September
Assuring Better Quality
Quality assurance saves money and time despite the initial investment. What are current methods? How do you manage and plan them? How can you implement existing models? How can you transition to new models? Find out.

Plus:
• Program Readability
• Test-Case Generation
• Distributed-Software Engineering

Ad Closing: August 1

November
Putting It All Together
Programming in the large and in the small. By the many and by the few. For the many and for the few. There are several approaches to software design. See how these approaches are being integrated into single environments. To complement this issue, Computer magazine will feature a series of related tutorials.

Plus:
• Multiple-Language Support
• Software-Engineering Research Efforts Status

Ad Closing: October 1

January
Parallel Programming
Save time and resources by breaking tasks into pieces and processing them concurrently — that's the promise of parallel programming. Explore languages, environments, performance, and algorithms to make it possible.

Plus:
• Legal aspects of software
• Modeling software

Ad Closing: December 1

March
CASE Tools
Can the software development cycle — from requirements to design to programming to maintenance — ever be truly automated? Software engineers are developing programs that integrate a modular approach with graphics and artificial intelligence in an effort to make CASE a reality.

Plus:
• Workstation software
• Scientific software

Feature Article Closing: August 1
Ad Closing: February 1

May
Compiling Environments
No longer must you search through hundreds of lines of code to find the obscure error that aborted your compilation. With new graphical, interactive compilers, you can easily find mistakes and test your methods while developing your code. See what these environments can do for you.

Plus:
• Porting techniques
• Estimating Software Cost

Feature Article Closing: October 1
Ad Closing: April 1

July
New Generations
Traditional software languages are making way for fourth- and fifth-generation languages that take advantage of new approaches and architectures. These languages promise versatility, flexibility, and capability not now available.

Plus:
• How Technology Affects Software Practice

Feature Article Closing: December 1
Ad Closing: June 1

Note: Article schedules may change.


Article submissions: Send eight double-spaced article submissions to Ted Lewis, Editor-in-Chief, IEEE Software, c/o Computer Science Dept., Oregon State University, Corvallis, OR 97331; Compmail+ t.lewis, CSnet lewis@oregon-state. Maximum length is 30 double-spaced typed pages (55K bytes), including references, figures, and tables. Maximum number of references is 10. The style should be active, concise, and direct. The aim of the article should be to help software engineers understand and apply the concepts to their work. All articles are peer-reviewed and edited.

Department submissions: Send double-spaced department submissions and press releases (news items, standards reports, product announcements, and conference schedules) to Angela Burgess, Managing Editor, IEEE Software, 10662 Los Vaqueros Cir., Los Alamitos, CA 90720.