The Mythical Man Month Essays On Software Engineering Anniversary Edition

The Mythical Man-Month The Mythical Man-month The Thoughtworks Anthology Programming on Purpose III How To Be a Geek The Mythical Man-month (summary) Software Conflict 2.0 The ThoughtWorks Anthology, Volume 2 Programming on Purpose II The Mythical Man-month Essays on Object-oriented Software Engineering Free Software, Free Society Scrumban - Essays on Kanban Systems for Lean Software Development From Software Engineering to Formal Methods and Tools, and Back The Design of Design Essays In Personalizable Software Perspectives on the Future of Software Engineering The Software Development Edge Principled Software Development Lord of the Files Software, Services, and Systems Rigorous Methods for Software Construction and Analysis Formal Methods in Software and Systems Modeling Mathematical Foundations of Software Engineering. Essays in Honour of Tom Maibaum on the Occasion of His 70th Birthday and Retirement Understanding Software Specification, Algebra, and Software Amplifying Your Effectiveness Logic, Computation and Rigorous Methods Coding Regulation Structured Writing II Computer Science in Perspective Writing Guide Software & Supplemental Exercises Digital Fabrications Dependable and Historic Computing Theories of Programming and Formal Methods Verification: Theory and Practice In an Outpost of the Global Economy The Innovation in Computing Companion Reliability, Maintainability, and Supportability Algorithmics

Eventually, you will completely discover a further experience and completion by spending more cash. yet when? pull off you acknowledge that you require to acquire those all needs like having significantly cash? Why dont you attempt to get something basic in the beginning? Thats something that will guide you to understand even more roughly the globe, experience, some places, like history, amusement, and a lot more?

It is your categorically own period to pretense reviewing habit. among guides you could enjoy now is The Mythical Man Month Essays On Software Engineering Anniversary Edition below.

Rigorous Methods for Software Construction and Analysis Jan 13 2021 This Festschrift volume, published in honor of Egon Börger, contains 14 papers from a Dagstuhl Seminar, which was organized as a "Festkolloquium" on the occasion of his 60th birthday in May 2006.

Focusing on applied formal methods, the volume covers a wide range of applied research, spanning from theoretical and methodological foundations to practical applications of Abstract State Machines, B, and beyond, emphasizing universal methods and tools that, regardless of their applicational orientation, are still committed to the ideal of mathematical rigor. In particular, the papers address the following central topics: methodological foundations of requirements specification and verification, characterization of specification languages and their logical foundations, advanced tool environments and systematic integration of tools, machine assisted validation and verification, distributed algorithms and concurrent protocols, novel applications in public safety, security and privacy, industrial case studies and experience reports, and the role of formal methods in computer science education.

In an Outpost of the Global Economy Sep 28 2019 This book examines the growth of India's information technology (IT) industry, the people who work in these industries, the nature of the work itself, and its wider social and cultural ramifications. It combines empirical research with theoretical insight to explore important questions about the trajectory of globalization in India.

The Software Development Edge May 17 2021 The new software management classic: in-the-trenches wisdom from legendary project leader Joe Marasco Over the course of a distinguished career, Joe Marasco earned a reputation as the go-to software project manager: the one to call when you were facing a brutally tough, make-or-break project. Marasco reflected on his experiences in a remarkable series of "Franklin's Kite" essays for The Rational Edge, Rational and IBM's online software development magazine. Now, Marasco collects and updates those essays, bringing his unique insights (and humor) to everything from modeling to scheduling, team dynamics to compensation. The result: a new classic that deserves a place alongside Frederick Brooks' The Mythical Man-Month in the library of every developer and software manager. If you want to ship products you're proud of... ship on time and on budget... deliver real customer value... you simply must read The Software Development Edge.

