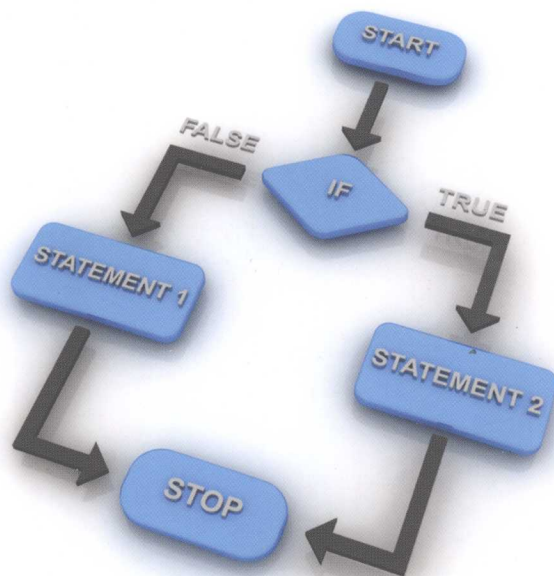


Chapman & Hall/CRC
Computational Science Series

Introduction to Elementary Computational Modeling

Essential Concepts, Principles,
and Problem Solving



José M. Garrido



CRC Press
Taylor & Francis Group

A CHAPMAN & HALL BOOK



30809206

Introduction to Elementary Computational Modeling

Essential Concepts, Principles,
and Problem Solving

José M. Garrido

Kennesaw State University

Georgia, USA



CRC Press

Taylor & Francis Group

Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

A CHAPMAN & HALL BOOK

MATLAB® is a trademark of The MathWorks, Inc. and is used with permission. The MathWorks does not warrant the accuracy of the text or exercises in this book. This book's use or discussion of MATLAB® software or related products does not constitute endorsement or sponsorship by The MathWorks of a particular pedagogical approach or particular use of the MATLAB® software.

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2012 by Taylor & Francis Group, LLC
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works

Printed in the United States of America on acid-free paper
Version Date: 2011915

International Standard Book Number: 978-1-4398-6739-6 (Paperback)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (<http://www.copyright.com/>) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

Visit the Taylor & Francis Web site at
<http://www.taylorandfrancis.com>

and the CRC Press Web site at
<http://www.crcpress.com>

Preface

Computational science is an emerging area (or discipline) that includes concepts, principles, and methods from applied mathematics and algorithmic design and computer programming; these are applied in various areas of science and engineering to solve large-scale scientific problems. A *computational model* is a computer implementation of the solution to a (scientific) problem for which a mathematical representation has been formulated. Developing a computational model includes formulating the mathematical representation and implementing it by applying computer science concepts, principles and methods. Computational modeling is the foundational component of computational science and focuses on reasoning about problems using *computational thinking* and developing models for problem solving.

The primary goal of this book is to introduce readers to the basic principles of computational modeling at the level of fundamental concepts. Emphasis is on reasoning about problems, conceptualizing the problem, elementary mathematical modeling, and their computational solution that involves computing results and visualization.

The book emphasizes analytical skill development and problem solving rather than programming language syntax and explains simple pseudo-code, simple algorithm design, and MATLAB[®], Octave, and FreeMat programming. MATLAB is a widely used scientific software tool; Octave and FreeMat are freely available software tools compatible with MATLAB.

MATLAB[®] is a registered trademark of The Mathworks, Inc. For product information, please contact: The MathWorks, Inc.

3 Apple Hill Drive
Natick, MA 01760-2098 USA
Tel: 508 647 7000
Fax: 508-647-7001
E-mail: info@mathworks.com
Web: www.mathworks.com

Although computational models require high-performance computing to solve large and complex scientific problems, this book presents only the elementary notions of computational models. For beginning undergraduate students (of science and engineering), the goal is to provide relevant material for

easy understanding of computational models and their development, as early as possible in their studies. Therefore, the book takes an “early introduction” approach to computational models by providing the readers with a strong foundation in computational modeling by applying elementary mathematical concepts and basic principles, methods, and techniques of computer science.

