

13 February 2023

**COMPUTATION
AND
PROBLEM SOLVING
IN
UNDERGRADUATE PHYSICS
Second Edition**

**PREFACE,
ACKNOWLEDGEMENTS,
DISCLAIMER**

DAVID M. COOK

Department of Physics
Lawrence University
711 E Boldt Way, SPC 24
Appleton, Wisconsin 54911

Copyright © First Edition 2000–16 by David M. Cook
Copyright © Second Edition 2016–23 by David M. Cook



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (creativecommons.org/licenses/by-nc-sa/4.0/). Any use not permitted by this license requires authorization in writing from David M. Cook.

The first edition of this publication was registered with the Library of Congress with the call number QC20.C66.2004.

The second edition was registered with the Library of Congress with the LCCN 2019947884. It has been assigned the ISBN 9780961342968 and carries the Library of Congress call number QC20.C66.2019.

Preface to the Second Edition

Note: Usually with a second edition of a book, the preface to the first edition¹ is preserved and a short preface to the second edition is added. In the present case, the preface to the first edition required here and there a number of edits which, had they not been made, would have been perhaps a bit confusing to readers of the second edition. Consequently, I have elected to depart from the normal practice and simply create a preface for the second edition, though much of its content essentially copies that of the preface to the first edition.

Note: Regardless of which components are included and which omitted in this version of *Computation and Problem Solving in Undergraduate Physics*, the preface, acknowledgements, and disclaimer in the front matter are those from the assemblage containing *all* components.

Since the mid 1980's (including the years since my official retirement in 2008), we in the Department of Physics at Lawrence University have been developing and offering curricular components that

- support efforts to acquaint students with computational procedures and resources early enough so that they will be motivated and prepared to use these resources *on their own initiative* when circumstances warrant and so that later work need not be interrupted to deal with computational issues as an aside to its main purposes, and
- provide students with both the background and the confidence to support informed reading of vendor manuals, which usually do a splendid job of listing capabilities exhaustively but typically burden the beginner with initially irrelevant refinements and fail to illustrate adequately how even the rudimentary capabilities can be combined to perform useful tasks.

Over the years since the mid 1980s, a wide assortment of instructional materials has been drafted and redrafted. This book brings these materials together into a single publication with the hope that it may prove useful to others who seek to achieve these same or similar objectives.²

With these objectives in mind, this book consciously focuses on helping students *get started*. It is not designed to be comprehensive or exhaustive, either in laying out the capabilities of any particular computational resource or in discussing numerical algorithms. Students must understand throughout that they must refer regularly to vendors' manuals and on-line help files for details beyond those discussed in the book—details that may, in fact, be necessary for successful completion of some of the exercises. The need for that activity is noted *here*; repeated reminders will *not* be included in the body of the book.

The book is also not a book about computational physics; it addresses uses of computational tools. Indeed, the sophomore course at Lawrence in which students first encounter this book would not in any way replace a course in computational physics. Rather, the materials treated here should

¹The first edition was published in 2003, though it experienced a number of edits and adjustments in subsequent years.

²The article by David M. Cook titled "Computation in Undergraduate Physics: The Lawrence Approach" and appearing in the American Journal of Physics (Am. J. Phys. **76**, 321-326 (April-May 2008)) describes these efforts in some detail.

provide strong background for a subsequent, junior-senior level course in computational physics, which would—I believe—be substantially enhanced if students came to it already familiar with the resources on which this book concentrates.

