introductory material on surfaces, there are subsections on Bézier surfaces, B-spline surfaces and Beta-spline surfaces, and the final one on Hermite and implicit cubic spline surfaces. Each subsection ends with comments on the implementation of the presented methods in the enclosed set of programs called Spline Guide, designed for an IBM PC or compatible machines. The interpolation and smoothing programs for the first two chapters are written in Fortran, the programs for the curves and surfaces of Chapters 3 and 4 are coded in C.

There are no references at the end of each subsection or chapter, only one list at the end of the book, which is very short and leaves out quite a number of pertinent books on splines. Therefore, in my opinion, there is no help for an uninitiated reader to find out about other splines, for example quadratic ones, or about proofs, or even who introduced the concepts and algorithms, which would assist in further studies. The reader is supposed to be content with the material as it is presented in the book and implemented in the programs. Finally, there are instances where the usage of English in the text should have been more carefully monitored by the publisher.

EWALD QUAK
DEPARTMENT OF MATHEMATICS
TEXAS A & M UNIVERSITY
COLLEGE STATION, TX 77843-3368

42[65-01, 65C05, 65C10]—*A primer for the Monte Carlo method*, by Ilya M. Sobol′, CRC Press, Boca Raton, FL, 1994, xx+107 pp., $21\frac{1}{2}$ cm, softcover, \$41.95

This is the English translation of a book which first appeared in Russian in 1968 and for which earlier translations into English have been brought out, e.g. by Mir Publishers, Moscow, under the title "The Monte Carlo Method". The book is still useful as a brief introduction to the subject for students. However, the bibliography is rather poor since it contains only 10 items.

H.N.

43[65-01, 68-01, 68U05]—*An introduction to scientific, symbolic, and graphical computation*, by Eugene Fiume, A K Peters, Wellesley, MA, 1995, xvi+306 pp., $23\frac{1}{2}$ cm, \$49.95

This undergraduate textbook is positioned at the interface between computer science and mathematics. It can be seen as an applied mathematics course with the field of application being computer graphics. Computer science students will get a useful introduction to the nontrivial mathematical prerequisites of computer graphics, and mathematics students will be exposed to an interesting field of application that is not taken from science or engineering. The topics are widely applicable in many fields, and this justifies the general title of the book.

A theme of the book is that symbolic and graphical uses of computers have become more or less equal partners with strictly numerical processing in developing solutions of scientific and other kinds of problems. These seminumerical uses have been present alongside numerical computation since the beginning of digital electronic computing but their impact has been diminished until recently by the lack of affordable and sufficiently powerful interactive systems. Now such systems as Maple, Mathematica and Matlab are installed routinely on personal computers and workstations.

The book employs Maple to introduce students to this enlarged view of scientific computing (which has already been adopted by much of the professional community of scientific computer users). Many examples and exercises are presented in terms of Maple sessions and procedures. A much smaller number are presented as programs in Turing or C. In comparison to the latter two languages, Maple facilitates the rapid generation of mathematical developments, both symbolically and numerically, together with ready access to graphical displays. This is particularly appropriate in a course that delivers a steady flow of applicable mathematical concepts.

The undergraduate character of the book is indicated by its avoidance of linear algebra and multivariate calculus. The author states that "single-variable calculus is the only real prerequisite (or corequisite), although some programming experience would be helpful." Unsurprisingly, there are no proofs. The descriptions, derivations and plausible arguments are all quite convincing.

The book contains the following chapters.

Chapter 0, Mathematical Computation, provides background material and illustrative examples of symbolic, graphical and numerical computation.

Chapter 1, Representation of Functions, discusses explicit, implicit and parametric representations with emphasis on polynomials. Examples include plane curves, space curves, and geometric figures such as the Koch snowflake. A good introduction to line and circle rendering on a raster graphics device includes the step-by-step development of an accurate and efficient integerized algorithm.

Chapter 2, Interpolation, is motivated by the space curve traced by the flight of a fly and sampled at discrete points by a "Fly Tracker." The emphasis is on piecewise polynomial interpolation with each polynomial parameterized for evaluation on the domain $[0,1]$. Accordingly, reparameterization and change-of-basis are discussed in some detail. The cubic Catmull-Rom interpolant (important in computer graphics and related to the cubic Hermite interpolant) is developed and compared favorably to Lagrange interpolants because of its smooth transition between curve segments.

Chapter 3, Approximation and Sampling, contains two parts. In the first part interpolation is compared to approximation by considering cubic Catmull-Rom and cubic B-spline bases. In the second part the subject of signal processing is introduced. Filters and their connection via convolution to Fourier series and integrals are discussed with many illustrative examples. The chapter concludes with a description of the sampling theorem.

Chapter 4, Computational Integration, derives quadrature rules based on the midpoint, trapezoid, Simpson and Catmull-Rom formulas. Their performance is compared numerically on four simple integrals (including an arc length). The chapter concludes with a discussion of Monte Carlo integration.

Chapter 5, Series Approximations, introduces the floating-point representation and rounding error of arithmetic operations. Then Taylor expansions are introduced and used to derive the familiar truncation error formulas for the quadrature rules presented in Chapter 4. Finally, the Fourier inversion formulas and the sampling theorem are revisited for further discussion.

The subject of the final Chapter 6, Zeros of a Function, is motivated by intersection problems in geometric modelling. Both symbolic and numerical approaches are discussed.

From the foregoing description it can be seen that this textbook covers a wealth of material from a somewhat unusual viewpoint. A number of misprints, mostly minor, were detected. Some of the mathematical notation seems excessive, or excessively pedantic, but not more than is usually observed in texts at this level. The index failed to provide the needed page reference on a few occasions when it was consulted.

The author maintains a World Wide Web site to provide services for and obtain feedback from users. Among the services provided are programs and procedures related to topics in the book, and an errata sheet.

D.W.L.

44[65-02, 65Yxx]—*High performance computing—Problem solving with parallel and vector architectures*, Gary W. Sabot (Editor), Addison-Wesley, Reading, MA, 1995, xvi+246 pp., 24 cm, $45.14

This book attempts, by means of a number of case studies from scientific computing, to illustrate the techniques for developing efficient programs on high-performance computers. Applications from shock-wave physics and weather prediction have been included. These applications, as well as algorithms from numerical linear algebra, dynamic tree searching, graph theory, and mathematical programming have been implemented in several programming languages, and parallel architectures and their performance analyzed and reported. The chapters of the book correspond to contributed case studies whose intent is to identify and address high-performance computing issues at the application, algorithm, language, and machine levels. The issues of portability and scalability are addressed in the last chapter of the book and within each case study. The spectrum of target machines covered ranges from SIMD and MIMD massively parallel machines to vector machines and network of workstations. The parallelization techniques and methodologies presented are tightly coupled with the particular case studies selected. The material of the book could be useful to application software developers and could provide supplementary topics for an introductory graduate course in high-performance computing.

Elias N. Houstis
Department of Computer Sciences
Purdue University
West Lafayette, IN 47907-1398

45[65F05, 65F15, 65F20, 65F35, 65Y05, 65Y20, 65Y99]—*LAPACK Users' guide*, by E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, second edition, SIAM, Philadelphia, PA, 1995, xx+325 pp., 28 cm, softcover, $28.50

As a successor to the software packages LINPACK and EISPACK, LAPACK provides more efficient and accurate routines for the solutions of dense systems of linear equations, least squares problems, eigenvalue and singular value problems. The second edition of the User's Guide is mainly for the September 30, 1994 release of version 2.0 of the package. In addition to what the first edition offers, the new edition provides pointers to the guides of several related packages including