The book consists of two parts. Part I presents the basic concepts and principles of computational models. In this part, the book discusses basic modeling and techniques for designing and implementing problem solutions, independent of software and hardware tools. Standard pseudo-code constructs and flowcharts are explained and applied in designing models for various case studies. These are implemented using only the minimum necessary knowledge of MATLAB, a widely used scientific software tool, and Octave, a freely available tool that is compatible with MATLAB. Part II of the book presents the elementary mathematical modeling principles and more detailed computer implementation of models with programming constructs using MATLAB and Octave. Examples and case studies demonstrate the computation and visualization of computational models.

The basic syntax constructs of MATLAB and Octave are presented gradually and the various programming principles are explained in an incremental manner for the actual implementation of computational models. This book avoids presenting too much syntax of the programming language at the beginning, which usually results in unnecessary difficulty for the student in understanding the underlying concepts in problem solution and programming.

There are very few books on the basic treatment of computational models; this book and its associated models were designed as teaching materials. The book is aimed at beginning college students in computer science, mathematics, statistics, science, and engineering. The main features of the book are the following:

- Explanation of the basic concepts and principles of computational models are explained.
- Discussion of the models is based on elementary mathematical models, at the level of pre-calculus.
- Emphasis on modularity and abstraction that help in dealing with large and complex models. These concepts are introduced early in Chapters 1 and 2.
- Discussion of the design of algorithmic solutions to problems using standard flowcharts and pseudo-code.
- Implementation of the problem solutions and the corresponding models in MATLAB and Octave.

The material in this book is adequate at the beginning undergraduate level, a CS0 course or a CS1 course. As mentioned previously, the material can also be used in other undergraduate curricula, e.g., mathematics, engineering, and the other sciences. Because of the level of mathematics used, the book can also be utilized for a course in college preparation at the high school level.

This book will be extremely helpful to readers who have never programmed before and the book can help improve the overall approach in teaching/learning programming principles by emphasizing more problem solving, abstraction, algorithm design with pseudo-code, and basic programming before teaching/learning more advanced programming principles with standard programming languages such as Java, C++, Ada, and others.

I acknowledge the help provided through discussions with some of my colleagues, namely Dr. Ben Setzer and Dr. Dick Gayler. I also acknowledge and give special thanks to Professor Rich Schlesinger for writing Chapter 1.

José M. Garrido
Kennesaw, Georgia

About the Author

José M. Garrido is professor of computer science in the Department of Computer Science, Kennesaw State University, Georgia. He holds a Ph.D. from George Mason University in Fairfax, Virginia, an M.S.C.S also from George Mason University, an M.Sc. from the University of London, and B.S. in electrical engineering from the Universidad de Oriente, Venezuela.

Dr. Garrido's research interest is on object-oriented modeling and simulation, multi-disciplinary computational modeling, formal specification of real-time systems, language design and processors, modeling systems performance, and software security. Dr. Garrido developed the Psim3, PsimJ, and PsimJ2 simulation packages for C++ and Java. He has recently developed the OOSimL, the Object-Oriented Simulation Language (with partial support from NSF).

Dr. Garrido has published several papers on modeling and simulation, and on programming methods. He has also published six textbooks on object-oriented simulation and operating systems.

Contents

List of Figures	xv
List of Tables	xix
Preface	xxi
About the Author	xxv
I Understanding Problem Solving	1
1 Understanding Word Problems	3
1.1 Introduction	3
1.2 Nouns and Verbs	4
1.3 Problem Decomposition	6
1.4 The Language of Computational Problems	8
1.4.1 Terms for Computed Values	8
1.4.2 Implied Phrases	9
1.4.3 Units of Measure	9
1.4.4 Conditions	11
1.4.5 Repetition	13
1.5 Objects	14
1.6 Problems with Many Numbers	15
1.6.1 Working with Patterns	16
1.7 Summary	17
Key Terms	17
Exercises	18
2 Problem Solving and Computing	21
2.1 Introduction	21
2.2 Programs	21
2.3 Data Definitions	22
2.3.1 Name of Data Items	22
2.3.2 Data Types	23
2.3.3 Data Declarations	24
2.4 Programming Languages	24

