eral relativity and Edwin Hubble's observations of receding nebulae.

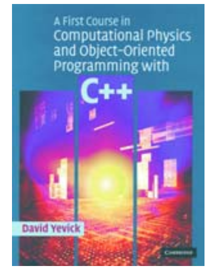
I noticed only a few errors or questionable interpretations; let me mention three. Nicolaus Steno, the innovative geologist and anatomist, is wrongly characterized as a Danish prelate, although he later became ordained as a Catholic priest. Alexander Friedmann's paper on the dynamic solutions of Einstein's cosmological field equations dates from 1922, not 1921. And Park states that Ptolemy probably conceived his heavenly spheres to be mathematical devices rather than real entities. This may be the impression that the *Almagest* leaves, but historians of astronomy now believe that Ptolemy did think of the spheres as real, as evidenced in his later work, the *Planetary Hypotheses*, which Park fails to mention.

All the same, *The Grand Contraption* is a masterful presentation of the long timelines in the history of cosmology. It is a remarkable book on the development of the worldview from chaos to cosmos, and from the most ancient cultures to modern time.

Helge Kragh
University of Aarhus
Aarhus, Denmark

A First Course in Computational Physics and Object-Oriented Programming with C++

David Yevick
Cambridge U. Press, New York, 2005. $70.00 (403 pp.). ISBN 0-521-82778-7, CD-ROM



The use of computers is ubiquitous in physics research. Even the most analytical theorists need to conduct numerical calculations to obtain specific results. Thus the physics curriculum should contain some instruction in understanding and applying computational tools. Such instruction is sometimes built into traditional courses, but many departments now offer standalone courses in computational physics. The title of David Yevick's text, *A First Course in Computational Physics and Object-Oriented Programming with C++*, suggests that this text would be appropriate for one of these newly developed computational physics courses. However, it is not. Instead it is a book on programming in C++ and is designed for the traditional scientific computing courses that were developed before the more recent computational physics courses.

One type of computational physics course involves a physics version of the numerical-methods courses taught in mathematics departments. The focus is on algorithm developments and their application to physical systems, and students frequently must already have some ability to program. Textbooks for such courses include Alejandro L. Garcia's second edition of *Numerical Methods for Physics* (Prentice Hall, 2000), Tao Pang's *An Introduction to Computational Physics* (Cambridge U. Press, 1997), and Paul L. DeVries's *A First Course in Computational Physics* (Wiley, 1994). A text that attempts to teach programming, along with the computational physics, is Rubin H. Landau's *A First Course in Scientific Computing: Symbolic, Graphic, and Numeric Problem Solving Using Maple, Java, Mathematica, and Fortran* (Princeton U. Press, 2005). This text is a hands-on workbook instead of a traditional text, and has the unique feature of discussing a number of different programming languages.

Another type of computational physics course treats numerical methods as tools to be used in computer simulations. This type starts with the physics, develops models, implements those models using numerical algorithms, and finally extracts information and insight from the resulting simulations. Undergraduate texts that follow this approach are the third edition of *An Introduction to Computer Simulation Methods: Applications to Physical Systems* (Addison-Wesley, 2006), which I cowrote with Harvey Gould and Wolfgang Christian, and the second edition of *Computational Physics* (Prentice Hall, 2006) by Nicholas J. Giordano and Hisao Nakanishi. The former discusses how to code simulations using Java and the recently developed Open Source Physics Library.

Yevick, a professor of physics at the University of Waterloo in Canada, has been involved in the numerical modeling of various aspects of optical communication systems for the past 25 years. He has also taught scientific programming for seven years in the university's science and engineering departments. In his book, Yevick covers mostly programming in C++. He presents only a cursory survey of a few algorithms, much less than one would find in any of the previously mentioned books. For example, the only ordinary differential equation solver mentioned is the Euler algorithm. Monte Carlo methods are briefly discussed in a rather abstract context. A student would never learn from the text that those methods are a major tool in statistical physics.

The strength of the text is its careful development of the C++ programming language, with a focus on those aspects that are useful in scientific programming. The text would be helpful for graduate students who need to learn C++ for their research. Also, it would be a useful reference for those already using C++.