One major difficulty in creating materials on computational topics is that different potential users favor different hardware platforms and software packages. Especially in the computational arena, the variety of options and combinations is so great that any single choice (or coordinated set of choices) is bound to limit the usefulness of the product to a small subset of all potential users. This book addresses that difficulty by being assembled from a wide assortment of components, some of which—the generic components—will be included in all versions and others of which—those specific to particular software packages—will be included only if the potential user requests them. Thus, the specific software and hardware discussed in the book can be tailored to the spectrum of resources available at the instructor’s site. Two versions may well differ in numerous respects. One may include the generic components and the components that discuss³ IDL, MAPLE, C (with Numerical Recipes), and L^AT_EX while another may include the generic components and the components that focus on MATLAB, *Mathematica*, and FORTRAN (including Numerical Recipes). The table of contents and index contain only entries from the included chapters and sections. To facilitate communication among users of different versions, however, chapter and section numbers and the numbers identifying package-independent exercises are preserved in all versions. In a version that does not include FORTRAN, for example, the FORTRAN sections will be omitted from Chapters 9, 10, 11, 13, 14, and 15 and Chapters 12 and 16—for which FORTRAN is prerequisite—will be omitted altogether. In addition, FORTRAN-specific exercises will be omitted from the end-of-chapter exercises. Because of version-specific omissions such as those just described, there will therefore be gaps in the chapter, section, and exercise numbers in any version that does not include all options. In contrast, within each chapter, equation numbers, figure numbers, table numbers, and footnote numbers advance from one without gaps, and page numbers run continuously from the beginning of the book to the end. In consequence, the numbers assigned to identical equations, figures, tables, footnotes, and pages may differ from version to version, but the numbers assigned to chapters and sections with identical titles and the numbers assigned to identical exercises will be the same in all versions (and will have gaps reflecting omitted chapters, sections, and exercises). Such flexibility would be impossible were we not able to exploit features of L^AT_EX, including the particular capabilities of the `ifthen` and `imakeidx` packages, to assemble the PDF files that permit the selected assemblage subsequently to be viewed on a screen or printed.

Even among sites that use the same spectrum of hardware and software, however, some aspects of local environments remain unique to individual sites. Local rules of citizenship; the features and elementary resources of the local operating system; local practices and policies governing structuring of public directories, assignment of accounts and passwords, backup schedules, and after-hours access; licensing restrictions on proprietary software; means to launch particular application programs, compile user-written FORTRAN and/or C programs, and access printers; and numerous other aspects are subject to considerable local variation. This book does not constrain local options in these matters. Instead, its users must draft a site-specific supplement, which we will refer to as the *Local Guide*, to which individuals should refer for site-specific particulars. A L^AT_EX template for that guide, specifically the one used at Lawrence, is available to users of this book, but it will require editing to reflect local practices. In particular, to give local sites flexibility in configuring their environments, we have in the book used symbols like `$HEAD`, `$IDLHEAD`, and `$NRHEAD` to stand for paths to the specific directories that sit at the head of particular directory trees. All such symbols must be expanded as described in the *Local Guide* when commands or statements illustrated in the book are submitted to the user’s machine.

With the broadest brush, Chapter 1 stands alone and focuses on a number of topics assumed as background for the rest of the book. The next several chapters introduce⁴

³Many of the packages mentioned in this list are commercial and proprietary, and the names are registered trademarks of the respective vendors. Full contact information for all mentioned packages will be found in Appendix Z.

⁴ The second edition has added the shareware programs OCTAVE and PYTHON and replaced the no-longer available commercial program MACSYMA with the shareware program MAXIMA. The addition of OCTAVE and

- Specific array processors (Chapter 2 on IDL, Chapter 3 on MATLAB, Chapter 4 on OCTAVE, Chapter 5 on PYTHON),
- Computer algebra systems (Chapter 6 on MAXIMA, Chapter 7 on MAPLE, Chapter 8 on *Mathematica*),
- Programming languages (Chapter 9—with sections on FORTRAN and C), and
- Subroutine libraries (Chapter 10 on Numerical Recipes, Chapter 12 on LSODE, Chapter 16 on MUDPACK).

The remaining chapters address several important categories of computational processing, specifically

- Solving ordinary differential equations (Chapters 11 and 12),
- Evaluating integrals (Chapter 13),
- Finding roots (Chapter 14),
- Solving partial differential equations (Chapters 15 and 16)

Each of Chapters 11, 13, 14, and 15 begins with a (generic) section in which several problems drawn from subareas of physics and using the computational technique on which the chapter focuses are laid out. Each of Chapters 11, 13, and 14 then continues with

- one or more (optional) sections in which some of the identified problems are addressed with whatever computer algebra systems are included in the version,
- a (generic) section on numerical approaches to the category of problem on which the chapter focuses, and
- several (optional) sections in which some of the problems laid out in the first section are addressed with whatever array processors, computer algebra systems, and programming languages are included in the version.

Somewhat in contrast, Chapter 15 continues with

- a (generic) section on finite difference methods (FDMs) for solving partial differential equations,
- several (optional) sections in which some of the identified problems are addressed using FDMs with each of several tools,
- a (generic) section on finite element methods (FEMs), and
- several (optional) sections in which some of the identified problems are addressed using FEMs.