2.4.1	High-Level Programming Languages	24
2.4.2	Interpreters	25
2.4.3	Compilers	25
2.4.4	Compiling and Execution of Java Programs	25
2.4.5	Compiling and Executing C++ Programs	26
2.5	Interpretation of Commands with MATLAB [®] and Octave . .	27
2.6	Computer Problem Solving	28
2.7	Summary	30
	Key Terms	31
	Exercises	31
3	MATLAB and Octave Programming	33
3.1	Introduction	33
3.2	The MATLAB and Octave Prompt	33
3.3	Variables and Constants	35
3.4	Assignment Statements	35
3.5	Simple Mathematical Expressions	36
3.6	Scientific Notation	37
3.7	Built-In Mathematical Functions	37
3.8	Internal Documentation	38
3.9	Summary	39
	Key Terms	39
	Exercises	39
II	Computational Models	41
4	Introduction to Computational Models	43
4.1	Introduction	43
4.2	Preliminary Concepts	43
4.3	A Simple Problem: Temperature Conversion	46
4.3.1	Initial Problem Statement	46
4.3.2	Analysis and Conceptual Model	46
4.3.3	The Mathematical Model	47
4.4	Using MATLAB and Octave	48
4.4.1	Basic MATLAB and Octave Commands	48
4.4.2	The Computational Model	48
4.4.3	Using Data Lists with MATLAB and Octave	50
4.4.4	Implementation of Model with Data Lists	53
4.5	Summary	55
	Key Terms	56
	Exercises	56

5	Computational Models and Simulation	57
5.1	Introduction	57
5.2	Categories of Computational Models	57
5.3	Development of Computational Models	59
5.4	Simulation: Basic Concepts	61
5.4.1	Simulation Models	62
5.4.2	Simulation Results	64
5.5	Modular Decomposition	64
5.6	Average and Instantaneous Rate of Change	65
5.7	Area under a Curve	67
5.8	The Free-Falling Object	68
5.8.1	Initial Problem Statement	69
5.8.2	Analysis and Conceptual Model	69
5.8.2.1	Assumptions	70
5.8.2.2	Basic Definitions	70
5.8.3	The Mathematical Model	70
5.8.4	The Computational Model	71
5.8.4.1	Simple Implementation	71
5.8.4.2	Implementation with Arrays	75
5.8.4.3	Computing the Rates of Change	78
5.9	Summary	79
	Key Terms	80
	Exercises	80
6	Algorithms and Design Structures	83
6.1	Introduction	83
6.2	Problem Solving	83
6.3	Algorithms	84
6.4	Describing Data	84
6.5	Notations for Describing Algorithms	85
6.5.1	Flowcharts	85
6.5.2	Pseudo-Code	86
6.6	Algorithmic Structures	87
6.6.1	Sequence	88
6.6.2	Selection	88
6.6.3	Repetition	88
6.7	Implementation of Algorithms	90
6.7.1	Programming Languages	91
6.7.2	Assignment and Arithmetic Expressions	92
6.7.3	Simple Numeric Computations	92
6.7.4	Simple Input/Output	93
6.8	Computing Area and Circumference	95
6.8.1	Specification	95

6.8.2	Algorithm with the Mathematical Model	95
6.9	Summary	97
	Key Terms	97
	Exercises	97
7	Selection	99
7.1	Introduction	99
7.2	Selection Structure	99
7.2.1	General Concepts of the Selection Structure	99
7.2.2	Selection with Pseudo-Code	100
7.2.3	Selection with MATLAB and Octave	101
7.2.4	Conditional Expressions	101
7.2.5	Example with Selection	102
7.3	Complex Numbers with MATLAB and Octave	104
7.4	A Computational Model with Selection	106
7.4.1	Analysis and Mathematical Model	106
7.4.2	Algorithm for General Solution	106
7.4.3	Detailed Algorithm	107
7.5	Multilevel Selection	109
7.5.1	General Multipath Selection	110
7.5.2	The Case Structure	111
7.6	Complex Conditions	113
7.7	Summary	114
	Key Terms	114
	Exercises	115
8	Repetition	117
8.1	Introduction	117
8.2	Repetition with While Construct	117
8.2.1	While-Loop Flowchart	118
8.2.2	The While Structure in Pseudo-Code	118
8.2.3	While-Loop with MATLAB and Octave	119
8.2.4	Loop Counter	119
8.2.5	Accumulator Variables	120
8.2.6	Summation of Input Numbers	121
8.3	Repeat-Until Construct	122
8.4	For Loop Structure	124
8.4.1	The Summation Problem with a For Loop	125
8.4.2	The Factorial Problem	126
	8.4.2.1 Mathematical Specification of Factorial	126
	8.4.2.2 Computing Factorial	126
8.5	Summary	127
	Key Terms	128