The Innovation in Computing Companion Aug 27 2019 This encyclopedic reference provides a concise and engaging overview of the groundbreaking inventions and conceptual innovations that have shaped the field of computing, and the technology that runs the modern world. Each alphabetically-ordered entry presents a brief account of a pivotal innovation and the great minds behind it, selected from a wide range of diverse topics. Topics and features: Describes the development of Babbage’s computing machines, Leibniz’s binary arithmetic, Boole’s symbolic logic, and Von Neumann architecture Reviews a range of historical analog and digital computers, significant mainframes and minicomputers, and pioneering home and *personal computers Discusses a selection of programming languages and operating systems, along with key concepts in software engineering and commercial computing Examines the invention of the transistor, the integrated circuit, and the microprocessor Relates the history of such developments in personal computing as the mouse, the GUI, Atari video games, and Microsoft Office Surveys innovations in communications, covering mobile phones, WiFi, the Internet and World Wide Web, e-commerce, smartphones, social media, and GPS Presents coverage of topics on artificial intelligence, the ATM, digital photography and digital music, robotics, and Wikipedia Contains self-test quizzes and a helpful glossary This enjoyable compendium will appeal to the general reader curious about the intellectual milestones that led to the digital age, as well as to the student of computer science seeking a primer on the history of their field. Dr. Gerard O'Regan is a CMMI software process improvement consultant with research interests including software quality and software process improvement, mathematical approaches to software quality, and the history of computing. He is the author of such Springer titles as World of Computing, Concise Guide to Formal Methods, Concise Guide to Software Engineering, and Guide to Discrete Mathematics.*

The Mythical Man-month (summary) May 29 2022

Structured Writing II May 05 2020 Structured Writing II takes up where volume one left off , showing how students can use the same techniques to build simple paragraph units into full, complex essays. With the help of concept maps, word processing tools, and colorcoded templates, students who were once intimidated by writing assignments can succeed. Structured Writing II makes the process incremental and manageable, and gives students a taste of writing success. FEATURES Systematic instructions for writing introductory and concluding paragraphs, expository and persuasive essays, and book reports Recursive, multisensory composition process perfect for students with special needs Templates for essay webs and outlines on CD-ROM Also available: Educator's Podcast Guide - ISBN 1564842312 English Language Arts Units for Grades 9-12 - ISBN 1564842401

Principled Software Development Apr 15 2021 This book presents a collection of research papers that address the challenge of how to develop software in a principled way that, in particular, enables reasoning. The individual papers approach this challenge from various perspectives including programming languages, program verification, and the systematic variation of software. Topics covered include programming abstractions for concurrent and distributed software, specification and verification techniques for imperative programs, and development techniques for software product lines. With this book the editors and authors wish to acknowledge – on the occasion of his 60th birthday – the work of Arnd Poetzsch-Heffter, who has made major contributions to software technology throughout his career. It *features articles on Arnd’s broad research interests including, among others, the implementation of programming languages, formal semantics, specification and verification of object-oriented and concurrent programs, programming language design, distributed systems, software modeling, and software product lines. All contributing authors are leading experts in programming languages and software engineering who have collaborated with Arnd in the course of his career. Overall, the book offers a collection of high-quality articles, presenting original research results, major case studies, and inspiring visions. Some of the work included here was presented at a symposium in honor of Arnd Poetzsch-Heffter, held in Kaiserslautern, Germany, in November 2018.*

From Software Engineering to Formal Methods and Tools, and Back Sep 20 2021 This volume was published in honor of Stefania Gnesi’s 65th birthday. The Festschrift volume contains 32 papers written by close collaborators and friends of Stefania and was presented to her on October 8, 2019 one-day colloquium held in Porto, Portugal, The Festschrift consists of eight sections, seven of which reflect the main research areas to which Stefania has contributed. Following a survey of Stefania's legacy in research and a homage by her thesis supervisor, these seven sections are ordered according to Stefania's life cycle in research, from software engineering to formal methods and tools, and back: Software Engineering; Formal Methods and Tools; Requirements Engineering; Natural Language Processing; Software Product Lines; Formal Verification; and Applications.

