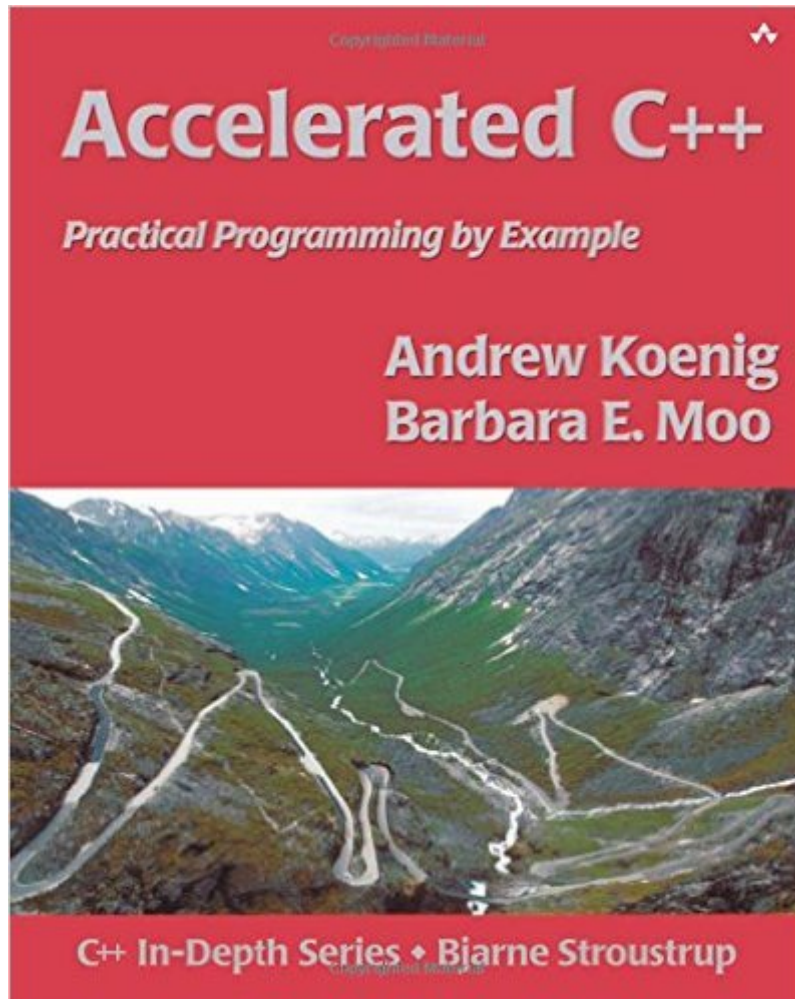# Accelerated C++: Practical Programming By Example

Synopsis

This is a first-rate introductory book that takes a practical approach to solving problems using C++. It covers a much wider scope of C++ programming than other introductory books Ive seen, and in a surprisingly compact format. - Dag Bruck, founding member of the ANSI/ISO C++ committee The authors present a clear, cogent introduction to C++ programming in a way that gets the student writing nontrivial programs immediately. - Stephen Clamage, Sun Microsystems, Inc., and chair of the ANSI C++ committee Anyone reading just this one book and working through the examples and exercises will have the same skills as many professional programmers. - Jeffrey D. Oldham, Stanford University Why is Accelerated C++ so effective? Because it: *Starts with the most useful concepts rather than the most primitive ones: You can begin writing programs immediately. *Describes real problems and solutions, not just language features: You see not only what each feature is, but also how to use it. * Covers the language and standard library together: You can use the library right from the start. The authors proved this approach in their professional-education course at Stanford University, where student

Book Information

Paperback: 352 pages

Publisher: Addison-Wesley Professional; 1 edition (August 24, 2000)

Language: English

ISBN-10: 020170353X

ISBN-13: 978-0201703535

Product Dimensions: 7.3 x 0.9 x 9.1 inches

Shipping Weight: 1.1 pounds (View shipping rates and policies)