Exercises	128
9 Data Lists	131
9.1 Introduction	131
9.2 Creating an Array	132
9.2.1 Creating Arrays in Pseudo-Code	132
9.2.2 Creating Arrays in MATLAB and Octave	133
9.3 Operations on Arrays	135
9.3.1 Array Elements in Pseudo-Code	136
9.3.2 Using Array Elements with MATLAB and Octave	136
9.3.3 Arithmetic Operations on Vectors	137
9.4 Multidimensional Arrays	138
9.4.1 Multidimensional Arrays with Pseudo-Code	138
9.4.2 Multidimensional Arrays with MATLAB and Octave	139
9.5 Applications Using Arrays	141
9.5.1 Problems with Simple Array Manipulation	141
9.5.1.1 The Average Value in an Array	141
9.5.1.2 Maximum Value in an Array	143
9.5.2 Searching	145
9.5.2.1 Linear Search	145
9.5.2.2 Binary Search	147
9.6 Average and Instantaneous Rate of Change	150
9.6.1 Average Rate of Change	150
9.6.2 Instantaneous Rate of Change	151
9.6.3 Computing the Rates of Change	152
9.7 Area under a Curve	156
9.8 Summary	159
Key Terms	160
Exercises	160
10 Modules	163
10.1 Introduction	163
10.2 Modular Design	163
10.3 MATLAB and Octave Script Files	165
10.4 Functions	166
10.4.1 Function Definition	166
10.4.2 Function Definition in MATLAB and Octave	167
10.4.3 Simple Function Calls	168
10.4.4 Functions with Parameters	169
10.4.5 Function Calls with Data	170
10.4.6 Functions with Return Data	171
10.5 Documenting MATLAB and Octave Functions	172
10.6 Summary	173

Key Terms	173
Exercises	174
11 Mathematical Models: Basic Concepts	175
11.1 Introduction	175
11.2 From the Real-World to the Abstract World	175
11.3 Discrete and Continuous Models	177
11.4 Difference Equations and Data Lists	177
11.5 Functional Equations	180
11.6 Validating a Model	181
11.7 Models with Arithmetic Growth	181
11.8 Using MATLAB and Octave to Implement the Model	182
11.8.1 MATLAB and Octave Implementation	182
11.8.2 Producing the Charts of the Model	183
11.9 Summary	185
Key Terms	185
Exercises	186
12 Models with Quadratic Growth	187
12.1 Introduction	187
12.2 Quadratic Growth	187
12.3 Differences of the Data	188
12.4 Difference Equations	191
12.5 Functional Equations	192
12.6 Models with Quadratic Growth	193
12.6.1 Simple Quadratic Growth Models	193
12.6.2 Models with Sums of Arithmetic Growth	194
12.7 Solution and Graphs of Quadratic Equations	196
12.8 Summary	198
Key Terms	198
Exercises	198
13 Models with Polynomial Functions	201
13.1 Introduction	201
13.2 General Forms of Polynomial Functions	201
13.3 Evaluation and Graphs of Polynomial Functions	202
13.3.1 Evaluating Polynomial Functions	202
13.3.2 Generating Graphs of Polynomial Functions	205
13.4 Solution to Polynomial Equations	206
13.5 Summary	209
Key Terms	209
Exercises	209