Chapter 12 begins with a brief orientation to LSODE—a large and well-tested publicly available FORTRAN program for solving systems of ordinary differential equations—and Chapter 16 begins with a general discussion of multigrid techniques for solving partial differential equations and then provides an orientation to MUDPACK—a large collection of publicly available FORTRAN programs for solving partial differential equations using those techniques. Each chapter then illustrates the use of the tools on which it focuses to address several representative problems. Every chapter in the book concludes with a collection of exercises using the techniques—both symbolic and numerical—of the chapter. The appendices introduce a publishing system (Appendix A on \LaTeX) and a (UNIX/LINUX) program for producing drawings (Appendix B on TGIF).

The order of presentation *in the book* does not compel any particular order of treatment *in a course or program of self-study*. To be sure, some later sections depend on some earlier sections, but the linkages are not particularly tight. In the Lawrence context, for example, the required sophomore course *Computational Mechanics* typically covers the chapters and appendices introducing either IDL or MATLAB, either MAPLE or *Mathematica*, and \LaTeX ; and finally covers either the IDL or MATLAB and either the MAPLE or the *Mathematica* portions of the chapters on ordinary

PYTHON explain why Chapters 4–12 in the first edition have become Chapters 6–14 in the second edition.

differential equations (ODEs), integration and root finding. The chapter on Numerical Recipes, the FORTRAN and/or C portions of the chapters on programming, ODEs, integration, and root finding, the chapter on partial differential equations and the chapters on LSODE and MUDPACK are the focus of the Lawrence elective junior/senior course *Computational Physics*.

Despite the organization of the chapters by program or by computational technique involved, the focus throughout is on physical contexts. The materials are designed to be used in conjunction with intermediate level courses, not introductory courses. While the illustrations of computational procedures highlight significant physical contexts and most of the examples and suggested exercises emerge from interesting physical situations, the objective is for students to become both fluent and wary in using computational resources in application to these physical situations, not to dwell excessively on the microscopic details of numerical analysis or to teach them the underlying physics (except insofar as successful computer-based solution of problems underscores the power of the fundamental physical ideas). The students are assumed

- to have completed an introductory survey course in physics,
- to have completed courses in calculus, differential equations and, to some extent, linear algebra, and
- to be embarking on intermediate-level studies in physics

as they undertake a study of this book. We focus not so much on the set up of the situations—that is assumed to be the province of other courses—as on computer-based techniques and strategies for determining the solution once the set up is complete. Examples are drawn from classical mechanics, classical electricity and magnetism, thermal physics, quantum mechanics, curve fitting, DC and AC circuit theory, optics, and several other areas.

David M. Cook
Appleton, Wisconsin
13 February 2023

Acknowledgements

First and foremost, I wish to acknowledge the assistance and contributions of Peter Strunk, LU '89, Kristi R. G. Hendrickson, LU '91, Todd G. Ruskell, LU '91, Stephen L. Mielke, LU '92, Ruth Rhodes, LU '92, Michelle Ruprecht, LU '92, Sandra Collins, LU '93, Mark F. Gehrke, LU '93, Karl J. Geissler, LU '94, Steven Van Metre, LU '94, Alain Bellon, LU '95, Peter Kelly Senecal, LU '95, Christopher C. Schmidt, LU '97, Michael D. Stenner, LU '97, Mark Nornberg, LU '98, Scot Shaw, LU '98, Jim Truitt, LU '98, Eric D. Moore, LU '99, Teresa K. Hayne, LU '00, Danica Dralus, LU, '02, Ryan T. Peterson, LU '03, Scott J. Kaminski, LU '04, Michelle L. Milne, LU '04, Lauren E. Kost, LU '05, Claire Weiss, LU '07, and Erik Garbacik, LU '08—all Lawrence students who, as undergraduates, contributed during the first decade and a half to the evolution of this book and the accompanying manual of solutions to representative exercises. While current versions in most instances deviate—often considerably—from the initial drafts to which these students contributed, there can be no denying that their contributions have played an important role in the evolution of the use of computational resources in the Lawrence curriculum and the evolution of this book. I thank them warmly and sincerely for their assistance. All of these students have given permission for me to use whatever has over the years become of their contributions in this publication.