In addition to providing a detailed discussion of the C++ elements necessary for scientific programming, Yevick covers the advantages of the object-oriented programming (OOP) paradigm, hardware and software architectures, and program optimization. However, some of the discussion on those topics will only make sense to those already seriously involved in scientific computing. Coverage of why OOP is preferable to procedural languages is good, but I did not find the toy example in the text convincing. It is difficult to make the case for OOP using simple examples. The strength of OOP is in managing large projects or libraries, such as those used to generate graphical user interfaces, animation, graphics, and large-scale simulations. But these topics are not discussed in any detail in the text. Another obvious omission is the topic of parallel computing.

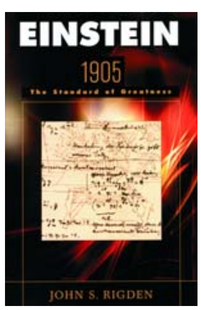
Nevertheless, *A First Course in Computational Physics and Object-Oriented Programming with C++* contains most of what one would need to program in C++ and includes many helpful exercises. Physical science and engineering students who are either very diligent or have some background in programming could learn C++ very well from Yevick's text. But many students with absolutely no programming background would need more basic assistance than the book provides.

Jan Tobochnik
Kalamazoo College
Kalamazoo, Michigan

Einstein 1905: The Standard of Greatness

John S. Rigden
Harvard U. Press, Cambridge, MA, 2005. $21.95 (173 pp.). ISBN 0-674-01544-4

Finally, 2005 ended and we survived the 1905 centennial festivities. We're left with various tributes to Albert Einstein's legacy, including John S. Rigden's *Einstein 1905: The Standard of Greatness*. The author aptly describes his book as a celebration of Einstein's achievements. In it, he introduces Einstein's famous works of 1905, seminal contributions that include the notion that light is absorbed and emitted as particles, a derivation of the size of sugar molecules in a liquid, the causes behind the erratic motions of pollen grains suspended in water, and an understanding of the effects of relative motion on our determinations of time and energy. I characterize Einstein's accomplishments that way because Rigden writes in very palpable terms. He sensibly illustrates Einstein's concepts with analogies to molasses, sand, bagged sugar, cars, fence posts, and airplanes.



The book is comparable to John Stachel's classic *Einstein's Miraculous Year: Five Papers That Changed the Face of Physics* (Princeton U. Press, 2005 [1998]), which includes translations of Einstein's 1905 papers. Yet Rigden's is much more accessible to readers who lack the knowledge or patience to understand those papers. Rigden was editor of the *American Journal of Physics* from 1978 to 1988. He is now an honorary professor of physics at Washington University in St. Louis, Missouri, and coeditor of the outstanding history journal *Physics in Perspective*.

Rigden shares insights about Einstein's ensemble of papers. For instance, it's common to hear that physicists cite Einstein's dissertation on molecular dimensions far more frequently than his papers on the light quantum and on relativity. Rigden explains that the hierarchy of citations is misleading because some papers are so fundamental that they become taken for granted.

He also argues that Einstein's great success stemmed from his approach of trying to understand God. Statements such as "In 1905, Einstein had a direct line to God's thoughts" stem partly from Einstein's occasional remarks; but it would be fair to at least mention that Einstein did not mean those remarks literally. Einstein was an agnostic who rejected the notion of a personal god; he argued that "no idea is divinely inspired," and that no superhuman authority rules ethics. So, to better answer the question "How did Einstein do it?" we might turn to recent works by historians Robert Rynasiewicz and Jürgen Renn.

Portions of the book resemble what Friedrich Nietzsche called monumental history: Elevated stories canonize the departed hero's feats. Rigden contends that in 1905 Einstein set "the standard of greatness" and distinguishes Einstein from other physicists. For example, Einstein "recognized truths about the world by pure acts of mind." But what about Einstein's explanation that intuition stems from reflection on accumulated empirical knowledge? We also read that "He saw Nature as it is."

In my opinion, the celebratory thread is too uncritical. Einstein himself repeatedly complained that the popular overestimation of his achievements was "simply grotesque." He rejected such admiration as having arisen through "no merit of my own." Most retort that Einstein was being humble. Some of us believe he was just being honest.

Minor mistakes on names and dates, and passing slips like the statement that at superluminal speeds,