A Practical Guide to Developing Computational Software

Dr. Yong-Ming Li

A Practical Guide to Developing Computational Software

Dr. Yong-Ming Li

A Practical Guide to Developing Computational Software Dr. Yong-Ming Li
This book is written for those who want to pursue a career in developing computational software for engineering and scientific applications. Unlike traditional numerical programming books that focus on the analysis and implementation of numerical methods, this book emphasizes on the development of a reliable and reusable software package. Readers will not only learn implementation of numerical methods but also the software development process that includes creating and using a dynamic-link library, designing flexible test drivers, writing scripting tools for productivity, performing and validating an automated test suite. Based on the computational library developed in this book, readers will also learn how to develop a windows-based application for data visualization and manipulation. Multi-core processors bring parallel computing to mainstream customers. The shift to parallel computing leads to fundamental changes in the design of software. For this reason, this book discusses also how classical numerical programs can be parallelized via Open Multi-Processing.

Numerical methods in this book include evaluation of polynomial and series, root-finding, linear and nonlinear systems, inverse of a matrix, eigenvalues and eigenvector, integration, and least squares approximation. These methods are grouped and presented based on their implementation styles rather than their relevance. This book is organized as follows:

Chapter 1 is a fast-paced brief introduction to C/C++ programming under Microsoft Visual Studio to familiarize readers with basic C/C++ syntax and debugging tools.

Chapter 2 discusses floating-point notation, comparison, and arithmetic. Rudimentary understanding of floating-point is a pre-requisite for programmers. Failure to understand it is often the source of problems in numerical programming.

Chapter 3 continues the study of advanced C/C++ programming such as default arguments, data structure and class, double pointers, dynamic memory allocations, and STL containers. Algorithm efficiency analysis and big O notation will also be discussed. This chapter is designed to help readers to gain the required C/C++ proficiency in implementing numerical methods.

Chapter 4 is devoted to give readers an insight on how a computational software library may actually be developed in a software house. Readers will learn how to create and use a dynamic-link library, how to design flexible test drivers, and how to write scripts to improve productivity, to execute test suites automatically, and to compare the test results with the predicted outcomes.

Chapter 5 deals with recursive algorithm. Because of its problem-solving power and simplicity in implementation, recursion in numerical methods will be discussed in this chapter with emphasizes on performance and memory usage.

Chapter 6 discusses linear systems. Topics include solution to system of linear equations, matrix manipulation, inverse of a matrix, eigenvalue and eigenvector.

Chapter 7 and 8 explore how to use function pointers, generic data pointer, and inheritance with polymorphism to design extensible and reusable code - an important topic in software engineering.

Chapter 9 discusses the least square approximation method whose applications can be found in many fields such as computer-aided design, metrology, image processing, etc.

Chapter 10 aims to develop a simple windows-based application for data visualization and manipulation. Through this miniature application, readers will get a glimpse of how sophisticated CAD/CAM systems are developed.

Chapter 11 discusses how classical numerical methods can be parallelized to take the advantage of multi-thread programming. Common problems associated with parallel computing such as data race conditions, workload balance, synchronization, and parallel slowdown are discussed in detail.

Appendix A is a brief introduction to Perl programming.

Appendix B contains answers to all seven pre-interview questions given in the preface.

From reader reviews:

Eva Oleary:

The book A Practical Guide to Developing Computational Software make one feel enjoy for your spare time. You can utilize to make your capable considerably more increase. Book can to become your best friend when you getting tension or having big problem along with your subject. If you can make studying a book A Practical Guide to Developing Computational Software to become your habit, you can get far more advantages, like add your own capable, increase your knowledge about some or all subjects. You could know everything if you like open up and read a guide A Practical Guide to Developing Computational Software. Kinds of book are several. It means that, science guide or encyclopedia or other individuals. So , how do you think about this e-book?

Stan Smith:

A lot of people always spent their particular free time to vacation or go to the outside with them friends and family or their friend. Did you know? Many a lot of people spent they free time just watching TV, as well as playing video games all day long. In order to try to find a new activity that's look different you can read a book. It is really fun for yourself. If you enjoy the book you read you can spent the whole day to reading a book. The book A Practical Guide to Developing Computational Software it doesn't matter what good to read. There are a lot of people that recommended this book. They were enjoying reading this book. When you did not have enough space to create this book you can buy the actual e-book. You can m0ore simply to read this book from a smart phone. The price is not very costly but this book features high quality.

Deanna Reed:

Playing with family inside a park, coming to see the sea world or hanging out with good friends is thing that usually you might have done when you have spare time, in that case why you don't try factor that really opposite from that. A single activity that make you not sensation tired but still relaxing, trilling like on roller coaster you already been ride on and with addition of knowledge. Even you love A Practical Guide to Developing Computational Software, you can enjoy both. It is good combination right, you still wish to miss it? What kind of hang-out type is it? Oh seriously its mind hangout folks. What? Still don't get it, oh come on its referred to as reading friends.

Georgia Evans:

The book untitled A Practical Guide to Developing Computational Software contain a lot of information on this. The writer explains your ex idea with easy technique. The language is very simple to implement all the people, so do certainly not worry, you can easy to read it. The book was written by famous author. The author brings you in the new era of literary works. You can actually read this book because you can keep reading your smart phone, or gadget, so you can read the book within anywhere and anytime. If you want to buy the e-book, you can open their official web-site as well as order it. Have a nice go through.

Read A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li for online ebook

A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li Free PDF d0wnl0ad, audio books, books to read, good books to read, cheap books, good books, online books, books online, book reviews epub, read books online, books to read online, online library, greatbooks to read, PDF best books to read, top books to read A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li books to read online.

Online A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li ebook PDF download

A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li Doc

A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li Mobipocket

A Practical Guide to Developing Computational Software by Dr. Yong-Ming Li EPub