Perspectives on the Future of Software Engineering Jun 17 2021 The dependence on quality software in all areas of life is what makes software engineering a key discipline for today’s society. Thus, over the last few decades it has been increasingly recognized that it is particularly important to demonstrate the value of software engineering methods in real-world environments, a task which is the focus of empirical software engineering. One of the leading protagonists of this discipline worldwide is Prof. Dr. Dr. h.c. Dieter Rombach, who dedicated his entire career to empirical software engineering. For his many important contributions to the field he has received numerous awards and recognitions, including the U.S. National Science Foundation’s Presidential Young Investigator Award and the Cross of the Order of Merit of the Federal Republic of Germany. He is a Fellow of both the ACM and the IEEE Computer Society. This book, published in honor of his 60th birthday, is dedicated to Dieter Rombach and his contributions to software engineering in general, as well as to empirical software engineering in particular. This book presents invited contributions from a number of the most internationally renowned software engineering researchers like Victor Basili, Barry Boehm, Manfred Broy, Carlo Ghezzi, Michael Jackson, Leon Osterweil, and, of course, by Dieter Rombach himself.

Several key experts from the Fraunhofer IESE, the institute founded and led by Dieter Rombach, also contributed to the book. The contributions summarize some of the most important trends in software engineering today and outline a vision for the future of the field. The book is structured into three main parts. The first part focuses on the classical foundations of software engineering, such as notations, architecture, and processes, while the second addresses empirical software engineering in particular as the core field of Dieter Rombach’s contributions. Finally, the third part discusses a broad vision for the future of software engineering.

Writing Guide Software & Supplemental Exercises Mar 03 2020

Specification, Algebra, and Software Sep 08 2020 This Festschrift volume, published in honor of Kokichi Futatsugi, contains 31 invited contributions from internationally leading researchers in formal methods and software engineering. Prof. Futatsugi is one of the founding fathers of the field of algebraic specification and verification and is a leading researcher in formal methods and software engineering. He has pioneered and advanced novel algebraic methods and languages supporting them such as OBJ and CafeOBJ and has worked tirelessly over the years to bring such methods and tools in contact with software engineering practice. This volume contains contributions from internationally leading researchers in formal methods and software engineering.

Understanding Software Oct 10 2020 Software legend Max Kanat-Alexander shows you how to succeed as a developer by embracing simplicity, with forty-three essays that will help you really understand the software you work with.About This Book Read and enjoy the superlative writing and insights of the legendary Max Kanat-Alexander* Learn and reflect with Max on how to bring simplicity to your software design principles* Discover the secrets of rockstar programmers and how to also just suck less as a programmerWho This Book Is ForUnderstanding Software is for every programmer, or anyone who works with programmers. If life is feeling more complex than it should be, and you need to touch base with some clear thinking again, this book is for you. If you need some inspiration and a reminder of how to approach your work as a programmer by embracing some simplicity in your work again, this book is for you.If you're one of Max's followers already, this book is a collection of Max's thoughts selected and curated for you to enjoy and reflect on. If you're new to Max's work, and ready to connect with the power of simplicity again, this book is for you!What You Will Learn* See how to bring simplicity and success to your programming world* Clues to complexity - and how to build excellent software* Simplicity and software design* Principles for programmers* The secrets of rockstar programmers* Max's views and interpretation of the Software industry* Why Programmers suck and how to suck less as a programmer* Software* *design in two sentences* What is a bug? Go deep into debuggingIn DetailIn Understanding Software, Max Kanat-Alexander, Technical Lead for Code Health at Google, shows you how to bring simplicity back to computer programming. Max explains to you why programmers suck, and how to suck less as a programmer. There's just too much complex stuff in the world. Complex stuff can't be used, and it breaks too easily. Complexity is stupid. Simplicity is smart.Understanding Software covers many areas of programming, from how to write simple code to profound insights into programming, and then how to suck less at what you do! You'll discover the problems with software complexity, the root of its causes, and how to use simplicity to create great software. You'll examine debugging like you've never done before, and how to get a handle on being happy while working in teams.Max brings a selection of carefully crafted essays, thoughts, and advice about working and succeeding in the software industry, from his legendary blog Code Simplicity. Max has crafted forty-three essays which have the power to help you avoid complexity and embrace simplicity, so you can be a happier and more successful developer.Max's technical knowledge, insight, and kindness, has earned him code guru status, and his ideas will inspire you and help refresh your approach to the challenges of being a developer.Style and approachUnderstanding Software is a new selection of carefully chosen and crafted essays from Max Kanat-Alexander's legendary blog call Code Simplicity. Max's writing and thoughts are great to sit and read cover to cover, or if you prefer you can drop in and see what you discover new every single time!*