Second, I wish to acknowledge and thank several individuals who have provided reviews of drafts or otherwise assisted in the refinement of this book, including

- numerous students (beyond those specifically named in the previous paragraph) who have enrolled in my courses over the years and who, directly and indirectly, have made comments that have influenced this book.
- my departmental colleagues J. Bruce Brackenridge, John R. Brandenberger, Jeffrey A. Collett, Matthew R. Stoneking, and Paul W. Fontana, who have listened to my sometimes lengthy observations about matters computational and curricular and offered valuable criticisms.
- Wolfgang Christian (Davidson College), Robert Ehrlich (George Mason University), and A. John Mallinckrodt (California Polytechnic Institute–Pomona), each of whom participated in a 1990 Sloan-supported conference on computation in advanced undergraduate physics that I organized at Lawrence and each of whom as a consultant throughout the writing project has offered many suggestions and constructive criticisms.
- all those faculty members¹ who participated in one of the NSF-supported week-long workshops offered at Lawrence in the summers of 2001, 2002, and 2003 and who, in that context,

¹In July 2001: Steve Adams (Widener University), Russel Kauffman (Muhlenberg College), Marilyn Bell (Lake Forest College), Lawrence A. Molnar (Calvin College), Thomas R. Greenlee (Bethel College), Elliott Moore (New Mexico Tech), Javier Hasbun (State University of West Georgia), Joelle L. Murray (Linfield College), Derrick Hylton (Spelman College), Paula C. Turner (Kenyon College), Ross Hyman (DePaul University), Scott N. Walck (Lebanon Valley College), William H. Ingham (James Madison University), and Tim Young (University of North Dakota). In July, 2002 (first workshop): Albert Batten (United States Air Force Academy), Andrea Cox LU '91 (Beloit College), Brian Cudnik (Prairie View A and M), Craig Gunsul (Whitman College), Kevin Lee (University of Nebraska), Kam-Biu Luk (University of California, Berkeley), Daryl Macomb (Boise State University), Viktor Martisovits (Central College), Donald Miller (Central Missouri State University), Dorn Peterson (James Madison University), Robert Ragan (University of Wisconsin – Lacrosse), Shafiq Rahman (Allegheny College), Rahmathullah Syed (Norwich University), Jorge Talamantes (California State University – Bakersfield), Mark Timko (Elmhurst College), Paul Tjossem

provided a critical examination of the evolving manuscript and offered numerous suggestions for improvement—in addition to finding many previously undetected typographical glitches.

- especially Javier Hasbun (State University of West Georgia)—who had students working with almost the entire manuscript—but also William H. Ingham (James Madison University), Russel Kauffman (Muhlenberg College), Joelle L. Murray (Linfield College), Shafiq Rahman (Allegheny College), John Thompson (DePaul University), and Tim Young (University of North Dakota), all of whom used at least portions of the growing manuscript in their teaching during the academic year 2001–02 and who collectively sent me a wide spectrum of varied and valuable comments.
- several anonymous reviewers engaged by potential publishers as they evaluated my efforts, even though all potential publishers ultimately decided they could not provide the microscopic customization the book required.

Third, I wish to acknowledge considerable debt to many individuals whom I do not know but whose contributions behind the scenes have been invaluable. Chief among these individuals are

- Donald Knuth, originator of $\text{T}_{\text{E}}\text{X}$;
- Leslie Lamport, originator of $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$;
- Han The Than, originator of $\text{pdfT}_{\text{E}}\text{X}$, which provides the engine underlying `pdf latex` for creating PDF files directly from $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ source code..
- Pehong Chen and Nelson Beebe, originator and current maintainer of `makeindex` for preparing indices to $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ documents;
- Enrico Gregorio, originator and current maintainer of the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ package `imakeidx` that reduces the number of passes required to format and include an index;
- David Carlisle and Sebastian Rahtz, originators, and David Carlisle, current maintainer, of the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ package `graphicx` for including graphic images in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ documents;
- Leslie Lamport, David Carlisle and other members of the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}3$ team, authors of the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ package `ifthen` for supporting conditional statements in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ source files;
- Sebastian Rahtz and Heiko Oberdiek, originator of the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ package `hyperref` for creating linked versions of documents as PDF files;
- Paul Vojta, author of `xdvi`, a versatile on-screen previewer for the `.dvi` files produced by $\text{T}_{\text{E}}\text{X}$ and $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$;
- Till Tantau et. al., originators and current maintainers of the $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ packages `tikz` and `pgf` for creating sophisticated diagrams using $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -like inclusions in the source file;
- William Chia-Wei Cheng, author of `TGIF`, a program used to create several of the figures;
- John Bradley, author of `xv`, a program used to convert a few bit-mapped files into PostScript;