Average Customer Review: 4.3 out of 5 starsÂ Â See all reviewsÂ (149 customer reviews)

Best Sellers Rank: #38,005 in Books (See Top 100 in Books)   #13 inÂ Books > Computers & Technology > Programming > Microsoft Programming > C & C++ Windows Programming   #18 inÂ Books > Computers & Technology > Programming > Languages & Tools > C & C++ > C++ #134 inÂ Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

This is a nice change for the better! Instead of introducing the basics and then going on to class design, inheritance, and polymorphism, Andrew and Barbara delay things such as class design until the latter part of the book and instead launch straight into use of the STL, illustrated with practical examples of how to use STL containers and algorithms to solve programming problems.As far as I

am concerned, this is definitely a change for the better. Most programming projects I see heavily abuse inheritance to do things that would have better been done with templates and, as a rule, most C++ projects continue to ignore the STL, even though it is probably the most fundamental contribution to C++ programmer productivity in years. Following this book, readers are encouraged to use STL from the word go, which is as it should be.The language and flow of the book are excellent, and the presentation is very lucid. What really shines through is that both authors are experts on the topic and really know what they are talking about. The explanations of language features are complete, correct, and precise. There is none of the muddled exposition that I find in so many other C++ books. The teaching experience that Andrew and Barbara have gained over the years is truly reflected in the prose. The book anticipates many questions and typically answers them just at the right time, namely, when the question is just about to pop into the reader's head.Information density of the book is high. A lot of information is packed into few words, so the reader has to concentrate on what is being said, and skipping paragraphs is not something that I would recommend.

I'll start by saying that I probably would not have understood this book as well as I have, if I had not already been teaching myself C++ through other forums (primers and online). The thing that most of these other forums have in common is that they start with the basics and build up slowly to the more abstract concepts. The problems come during the switchovers (char* to string, procedural programming to object oriented, pointers to iterators, linked lists to containers, etc). In almost all cases you learn the more basic, and paradoxically more difficult concepts first. Then you have to 'unlearn what you have learned' in order to use the more advanced concepts.So what's different about this book? It teaches a mix of syntax and 'advanced' concepts right from the beginning. You learn the basics of loops and choice statements while using the Standard Library. You also use them in specific examples that have real world uses (the grading program in the first few chapters for example). The Standard Library is _easy_ compared to arrays, char*, rolling your own linked list, using pointers, etc. Since it takes far less time to learn, you can be writing useful programs very quickly. _Then_ the authors go on to describe some of the more 'basic' concepts, usually in terms of how they implement some of the ideas behind the Standard Library. Since you have that understanding already, things like pointers become easier not only to learn, but to understand how they can be used.I have one complaint about the book, and that's with the grading program, specifically how it appears in Chapter 4. As written, it's very confusing to actually enter data to get it to run correctly. A minor complaint though, considering how many times I thought to myself 'Aha!

Since I'm reviewing both "Accelerated C++" as well as "Thinking in C++", I thought I'd post the review on both book links.There have already been excellent reviews of this book, but I would think the main reason people purchase these books is because they have an explicit purpose, which is to learn Standard C++. That being said, C++ is one of the most powerful and multi-faceted languages around, and no other language provides both high level abstractions and low level control in one programming language as C++.Because of these features, it is often opinied that C++ is too complicated, large and takes too long to master. While there are some merits to this criticism, many important real world systems are being bult with C++ and professional developers need to master the fundamentals of C++ now. With that in mind, and after spending (or wasting) much money on various books proclaiming to teach C++ from the ground up, it was not until reading Konig and Moo's "Accelerated C++" and Eckel's "Thinking in C++" and in that order, that I finally "got it".Why I emphasize "in that order" in the last paragraph, is due to the methodologies used to teach C++ by ACPP and TICPP, and due to this, its best to read ACPP first, then TICPP for the most effective learning. Here's my reasons:ACPP teaches C++ in a top down fashion. What I mean by this, is that the higher level Standard Libraries are integrated right from the start to teach programming contructs such as looping, selection and decisions making using library facilities such as , , and . The Standard is utilized from chapter zero, and relieves much of the burden of C strings and all the low level details one would have to know to use them properly.