The Mythical Man-month Jan 25 2022 On software project management

Software Conflict 2.0 Apr 27 2022 The nearly 60 essays in this book--always easily digestible, often profound, and never too serious--take up large themes and important questions, never shying away from controversy. (Computer Books)

Reliability, Maintainability, and Supportability Jul 27 2019 Focuses on the core systems engineering tasks of writing,managing, and tracking requirements for reliability,maintainability, and supportability that are most likely to satisfycustomers and lead to success for suppliers This book helps systems engineers lead the development ofsystems and services whose reliability, maintainability, andsupportability meet and exceed the expectations of their customersand promote success and profit for their suppliers. This book isorganized into three major parts: reliability, maintainability, andsupportability engineering. Within each part, there is material onrequirements development, quantitative modelling, statisticalanalysis, and best practices in each of these areas. Heavy emphasisis placed on correct use of language. The author discusses the useof various sustainability engineering methods and techniques incrafting requirements that are focused on the customers'needs, *unambiguous, easily understood by the requirements'stakeholders, and verifiable. Part of each major division of thebook is devoted to statistical analyses needed to determine whenrequirements are being met by systems operating in customerenvironments. To further support systems engineers in writing,analyzing, and interpreting sustainability requirements, this bookalso Contains "Language Tips" to help systems engineerslearn the different languages spoken by specialists andnon-specialists in the sustainability disciplines Provides exercises in each chapter, allowing the reader to tryout some of the ideas and procedures presented in the chapter Delivers end-of-chapter summaries of the current reliability,maintainability, and supportability engineering best practices forsystems engineers Reliability, Maintainability, and Supportability is a referencefor systems engineers and graduate students hoping to learn how toeffectively determine and develop appropriate requirements so thatdesigners may fulfil the intent of the customer.*

Algorithmics Jun 25 2019 The best selling 'Algorithmics' presents the most important, concepts, methods and results that are fundamental to the science of computing. It starts by introducing the basic ideas of algorithms, including their structures and methods of data manipulation. It then goes on to demonstrate how to design accurate and efficient algorithms, and discusses their inherent limitations. As the author himself says in the preface to the book; 'This book attempts to present a readable account of some of the most important and basic topics of computer science, stressing the fundamental and robust nature of the science in a form that is virtually independent of the details of specific computers, languages and formalisms'.

Dependable and Historic Computing Jan 01 2020 This Festschrift volume, published in honor of Brian Randell on the occasion of his 75th birthday, contains a total of 37 refereed contributions. Two biographical papers are followed by the six invited papers that were presented at the conference 'Dependable and Historic Computing: The Randell Tales', held during April 7-8, 2011 at Newcastle University, UK. The remaining contributions are authored by former scientific colleagues of Brian Randell. The papers focus on the core of Brian Randell's work: the development of computing science and the study of its history. Moreover, his wider interests are reflected and so the collection comprises papers on software engineering, storage fragmentation, computer architecture, programming languages and dependability. There is even a paper that echoes Randell's love of maps. After an early career with English Electric and then with IBM in New York and California, Brian Randell joined Newcastle University. His main research has been on dependable computing in all its forms, especially reliability, safety and security aspects, and he has led several major European collaborative projects.

