents. A few subjects were under-explained, in my opinion, as occasionally the information wasn't enough to complete the required work. Buuuut, there's plenty of unofficial supplementary material on stackoverflow.

If you want to learn about data structures, then get this book; you won't find a better one. But you'll need a good grip on the C++ language to take full advantage of this book. Don't make the mistake of thinking that this book will also teach you C++. The book is written to teach you data structures; and hopefully you'll pick up some good design habits along the way.

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

Data Abstraction and Problem Solving with C++: Walls and Mirrors (3rd Edition) Data Abstraction and Problem Solving with Java: Walls and Mirrors (3rd Edition) Data Abstraction and Problem Solving with C++: Walls and Mirrors (4th Edition) Intermediate Problem Solving and Data Structures: Walls and Mirrors (The Benjamin/Cummings Series in Computer Science) Data Abstraction and Problem Solving with Java (2nd Edition) Clinical Problem Solving in Orthodontics and Paediatric Dentistry, 2e (Clinical Problem Solving in Dentistry) Clinical Problem Solving in Periodontology and

Implantology, 1e (Clinical Problem Solving in Dentistry) Data Analytics: Practical Data Analysis and Statistical Guide to Transform and Evolve Any Business. Leveraging the Power of Data Analytics, Data ... (Hacking Freedom and Data Driven) (Volume 2) Data Analytics: What Every Business Must Know About Big Data And Data Science (Data Analytics for Business, Predictive Analysis, Big Data) Swift: Programming, Master's Handbook; A TRUE Beginner's Guide! Problem Solving, Code, Data Science, Data Structures & Algorithms (Code like a PRO in ... engineering, r programming, iOS development) Ruby: Programming, Master's Handbook: A TRUE Beginner's Guide! Problem Solving, Code, Data Science, Data Structures & Algorithms (Code like a PRO in ... web design, tech, perl, ajax, swift, python,) Java Programming: Master's Handbook: A TRUE Beginner's Guide! Problem Solving, Code, Data Science, Data Structures & Algorithms (Code like a PRO in ... web design, tech, perl, ajax, swift, python) Php: Programming, Master's Handbook: A TRUE Beginner's Guide! Problem Solving, Code, Data Science, Data Structures & Algorithms (Code like a PRO in ... engineering, r programming, iOS development,) Python: Programming, Master's Handbook; A TRUE Beginner's Guide! Problem Solving, Code, Data Science, Data Structures & Algorithms (Code like a PRO ... engineering, r programming, iOS development) Framing Floors, Walls and Ceilings: Floors, Walls, and Ceilings (For Pros By Pros) Exploring Stone Walls: A Field Guide to New England's Stone Walls Analytics: Data Science, Data Analysis and Predictive Analytics for Business (Algorithms, Business Intelligence, Statistical Analysis, Decision Analysis, Business Analytics, Data Mining, Big Data) ADTs, Data Structures, and Problem Solving with C++ Java Artificial Intelligence: Made Easy, w/ Java Programming; Learn to Create your \* Problem Solving \* Algorithms! TODAY! w/ Machine Learning & Data ... engineering, r programming, iOS development) Javascript Artificial Intelligence: Made Easy, w/ Essential Programming; Create your \* Problem Solving \* Algorithms! TODAY! w/ Machine Learning & Data ... engineering, r programming, iOS development)

[Dmca](#)