<u>*Download to continue reading...*</u>

Accelerated C++: Practical Programming by Example Programming #8:C Programming Success in a Day & Android Programming In a Day! (C Programming, C++programming, C++ programming language, Android , Android Programming, Android Games) Programming #57: C++ Programming Professional Made Easy & Android Programming in a Day (C++ Programming, C++ Language, C++for beginners, C++, Programming ... Programming, Android, C, C Programming) Accelerated Spanish: Learn fluent Spanish with a proven accelerated learning system Learning: 25 Learning Techniques for Accelerated Learning - Learn Faster by 300%! (Learning, Memory Techniques, Accelerated Learning, Memory, E Learning, ... Learning Techniques, Exam Preparation) Programming #45: Python Programming Professional Made Easy & Android Programming In a Day! (Python Programming, Python Language, Python for beginners, ... Programming Languages, Android Programming) Programming: Computer Programming for Beginners: Learn the Basics of Java, SQL & C++ - 3. Edition (Coding, C Programming, Java Programming, SQL Programming, JavaScript, Python, PHP) Raspberry Pi 2: Raspberry Pi 2 Programming Made Easy (Raspberry Pi,

Android Programming, Programming, Linux, Unix, C Programming, C+ Programming) Android: Programming in a Day! The Power Guide for Beginners In Android App Programming (Android, Android Programming, App Development, Android App Development, ... App Programming, Rails, Ruby Programming) DOS: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of DOS programming (DOS, ADA, Programming, DOS Programming, ADA ... LINUX, RPG, ADA Programming, Android, JAVA) ASP.NET: Programming success in a day: Beginners guide to fast, easy and efficient learning of ASP.NET programming (ASP.NET, ASP.NET Programming, ASP.NET ... ADA, Web Programming, Programming) C#: Programming Success in a Day: Beginners guide to fast, easy and efficient learning of C# programming (C#, C# Programming, C++ Programming, C++, C, C Programming, C# Language, C# Guide, C# Coding) FORTRAN Programming success in a day:Beginners guide to fast, easy and efficient learning of FORTRAN programming (Fortran, Css, C++, C, C programming, ... Programming, MYSQL, SQL Programming) Prolog Programming; Success in a Day: Beginners Guide to Fast, Easy and Efficient Learning of Prolog Programming (Prolog, Prolog Programming, Prolog Logic, ... Programming, Programming Code, Java) R Programming: Learn R Programming In A DAY! - The Ultimate Crash Course to Learning the Basics of R Programming Language In No Time (R, R Programming, ... Course, R Programming Development Book 1) Parallel Programming: Success in a Day: Beginners' Guide to Fast, Easy, and Efficient Learning of Parallel Programming (Parallel Programming, Programming, ... C++ Programming, Multiprocessor, MPI) MYSQL Programming Professional Made Easy 2nd Edition: Expert MYSQL Programming Language Success in a Day for any Computer User! (MYSQL, Android programming, ... JavaScript, Programming, Computer Software) Programming Raspberry Pi 3: Getting Started With Python (Programming Raspberry Pi 3, Raspberry Pi 3 User Guide, Python Programming, Raspberry Pi 3 with Python Programming) VBScript: Programming Success in a Day: Beginner's Guide to Fast, Easy and Efficient Learning of VBScript Programming (VBScript, ADA, ASP.NET, C#, ADA ... ASP.NET Programming, Programming, C++, C) Apps: Mobile App Trends in 2015 (iOS, Xcode Programming, App Development, iOS App Development, App Programming, Swift, Without Coding) ((Android, Android ... App Programming, Rails, Ruby Programming))