The Mythical Man-month Oct 02 2022 The orderly Sweet-Williams are dismayed at their son's fondness for the messy pastime of gardening.

Programming on Purpose II Feb 23 2022 A collection of essays drawn from Plauger's popular monthly column, "Programming on Purpose", in Computer Language magazine. Focusing throughout on people-related matters, Plauger shows software writers how to be really ingenious; how to protect the fruits of their ingenuity; how to mix technology and politics; and how NOT to write shelfware.

Lord of the Files Mar 15 2021 Software engineering is a social activity; forget that and your career is lost... Starting with the premise that a good software engineer is necessarily both a good programmer and a good person, this unique new book on the culture of programmers emphasizes the importance of empathy, introspection, and the acceptance of oneself and others on the journey to quality software. Based on the author's extensive experience teaching software engineering, working as a computer programmer, and leading a social game startup from inception to acquisition, Lord of the Files is sensitive to the frailties of the human condition and full of innovative survival and success strategies for students, programmers, managers, and entrepreneurs. Contents: I, Programmer The Software Engineer Life Cycle Your Favourite Methodology is eXtremely Gay White Trash Software Engineer What the Bleep Should We Know ? Nobody Ever Got Laid For Buying IBM Equipment All We Really Need To Know about Software Engineering Is in the Film Office Space A Seven-Layer Hierarchy of Careers in Computer Science What's Your Secret Sauce? Pandemonium Reigned

Software, Services, and Systems Feb 11 2021 This book is dedicated to Professor Martin Wirsing on the occasion of his emeritation from Ludwig-Maximilians-Universität in Munich, Germany. The volume is a reflection, with gratitude and admiration, on Professor Wirsing’s life highly creative, remarkably fruitful and intellectually generous life. It also gives a snapshot of the research ideas that in many cases have been deeply influenced by Professor Wirsing’s work. The book consists of six sections. The first section contains personal remembrances and expressions of gratitude from friends of Professor Wirsing. The remaining five sections consist of groups of scientific papers written by colleagues and collaborators of Professor Wirsing, which have been grouped and ordered according to his scientific evolution. More specifically, the papers are concerned with logical and algebraic foundations; algebraic specifications, institutions and rewriting; foundations of software engineering; service oriented systems; and adaptive and autonomic systems.

Computer Science in Perspective Apr 03 2020 By presenting state-of-the-art aspects of theoretical computer science and practical applications in various fields, this book commemorates the 60th birthday of Thomas Ottmann. The 26 research papers presented span the *whole range of Thomas Ottmann's scientific career, from formal languages to algorithms and data structures, from topics in practical computer science like software engineering or database systems to applications of Web technology, groupware, and e-learning.*

Scrumban - Essays on Kanban Systems for Lean Software Development Oct 22 2021 Corey Ladas' groundbreaking paper "ScrumBan" has captured the imagination of the software development world. Scrum and agile methodologies have helped software development teams organize and become more efficient. Lean methods like kanban can extend these benefits. Kanban also provides a powerful mechanism to identify process improvement opportunities. This book covers some of the metrics and day-to-day management techniques that make continuous improvement an achievable outcome in the real world. ScrumBan the book provides a series of essays that give practitioners the background needed to create more robust practices combining the best of agile and lean.

Logic, Computation and Rigorous Methods Jul 07 2020 This Festschrift was published in honor of Egon Börger on the occasion of his 75th birthday. It acknowledges Prof. Börger's inspiration as a scientist, author, mentor, and community organizer. Dedicated to a pioneer in the fields of logic and computer science, Egon Börger's research interests are unusual in scope, from programming languages to hardware architectures, software architectures, control systems, workflow and interaction patterns, business processes, web applications, and concurrent systems. The 18 invited contributions in this volume are by leading researchers in the areas of software engineering, programming languages, business information systems, and computer science logic.