14 Data Estimation and Empirical Modeling	211
14.1 Introduction	211
14.2 Interpolation	211
14.2.1 Linear Interpolation	212
14.2.2 Nonlinear Interpolation	215
14.3 Curve Fitting	218
14.4 Summary	220
Key Terms	221
Exercises	221
15 Models with Geometric Growth	225
15.1 Introduction	225
15.2 Basic Concepts of Geometric Growth	225
15.2.1 Geometric Growth with Increasing Data	226
15.2.2 Geometric Growth with Decreasing Data	226
15.2.3 Geometric Growth: Case Study 1	227
15.2.4 Geometric Growth: Case Study 2	230
15.3 Functional Equations in Geometric Growth	232
15.4 Properties of Exponential Functions	233
15.4.1 Exponentiation	233
15.4.2 Logarithms	233
15.5 Summary	236
Key Terms	236
Exercises	236
16 Vectors and Matrices	239
16.1 Introduction	239
16.2 Vectors	240
16.3 Simple Vector Operations	240
16.3.1 Arithmetic Operations	241
16.3.2 Applying Vector Functions	243
16.4 Matrices	245
16.4.1 Arithmetic Operations	246
16.4.2 Function Application	247
16.5 Array Indexing	247
16.6 Plotting Vectors	249
16.7 Summary	251
Key Terms	251
Exercises	251

17 Text Data	253
17.1 Introduction	253
17.2 String Vectors	253
17.2.1 String Operations	253
17.2.2 String Functions	254
17.3 String Matrices	257
17.4 Summary	258
Key Terms	258
Exercises	258
18 Advanced Data Structures	259
18.1 Introduction	259
18.2 Cell Arrays	259
18.3 Structures	264
18.4 Summary	268
Key Terms	268
Exercises	268
Appendix A MATLAB and GNU Octave Software	271
A.1 Introduction	271
A.2 The MATLAB Components	271
A.3 The Desktop	272
A.4 Starting MATLAB	274
A.5 Exiting MATLAB	274
A.6 The Command Window	275
A.7 Current User Folder	275
A.8 The Startup Folder	276
A.9 Using Command Files (Scripts)	276
A.10 MATLAB Functions	280
A.11 GNU Octave	285
Appendix B Computer Systems	289
B.1 Introduction	289
B.2 Computer Systems	289
B.2.1 Hardware Components	290
B.2.1.1 Processors	290
B.2.1.2 Main Memory	291
B.2.1.3 Storage Devices	291
B.2.1.4 Input Devices	292
B.2.1.5 Output Devices	292
B.2.1.6 Bus	292
B.2.2 Computer Networks	292
B.2.3 Software Components	293

B.3	Operating Systems	294
B.3.1	Operating System User Interfaces	294
B.3.2	Contemporary Operating Systems	295
B.3.2.1	Unix	295
B.3.2.2	Microsoft Windows	296
B.4	Summary	296
Key Terms		296
Bibliography		297
Index		299

List of Figures

1.1	Identifying nouns and verbs.	4
1.2	Area of rectangle.	4
1.3	Formula from nouns and verbs.	5
1.4	Heating water.	5
1.5	Multiple steps problem.	7
1.6	Diagram of counting example.	7
1.7	Diagram with average.	8
1.8	Validating conditions.	11
1.9	Simple data validation.	12
1.10	Multiple conditions.	12
1.11	OR used with one value.	13
1.12	OR used with multiple values.	13
2.1	General structure of a program.	22
2.2	Compiling a Java source program.	26
2.3	Executing a Java program.	26
2.4	Compiling a C++ program.	27
2.5	Linking a C++ program.	27
2.6	MATLAB/Octave interpreter.	28
2.7	The waterfall model.	29
3.1	MATLAB/Octave interpreter.	33
3.2	Simple commands in an Octave window.	34
4.1	Computational science as an integration of several disciplines.	45
4.2	Simple commands in an Octave window.	49
4.3	Plot of the temperature conversion.	54
5.1	Discrete changes of number of cars in the queue.	58
5.2	Development of computational models.	59
5.3	Model development and abstract levels.	61
5.4	High-level view of a simulation model.	63
5.5	The slope of a line.	65
5.6	The slope of a secant.	66