(Grinnell College), and John Walkup (California Polytechnic Institute – San Luis Obispo). In July, 2002 (second workshop): Julio Blanco (California State University–Northridge), William Briscoe (George Washington University), Paul Bunson (Lawrence University), Albert Chen (Oklahoma Baptist University), K. Kelvin Cheng (Texas Tech University), Bob Davis (Taylor University), Mirela Fetea (University of Richmond), David Grumbine (St. Vincent College), Kristi Hendrickson LU '91 (University of Puget Sound), Tom Herrmann (Eastern Oregon University), Seamus Lagan (Whittier College), Eric Lane (University of Tennessee at Chattanooga), Ernest Ma (Montclair State University), Norris Preyer (College of Charleston), Michael Schillaci (Francis Marion University), Mihir Sejpal (Earlham College), Laura Van Wormer (Hiram College), Mary Lou West (Montclair State University), and Douglas Young (Mercer University). In July, 2003: Stephen Addison (University of Central Arkansas), Mohan Aggarwal (Alabama A&M), Anne Caraley (SUNY at Oswego), William Dieterle (California University of Pennsylvania), Timothy Duman (University of Indianapolis), Brett Fadem (Colby College), Michael Gray (American University), James Meyer (St. Gregory's University), Prabasaj Paul (Denison University), David Peterson (Francis Marion University), Edward Pogozelski (SUNY at Geneseo), Chandra Prayaga (University of West Florida), Bruce Richards (Oberlin College), David Seely (Albion College), Adam Smith (Hillsdale College), Walther Spjeldvik (Weber State University), Julie Talbot (State University of West Georgia), Paul Thomas (University of Wisconsin–Eau Claire), Srinivasa Venugopalan (SUNY at Binghamton), and Alma Zook (Pomona College).

- The many contributors to the MiKTeX project;
- The developers, authors, and maintainers of WinEdt, the text editor used during the latter years of writing CPSUP; and
- The developers and maintainers of ps2pdf for converting PostScript files to PDF and the developers and maintainers of pdfcrop for pruning excessive white borders from PDF files.
- Radical Eye Software, which holds the copyright on dvips, a program for converting .dvi files to PostScript.

Quite simply, this project would have been impossible without the availability of these several programs and utilities, each of which played a necessary role behind the scenes in preparing or processing the files from which, ultimately, a printable PostScript or PDF file for the finished book emerged.

Fourth, I point out that the names of several pieces of commercial software are, in fact, trademarks or registered trademarks belonging to the vendors of those software products. Each such trademark is identified at its first occurrence in the text proper, and detailed contact information for every vendor is compiled in Appendix Z.

Fifth, I acknowledge the following specific permissions, each of which is more fully explained at the point in the text where the permission is explicitly invoked. In particular, I thank

- The MathWorks, Inc., for permission to incorporate in this book and distribute IDL source code for the routines `ludiffeq_23` and `ludiffeq_45`, which code uses algorithms patterned after those used in 1991 in the MATLAB routines `ode23` and `ode45`.
- Wayne Landsman, author of the IDL routines `qsimpson` and `trapzd` in the IDL Astronomy User's Library, for permission to use those routines as the basis for the routines `luqsimp` and `lutrapzd` and to distribute the source code for `luqsimp` and `lutrapzd` as supplements to this book.
- Research Systems (later Exelis Visual Information Solutions and now part of Harris Geospatial Solutions), Incorporated, for permission to use portions of any RSI-supplied and/or edited `.pro` code—most particularly evident in RSI contributions to `ludiffeq_23.pro`, `ludiffeq_45.pro`, and `luqsimp.pro`—and to use the IDL name and trademark.
- Numerical Recipes Software (a) for permission to use the names and calling sequences of several Numerical Recipes routines² at various places in this book, (b) for permission to refer to the C header files `nr.h` and `nrutil.h` and the file `nrutil.c` containing assorted utilities used by various C recipes, and (3) for permission to use the names and calling sequences of several IDL routines that are derived from Numerical Recipes routines (and for the use of which Research Systems Incorporated has permission from Numerical Recipes Software).
- William Chia-Wei Cheng, author of TGIF, for permission to reproduce in the appendix on that program several of the icons used in its many screen displays.