Digital Fabrications Jan 31 2020 Digital Fabrications is a collection of essays and half-true stories about design software and hardware. Written from the perspective of architectural design, each piece expands on emerging trends, devices, foibles, and phenomena engendered by an increased reliance on interactions with interfaces in the discipline. The essays ask, how do we characterize our post-digital design labor? What are the politics of design software? How is architecture adapting to a world largely dependent on platforms and scripts? What are the spatial mechanisms of the internet and VR? Using storytelling techniques, this book accepts that software is everywhere, and narrows in on a few ways it has taken command of our cultural products. From the perspective of architectural design, a field traditionally associated with sketching and its own myths of creativity, computers are an essential workplace tool. Projects rely on a wide assortment of software packages and standalone applications, but rarely do architects reflect on the structure of those programs or how they have infiltrated our disciplinary conventions. PDFs and JPGs are as much a part of our vocabulary as *plans, sections, and elevations. A drawing today might refer to a rendering, a CAD document, a proprietary BIM file, or anything that describes a project visually. While one way of examining this disciplinary shift might be to re-imagine what digital drawing can be, this collection of essays puts forth another way: to look at the behaviors, phenomena, collective trends, and oddities emerging as a result of global software proliferation. In other words, this book accepts that software is everywhere, and narrows in on a few ways it has taken command of our cultural products.*

Formal Methods in Software and Systems Modeling Dec 12 2020 By presenting state-of-the-art research results on various aspects of formal and visual modeling of software and systems, this book commemorates the 60th birthday of Hartmut Ehrig. The 24 invited reviewed papers are written by students and collaborators of Hartmut Ehrig who are established researchers in their fields. Reflecting the scientific interest and work of Hartmut Ehrig, the papers fall into three main parts on graph transformation, algebraic specification and logic, and formal and visual modeling.

The ThoughtWorks Anthology, Volume 2 Mar 27 2022 When you hit a rough spot in software development, it's nice to know that someone has been there before. The domain experts at ThoughtWorks share what they've learned in this anthology, bringing together the best field-tested insights in IT and software development. You'll benefit from their experience in areas from testing to information visualization, from object oriented to functional programming, from incremental development to driving innovation in delivery. You'll find yourself referring to this collection of solved problems whenever you need an expert's insight. This new collection of essays from the experts at ThoughtWorks offers practical insight and advice on a range of challenges faced daily by software developers and IT professionals. It covers a broad spectrum of software development topics, from tuning agile methodologies to hard-core language geekery. This anthology captures the wide-ranging intellect and diversity of ThoughtWorks, reflected through practical and timely topics. In it, you'll find from-the-trenches advice on topics such as continuous integration, testing, and improving the software delivery process. See how people use functional programming techniques in object-oriented languages, modern Java web applications, and deal with current problems in JavaScript development. Scan an overview of the most interesting programming languages today and the current state of information visualization. And it's all field-tested insight, because it comes from the practical perspective of ThoughtWorks experts. Each essay focuses on extending your skills and enlarging your toolkit. And each is drawn from practical experience gained in the field. You'll benefit from this book if you are involved in developing, deploying, or testing software, either as a manager or developer.

Verification: Theory and Practice Oct 29 2019 This festschrift volume constitutes a unique tribute to Zohar Manna on the occasion of his 64th birthday. Like the scientific work of Zohar Manna, the 32 research articles span the entire scope of the logical half of computer science. Also included is a paean to Zohar Manna by the volume editor. The articles presented are devoted to the theory of computing, program semantics, logics of programs, temporal logic, automated deduction, decision procedures, model checking, concurrent systems, reactive systems, hardware and software verification, testing, software engineering, requirements specification, and program synthesis.

