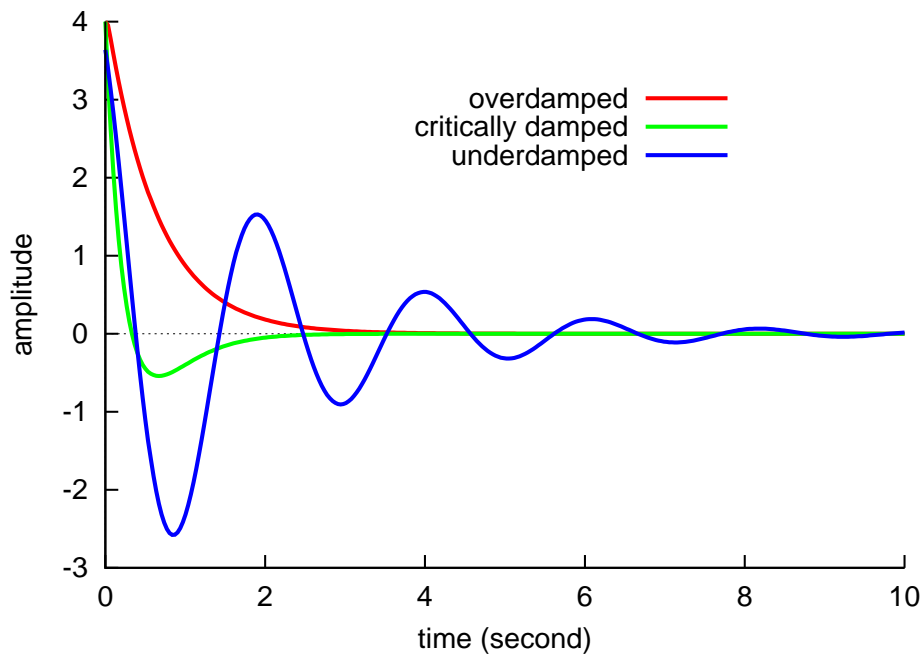# Instructor's Guide

C for Engineers and Scientists

An Interpretive Approach

Harry H. Cheng

University of California, Davis

Revision 1.0, March 2009

Table of Contents

Preface to the Instructor

C for Engineers and Scientists: An Interpretive Approach is a comprehensive book on software development for solving problems in engineering and science with complete coverage of the C language and other programming languages. The materials are more than enough for a one-semester course. The book can be used as an introductory textbook for students majoring in engineering, computer science, and science.

Unlike other introductory textbooks for teaching one course only, this book can also serve as a reference book for students who intend to take other advanced courses or during engineering practice after their graduation. Sections and chapters marked with the double dagger symbol '‡' in the title can be skipped without hindering learning about the later chapters. The materials marked with the symbol '‡', however, are very useful for those who plan on serious software development in C, need to read existing C code written by experienced programmers, or intend to take other advanced courses.

Part I is organized cumulatively except for Chapter 16 "Scientific Computing in the Entire Real Domain in C99," and Chapter 17 "Programming with Complex Numbers in C99 and C++." These two chapters can be read after Chapter 11 "Pointers" is finished. Other parts are self-contained. After covering sections not marked with the double dagger symbol '‡' in Part I, one can move to Part II. Likewise, after finishing Chapter 10 "Arrays" in Part I, one can continue to Part III or Part IV.

At the first glance, some people might think that the book is more machine-oriented. Actually, the topics in the book are carefully organized for the best information flow for beginners. For example, the representations of different integer numbers are first introduced in section 3.1 in Chapter 3. The remaining sections in Chapter 3 then describe how variables and constants for these different integral types are represented in C. As another example, a statistics library is developed in Chapter 10 on Arrays. This statistics library needs the information about how header files are used, which is described in Chapter 7 on Preprocessing Directives. A homework in Chapter 10 also asks students to create a matrix-vector library with a header file. These application-oriented libraries with header files greatly enhance students' understanding how arrays are used to solve real-world problems. Therefore, the preprocessing directives in Chapter 7 are presented before Chapter 10 on Arrays. However, most topics in Chapter 7 on preprocessing directives should be skipped for an introductory class. As a result, most sections in Chapter 7 are marked with double daggers '‡'. In my teaching, I only spend 35 minutes on Chapter 7.