Sixth, I wish to acknowledge and thank the W. M. Keck Foundation for three grants [#880969 (1988), #931248 (1993), #011795 (2001)], the National Science Foundation for three grants [USE-8851685 (ILI-1988), DUE-9350667 (ILI-1993), and DUE-9952285 (CCLI-EMD-2000)], and Lawrence University for matching and other support.³ A small conference held at Lawrence in 1990 and supported by the Alfred P. Sloan Foundation stimulated thinking among the participants about

²Specifically `flmoon`, `xflmoon`, `caldat`, `julday`, `xjulday`, `avevar`, `xavevar`, `rk4`, `xrk4`, `rkqs`, `rkck`, `mmid`, `bsstep`, `rkdumb`, `odeint`, `trapzd`, `xtrapzd`, `qtrap`, `xqtrap`, `qsimp`, `qromb`, `polint`, `rtbis`, `xrtbis`, `rtnewt`, `xrtnewt`, `rtsafe`, `xrtsafe`, `zbrak`, `gaussj`, `ludcmp`, `lubksb`, `tridag`, `svdcmp`, `svbksb`, `mnewt`, `newt`, and `broydn` (both in FORTRAN and in C).

³Any opinions, findings, and conclusions or recommendations expressed in this book are those of the author and do not necessarily reflect the views of any of these granting foundations or agencies.

uses of computers in upper-division undergraduate physics. All of these grants have contributed in many ways to the developments at Lawrence that have culminated in the writing of this book. In particular, the NSF CCLI-EMD grant made in February, 2000, supported my sabbatical while I finalized the text of (the first edition of) this book. That grant also supported four week-long summer faculty workshops that have, on the one hand, provided constructive feedback on a succession of drafts and, on the other hand, enhanced awareness nationally of this book and of the developments at Lawrence.

Finally (and most important of all), I wish to express deep gratitude for the decades during which my wife, Cynthia, has shown patience, understanding, and support beyond any possibility of my ever fully knowing. In truth, I do not know how to thank her.

Disclaimer

The statements described in the various chapters of this book have been tested extensively but have certainly not been tested with all versions of all software packages on all possible platforms with all possible versions of the underlying operating systems. Differences from version to version of the software packages, from operating system to operating system, and from platform to platform exist. This brief section identifies the versions of the various programs that have been tested and the operating systems and platforms on which those tests have been carried out. That the behavior of other combinations of version, operating system, and platform will conform in every detail to that herein described can, of course, not be guaranteed. One can, however, have some confidence that the behavior in combinations not explicitly tested will not differ enormously from that described herein—except that newer versions of a software package may well have features not implemented in earlier versions (and occasionally a feature or specific syntax available in an earlier version has been removed altogether from more recent versions). With reasonable confidence, one can presume that the commands and syntax and features described in this book will work on other platforms with the tested versions of the programs and with subsequent versions. Statements herein that exploit features implemented for the first time in the tested versions will, of course, not be accepted in earlier versions, but those “glitches” should not be numerous or extensive. Where, in the months and years since the original draft was created, I have become aware of such glitches, I have made appropriate updates and subsequent productions have incorporated those updates.¹ Nothing, however, assures that I have identified *all* such glitches.

That disclaimer having been stated, I now present for each program a brief tally of the version(s) tested and the platform(s) and operating system(s) on which those tests have been carried out:

- The MAXIMA codings herein have been fully verified with
 - MAXIMA Version 5.36.1 and wxMAXIMA Version 15.04.0 on a Hewlett-Packard platform running Windows 7,
 - MAXIMA Version 5.38.1 and wxMAXIMA Version 16.04.2 on a Hewlett-Packard platform running Windows 7, and
 - MAXIMA Version 5.36.1 and wxMAXIMA Version 12.01.0 on a Hewlett-Packard platform running the Fedora 17 implementation of LINUX.

In addition, these codings have been spot-checked with

- MAXIMA Version 5.38.0 and wxMAXIMA 16.04.1,
- MAXIMA 5.39.0 and wxMAXIMA 16.12.0, and

¹The date of production of each version of *CPSUP* is displayed at the top of the cover page on that version. I have maintained a dated list of edits made to the source files, so changes made after the date of production of a particular version of *CPSUP* and a subsequent production of that version can readily be identified for anyone who wishes to update an outdated production. Generally, updated productions fairly promptly replace the previous production at psrc.aapt.org/curricula/cpsup. Versions dated between 10 and 31 January 2021 provide the base. Edits made after 31 January 2021 are recorded in the file of edits.

