<<软件工程>>

图书基本信息

书名：<<软件工程>>

13位ISBN编号：9787302041399

10位ISBN编号：7302041393

出版时间：2001-1

出版时间：清华大学出版社

作者：Roger S.Pressman

页数：860

前言

When a computer software succeeds——when it meets the needs of the peoplewho use it, when it performs flawlessly over a long period of time, when it iseasy to modify and even easier to use——it can and does change things for the better.But when software fails——when its users are dissatisfied, when it is error prone, whenit is difficult to change and even harder to use——bad things can and do happen. Weall want to build software that makes things better, avoiding the bad things that lurkin the shadow of failed efforts. To succeed, we need discipline when software isdesigned and built. We need an engineering approach. In the 20 years since the first edition of this book was written, software engineer-ing has evolved from an obscure idea practiced by a relatively small number of zealotsto a legitimate engineering discipline. Today, it is recognized as a subject worthy ofserious research, conscientious study, and tumultuous debate. Throughout the indus-try, software engineer has replaced programmer as the job title of preference. Softwareprocess models, software engineering methods, and software tools have been adoptedsuccessfully across a broad spectrum of industry applications. Although managers and practitioners alike recognize the need for a more disci-plined approach to software, they continue to debate the manner in which disciplineis to be applied. Many individuals and companies still develop software haphazardly, even as they build systems to service the most advanced technologies of the day.Many professionals and students are unaware of modern methods. And as a result, the quality of the software that we produce suffers and bad things happen. In addi-tion, debate and controversy about the true nature of the software engineeringapproach continue. The status of software engineering is a study in contrasts. Atti-tudes have changed , progress has been made, but much remains to be done beforethe discipline reaches full maturity. The fifth edition of Software Engineering：A Practitioners Approach is intended toserve as a guide to a maturing engineering discipline. The fifth edition, like the foureditions that preceded it, is intended for both students and practitioners, retaining itsappeal as a guide to the industry professional and a comprehensive introduction tothe student at the upper level undergraduate or first year graduate level. The formatand style of the fifth edition have undergone significant change, making the presen-tation more reader-friendly and the content more easily accessible.

<<软件工程>>

内容概要

《大学计算机教育国外著名教材系列·影印 软件工程实践者之路(第5版)》一书，20年来，受到了学习软件工程的学生和该领域的专业人士的极大欢迎和推崇，在软件工程教学中被奉为经典。
《软件工程》第5版在内容设计上做了较大的改变，更新了关键内容，篇幅扩充到32章，重点阐述了现在很多人称之为“21世纪工程准则”中所包含的每项内容。
《软件工程》的版式和文字体例经过重新修订，更易于课堂教学和自学指导。
此外，设置了一个全新的网站，为软件工程领域的学生、教师以及专业人士提供关于软件工程资源的全面服务。
《软件工程》分五大部分。
第一部分引入软件产品、过程等基本概念；第二部分介绍软件项目管理，包括管理概念、过程与项目度量、项目计划、风险管理、项目进度与跟踪、质量保证、配置管理；第三部分介绍传统的软件工程方法，包括系统工程、需求分析、分析建模、设计概念、体系结构设计、用户界面设计、构件层设计、测试技术、测试策略、技术度量等；第四部分介绍面向对象软件工程，包括概念、分析、设计、测试、技术度量等；第五部分是高级话题，包括形式化方法、净室软件工程，基于构件的开发、客户机/服务器软件工程、Web工程和CASE。

作者简介

Roger S. Pressman is an internationally recognized authority in software processimprovement and software engineering technologies. For over three decades, he hasworked as a software en~~neer. a manager, a professor, an author, and a consultant, focus.As an industry practitioner and manager, Dr. Pressman worked on the development ofCAD／CAM systems for advanced engineering and manufacturing applications. He has alsoheld positions with responsibility for scientific and systems programming. After receiving a Ph. D. in engineering from the University ofConnecticut, Dr. Pressmanmoved to academia where he became Bullard Associate Professor OfComputer Engineeringat the University of Bridgeport and director of the university。
ComputerAided Design andManufacturing Center. Dr. Pressman is currently president of R. S. Pressman＆Associates, lnca consultingfirm specializing in software engineering methods and training. He serves as principle consultant, helping companies establish effective software engineering practices. He also designed and evelopedthecompanyssoRwareengineeringtraining and process improve-mentproducts——EssentialSoftwareEngineenng, acompletevideo curriculumthatisamongthe industry。
s most comprehensive treatments of the subject, and Process Advisor, a self-directed system for software engineering process improvement. Both produc~~are used by hundreds of companies worldwide. Dr. Pressman has written many technical papers, is a regular contributor to industryperiodicals, andisauthorofsixbooks. InadditiontoSoftwareEngineenng：APractitioner Approach, he has written A Manager‘s Guide to Software Engineenng（McGraw-Hill）, an award-winning book that uses a unique Q＆A format to present management guidelines for instituting and understanding software engineering technology；Making Software Engi-neenngHappen（Prentice-Hall）, the first book to address the critical management problems associated with software process improvement；and Software Shock（Dorset House）, a treat-ment that focuses on software and its impact on business and society. Dr. Pressman is onthe Editorial Boards ofIEEESoftware and the CutterITJournal, and for many years, was editorofthe“ ManageV ’columninIEEESoftware. Dr. Pressman is a well-known speaker. keynoting a number of major industry confer- ences. He has presented tutorials at the International Conference on Software Engineer-ing and at many other industry meetings. He is a member of the ACM. IEEE, and Tau Beta Pi, Phi Kappa Phi, Eta Kappa Nu, and Pi Tau Sigma.

<<软件工程>>

书籍目录

章节摘录

15. 1.2 ReducetheUsersMemory Load The more a user has to remember, the more error-prone will be the interaction wil the system. It is for this reason that a well-designed user interface does not tax tl user, s memory. Whenever possible, the system should " remember pertinent info mation and assist the user with an interaction scenario that assists recall. Mand IMAN97defines design principles that enable an interface to reduce the user ' s mer oryload：Reduce demand on short-term memory. When users are lnvolved in corn-plex tasks , the demand on short-term memory can be significant. The interface should be designed to reduce the requirement to remember past actions and results. This can be accomplished by providing visual cues that enable a user to recognize past actions, rather than having to recall hem. Establish meaIlinI口叫defaults. The initial set of defaults should make sense for the average user, but a user should be able to specify individual preferences . However. a " reset " option should be available, enabling the redefinition of orig-inal default values. Define shortcuts that are intuitive. Whan mnemonics are used to ccomplish a system function(e. g. , alt-P to invoke the print function), the mnemonic should be tied to the action in a way that is easy to remember(e. g. , first letter ofthe task to be invoked). The visual layout of the interface should be based on a real world metaphor. For example, a bill payment system should use a check book and check register metaphor to guide the user through the bill paying process. This enables the user to rely on well-understood visual cues. rather than memoriz sequence. Discloseinfornmtioninapro~~essivefashion. Theinterfaceshouldbeorga-nized hierarchically . That is, information about a task, an object. or some behav ' ior should be presented first at a high level of abstraction. More detail should be presented after the user indicates interest with a mouse pick. An example , corn-mon to many word-processing applica.

版权说明