The sample topics and their lecture time, homeworks, quizzes, and midterm and final examinations in this *Instructor's Guide* have been used for an introductory computer programming course **Computer Programming for Engineering Applications** for freshman students at the University of California, Davis over the years. Either ten-minutes weekly quizzes or a midterm examination are given. In this course, it is assumed that students have no any prior computer programming experience. All sections and chapters not marked with the double dagger symbol '‡' are covered in one quarter with ten weeks. Each week has three one-hour lectures, one hour discussion, and several lab hours. It takes eight weeks to cover Chapters 1-8 and 10-14 in Part I, one week on computational arrays in Chapter 21 in Part III and plotting in Chapter 20 in Part II, and one week on Chapter 23 in Part IV for comparison study with MATLAB.

Depending on your teaching objectives, you may teach at a different pace and select different topics. If your course is offered in a semester, additional topics marked with the double dagger symbol '‡' may be covered. The sample topics and their lecture time, homeworks, quizzes, and midterm and final examinations in this *Instructor's Guide* are just for your reference. For your convenience. PDF files and source code for homeworks, quizzes, and midterm and final examinations are available in separate directories. The solutions with source code for exercises in each chapter of the book can be found in *Solutions Manual*.

Harry H. Cheng

Topics

1. Introduction to Computer Hardware and Software (Chapter 1)

2. Getting Started (Chapter 2)

3. Scalar Types, Number Systems, and Input/Output (Chapter 3)

4. Operators and Expressions (Chapter 4)

5. Statements and Control Flow (Chapter 5)

6. Functions and Introduction to Two-Dimensional Plotting Using **fplotxy()** and **fplotxyz()** (Chapter 6)

7. Preprocessing Directives (Chapter 7)

8. Storage Classes (Chapter 8)

9. Arrays and Introduction to Two-Dimensional Plotting Using **plotxy()** (Chapter 10)

10. Pointers (Chapter 11)

11. Characters and Strings (Chapter 12)

12. Structures and Enumerations (Chapter 13)

13. File Processing (Chapter 14)

14. Computational Arrays and Matrix Computations in Ch (Chapter 21)

15. Two- and Three-Dimensional Plotting (Chapter 20)

16. Introduction to MATLAB and Comparison Study with C/Ch (Chapter 23)

   - Comparison study of type-less versus typed languages. Style of MATLAB programming.
   - Variables and arrays in MATLAB
   - Matrix computation and linear algebra in MATLAB.
   - Operators in MATLAB.
   - Iterative and selection statements in MATLAB .
   - Formatted input and output in MATLAB .
   - File processing in MATLAB.
   - Two- and three-dimensional plotting in MATLAB.
   - Functions and function files (M-files) in MATLAB.
   - Functions using arrays in MATLAB.

17. Introduction to numerical analysis functions, some commonly used libraries and software packages in C/Ch/C++/MATLAB for applications in engineering and science as well as other advanced courses.

Topics and Their Lecture Time

1. Introduction to Computer Hardware and Software (Chapter 1, 30 minutes)

2. Getting Started (Chapter 2, 50 minutes)

3. Scalar Types, Number Systems, and Input/Output (Chapter 3, 100 minutes)

4. Operators and Expressions (Chapter 4, 60 minutes)

5. Statements and Control Flow (Chapter 5, 130 minutes)

6. Functions and Introduction to Two-Dimensional Plotting Using **fplotxy()** and **fplotxyz()** (Chapter 6, 110 minutes)

7. Preprocessing Directives (Chapter 7, 35 minutes)

8. Storage Classes (Chapter 8, 50 minutes)

9. Arrays and Introduction to Two-Dimensional Plotting Using **plotxy()** (Chapter 10, 150 minutes)

10. Pointers (Chapter 11, 140 minutes)

11. Characters and Strings (Chapter 12, 70 minutes)

12. Structures and Enumerations (Chapter 13, 170 minutes)

13. File Processing (Chapter 14, 70 minutes)

14. Computational Arrays and Matrix Computations in Ch (Chapter 21, 60 minutes)

15. Two- and Three-Dimensional Plotting (Chapter 20, 30 minutes)

16. Introduction to MATLAB and Comparison Study with C/Ch (Chapter 23, 120 minutes)

   - Comparison study of type-less versus typed languages. Style of MATLAB programming.
   - Variables and arrays in MATLAB
   - Matrix computation and linear algebra in MATLAB.
   - Operators in MATLAB.
   - Iterative and selection statements in MATLAB .
   - Formatted input and output in MATLAB .
   - File processing in MATLAB.
   - Two- and three-dimensional plotting in MATLAB.
   - Functions and function files (M-files) in MATLAB.
   - Functions using arrays in MATLAB.

17. Introduction to numerical analysis functions, some commonly used libraries and software packages in C/Ch/C++/MATLAB for applications in engineering and science as well as other advanced courses (20 minutes).