- MAXIMA 5.42.0 and wxMAXIMA 18.10.1

on a Hewlett-Packard platform running Windows 10.

- The MAPLE codings herein have been fully verified with MAPLE Version 16 on a Hewlett-Packard platform running Windows 7 and on a Hewlett-Packard platform running the Fedora 17 implementation of LINUX. These codings have also been spot-checked with MAPLE Version 17 on a Hewlett-Packard platform running Windows 10.
- The *Mathematica* codings have been fully verified with *Mathematica* Version 11.3 on a Hewlett-Packard platform running Windows 7 and on a Hewlett-Packard platform running the Fedora 17 implementation of LINUX. The *Mathematica* codings have also been spot-checked with *Mathematica* 12.0 on a Hewlett-Packard platform running Windows 10.
- The IDL details codings have been fully verified with IDL Versions 8.3 and 8.5 on a Hewlett-Packard platform running Windows 7 and a Hewlett-Packard platform running the Fedora 17 implementation of LINUX. These versions of IDL have also been spot-checked on a Hewlett-Packard platform running Windows 10.
- The MATLAB codings have been fully verified with MATLAB Version R2012a on a Hewlett-Packard Platform running Windows 7 and a Hewlett-Packard platform running the Fedora 17 implementation of LINUX. These codings have also been spot-checked on a Hewlett-Packard platform running Windows 10.
- The OCTAVE codings have been fully verified with
 - OCTAVE Version 4.0.0 on a Hewlett-Packard platform running Windows 7,
 - OCTAVE Version 3.6.3 on a Hewlett-Packard platform running the Fedora 17 implementation of LINUX, and
 - OCTAVE Version 4.0.3 on a Hewlett-Packard platform running the Fedora 25 implementation of LINUX.

OCTAVE Version 4.2.2 has been spot-checked on a Hewlett-Packard platform running Windows 7, and OCTAVE Versions 4.0.0 and 5.2.0 have been spot-checked on a Hewlett-Packard platform running Windows 10.

- Except where otherwise noted in the text, the PYTHON codings have been fully verified with
 - PYTHON 2.7.16 from the Anaconda2 distribution installed on a Hewlett-Packard platform running Windows 10, using the Anaconda2 prompt and also using the Anaconda2 Python Shell.²
 - PYTHON 3.7.3 from the Anaconda3 distribution installed on a Hewlett-Packard platform running Windows 10, using the Anaconda3 prompt and also using the Anaconda3 Python Shell.³

The codings in the PYTHON chapter and some of the PYTHON codings in other chapters have been verified with PYTHON 3.9.13 from the Anaconda3 distribution installed on a Hewlett-Packard platform running Windows 10, using the Anaconda3 Python shell.

- The Numerical Recipes codings have been fully verified with Numerical Recipes Version 2.10 only on a Hewlett-Packard platform running the Fedora 17 implementation of LINUX. Those codings have also been spot-checked on a Hewlett-Packard platform running Windows 10 using 64-bit GNU Fortran Version 7.1.0 and 64-bit GNU C Version 7.1.0.

²See the *Local Guide* for ways to bring up the prompt and the shell in your environment.

³See the previous footnote.

- The LSODE codings have been fully verified with LSODE whose README file bears the date 30 March 1987 only on a Hewlett-Packard platform running the Fedora-17 implementation of LINUX. The codings have also been spot-checked with LSODE whose README file (`opkd-sum`) bears the date 20 June 2001 on a Hewlett-Packard platform running Windows 10 with 64-bit GNU Fortran Version 7.1.0.
- The L^AT_EX details apply specifically to L^AT_EX 2_ε with the MiK_TE_X implementation on a Hewlett-Packard platform running Windows 7 and on a Hewlett-Packard platform running Windows 10. L^AT_EX normally responds to the same source code on all platforms.
- The TGIF details apply specifically to TGIF Version 4.2 (patchlevel 5) on a Hewlett-Packard platform running the Fedora 17 implementation of LINUX. (TGIF is exclusively a UNIX package.) Other versions and patchlevels and other platforms will surely have similar behavior, but may not conform *exactly* to the behavior here described.
- The MUDPACK codings have been fully verified with MUDPACK Version 5.0.1 whose README file bears the date 6 December 2011 only on a Hewlett-Packard platform running Windows 10 with 64-bit GNU Fortran Version 7.1.0.

