Ted G. Lewis is associate professor of computer science at Oregon State University. He is the coinventor of the Tausworthe-Lewis-Payne algorithm (used extensively in cryptography) and is the developer of the first retargetable microprogramming language system for implementing efficient microprograms. Lewis has published extensively in the areas of software engineering, distributed operating systems, and computer organization and architecture, authoring a number of books on personal computers, software engineering, and Pascal. His address is 4400 Sulphur Springs Rd., Corvallis, OR 97330.

John B. Munson is corporate vice president for technical operations at System Development Corporation, in Camarillo, California. He is responsible for all technical activities for the company, with special emphasis on advanced software engineering activities. Munson has over 25 years experience in data processing as a programmer and as a manager of programming projects, primarily large, military real-time computer systems. He has a BA in math, a graduate certificate in business management, and is a senior member of the IEEE. His address is System Development Corp., 5151 Camino Ruiz, MS 02/B26, Camarillo, CA 93010.

John D. Musa is currently supervisor of computer measurements, capacity planning, and software reliability at AT&T Bell Laboratories. He has had extensive experience managing software projects as well as experience in the fields of software engineering, computer graphics, data management, human factors, system engineering, and management information systems. Musa was recently elected to the Governing Board of the Computer Society. He holds BA and MS degrees in electrical engineering from Dartmouth. His address is Bell Telephone Laboratories, Room 3A-332, Whippnany Rd., Whippnany, NJ 07981.

Dennis R. Allison is currently a self-employed independent consultant and part-time lecturer in the Computer Systems Laboratory at Stanford. His areas of interest include personal computing and computer literacy, programming language and environment design, computer architecture, compiler design, software engineering, parallel machines, and algorithm analysis. Allison has served on the Computer Society Governing Board and is chairman of the Technical Committee on Microprocessors and Microcomputers. He holds a degree in physics from the University of California at Berkeley. His address is 169 Spruce Ave., Menlo Park, CA 94025.

Edith W. Martin is deputy under secretary of defense for research and advanced technology in the US Department of Defense. Her current initiatives include technologies essential to military-critical computer systems, such as Ada, STARS, and VHSIC; export control and technology transfer; and science and engineering education. Martin received her PhD and MS degrees in information and computer science from the Georgia Institute of Technology. Her address is OUSDRE (R&T), The Pentagon, Room 3E-114, Washington, DC 20301.

William McKeeman is chairman of the faculty and a professor in the School of Information Technology at the Wang Institute of Graduate Studies. He was most recently with Xerox PARC as a visiting scientist. He has coauthored several books and has published papers in the areas of compilers, programming language design, and programming methodology. McKeeman received a PhD from Stanford University in computer science, an MA in math from George Washington University, and a BA in math from the University of California at Berkeley. His address is Wang Institute of Graduate Studies, Tyng Rd., Tyngsboro, MA 01879.

Robert M. Poston is president of Programming Environments, Inc. He has 20 years of experience managing and developing software for process control, simulation, radar, communication, numerical control, business, and operating systems. From 1978 to 1982, Poston chaired the IEEE Software Engineering Standards Subcommittee. Currently, he is the standards coordinator between the IEEE-CS and the American National Standards Committee, X3, and serves as technical director for IEEE seminars on software engineering standards. His address is Programming Environments, Inc., 58 Lambert Johnson Dr., Ocean, NJ 07712.
Richard H. Eckhouse, Jr., cofounded and is currently vice president of Moco, Inc., where he conducts biomedical research related to human performance. Previously, he has worked with DEC and IBM Research at Yorktown Heights, in addition to consulting for several industrial and academic institutions. Eckhouse has published two books, *Minicomputer Systems: Organization, Programming, and Applications* and *Computer Programming and Architecture*. He has taught computer science at a number of major universities and is an active ACM member. His address is Moco, Inc., 5 Champlain Dr., Hudson, MA 01749.

Wiley R. McKinzie is currently director of the School of Computer Science and Technology at Rochester Institute of Technology. His current interests include language processors, personal computing software. During his 23 years in the computing field, he has been a systems programmer, systems analyst, data processing manager, teacher, and consultant. McKinzie received a BA in mathematics from the University of Wichita and an MS in computer science from SUNY, Buffalo. His address is Rochester Institute of Technology, One Lomb Memorial Dr., Rochester, NY 14623.

Mary Shaw is currently an associate professor of computer science at Carnegie-Mellon, where she has been on the faculty since 1971. Her primary research interests are in programming systems and software engineering, developing and evaluating software. She received a BA in math from Rice University in 1965 and a PhD in computer science from Carnegie-Mellon University in 1972. Shaw is a member of the ACM, IEEE-CS, and the New York Academy of Sciences. Her address is Computer Science Department, Carnegie-Mellon University, Schenley Park, Pittsburgh, PA 15213.

**Guidelines for Authors**

**Audience**

*IEEE Software* is a publication of the IEEE Computer Society. Its readers specify, design, document, test, use, maintain, purchase, engineer, sell, teach, research, and manage the production of software or systems that include software. This readership uses *IEEE Software* to learn about tools, techniques, concepts, and current research and development activities to aid them in understanding software and software-related issues in greater depth.

**Content**

*IEEE Software* emphasizes current practice and experience together with promising new ideas that are likely to be used in the near term. It is directed to the practice of the software profession. We welcome papers on topics across the software spectrum. Sample topics are:

- data engineering and database software,
- programming environments,
- languages and language-related issues,
- knowledge-based and decision support systems,
- program and system debugging and testing,
- distributed, network, and parallel systems,
- education of software professionals,
- design, development, and programming methodologies,
- algorithms, their analysis and pragmatics,
- performance measurement and evaluation,
- program and system reliability, security, and verification,
- software-related social and legal issues, and
- human factors and software metrics.

Articles describing how software is developed in specific companies, laboratories, and university environments, as well as articles describing new tools to aid in the software and system development process, are welcome. The intent is to provide the reader with information on advances in software technology, specifics on novel features and applications, contrasts in designing and programming in the small, discussions of limitations and failures, and an awareness of the trends in this rapidly evolving area. Tutorials, survey articles, standards, guided tours through descriptions of projects, designs, or algorithms, and case studies are also encouraged.

**Originality**

All manuscripts submitted for publication should be original. Articles published in other magazines or journals will not be considered for publication in *IEEE Software*. However, papers published in conference proceedings will be considered if they are of importance or contain interesting new results. If a manuscript has been published or is being considered for publication elsewhere, such information must be included in the submission cover letter. The authors are responsible for securing all necessary clearances.

**Standards for content and writing style**

All manuscripts considered for publication must pass a refereeing process consistent with that of articles submitted to an archival publication. However, the style of presentation is somewhat different. The articles should be written in a concise, clear, and organized manner, much the same as the lively narrative style of articles appearing in our sister publication, *Computer*. Every effort should be made to assist the reader in understanding the concepts presented in the text. Thus the use of examples, diagrams, figures, and photographs that aid in this process is encouraged.

If there are terms or concepts that may not be understood by the majority of the readers of your article, you should consider submitting independent clarifications of these terms and concepts, preferably with examples. These would be published in "break-out" boxes within the text of the article. Readers familiar with the concepts would pass over the boxes, while those who need clarification would stop and read their contents.

The article should be accompanied by a list of keywords from *Computing Reviews* classification system identifying the key areas of article content.

January 1984