Free Software, Free Society Nov 22 2021 Essay Collection covering the point where software, law and social justice meet.

The Mythical Man-Month Nov 03 2022 Few books on software project management have been as influential and timeless as The Mythical Man-Month. With a blend of software engineering facts and thought-provoking opinions, Fred Brooks offers insight for anyone managing complex projects. These essays draw from his experience as project manager for the IBM System/360 computer family and then for OS/360, its massive software system. Now, 20 years after the initial publication of his book, Brooks has revisited his original ideas and added new thoughts and advice, both for readers already familiar with his work and for readers discovering it for the first time. The added chapters contain (1) a crisp condensation of all the propositions asserted in the original book, including Brooks' central argument in The Mythical Man-Month: that large programming projects suffer management problems different from small ones due to the division of labor; that the conceptual integrity of the product is therefore critical; and that it is difficult but possible to achieve this unity; (2) Brooks' view of these propositions a generation later; (3) a reprint of his classic 1986 paper "No Silver Bullet"; and (4) today's thoughts on the 1986 assertion, "There will be no silver bullet within ten years."

The Thoughtworks Anthology Sep 01 2022 ThoughtWorks is a well-known global consulting firm; ThoughtWorkers are leaders in areas of design, architecture, SOA, testing, and agile methodologies. This collection of essays brings together contributions from well-known ThoughtWorkers such as Martin Fowler, along with other authors you may not know yet. While ThoughtWorks is perhaps best known for their work in the Agile community, this anthology confronts issues throughout the software development life cycle. From technology issues that transcend methodology, to issues of realizing business value from applications, you'll find it here.

Essays on Object-oriented Software Engineering Dec 24 2021 Examines object-oriented methods, practices, terminology, and concepts

Coding Regulation Jun 05 2020 The collected essays in this book *concern the intriguing matter of the interaction between law and technology and the normative role of information technology. More precisely, they focus on the way information and communication technologies regulate human behaviour. Can information technology be an alternative to legal regulation and, if so, what are the risks? In this book the question is addressed as to how law interacts with technology in general and whether software (‘code’) can be understood as law. Comparisons are made between different technologies of communication (fixed and mobile telephony and the Internet) from the angle of control of communications. A problem is that the tool becomes more and more intransparent: it becomes a means of controlling human behaviour that replaces instruments of law. The impact of this development in the domains of freedom of expression, privacy and property is therefore analysed in depth. The issues raised in this book were discussed during a conference entitled Code as Code, held in Amsterdam, The Netherlands. The report of the debate between leading experts who attended the Conference forms the round-up in the book, as do the proposals for a future agenda for research. Together and individually, the authors provide substantial and meaningful contributions to the lively debate on the relationship between ‘software code’ and ‘legal code’ as it was initiated by Lawrence Lessig’s book Code and other Laws of Cyberspace (1999). The book is therefore highly recommended to professionals in IT and law, legal scholars, government policy makers and – not to forget – students of IT and law. Coding Regulation is edited by Egbert J. Dommering, Professor of Information Law at the University of Amsterdam, and Lodewijk F. Asscher, researcher at the Institute for Information Law, University of Amsterdam, and Deputy Mayor for Economic Affairs of the City of Amsterdam. This is Volume 12 in the Information Technology and Law (IT&Law) Series*

Essays In Personalizable Software Jul 19 2021 The idea of personalizable software is fashionable today. I explored it in a number of software prototypes a decade or two earlier. The perspectives mechanism in Hermes, my dissertation software system, was an initial major initiative in this direction. WebNet was a follow-up system to integrate the perspective mechanism into discussion-forum collaboration software. Subsequent systems explored personalization mechanisms in systems for work and for learning, including TCA for teachers developing and sharing curriculum and systems for automated critics in design systems or reviewers of journal articles. In each case, the mechanisms were intended to support users to view and discuss materials from their personal perspectives and to share those views with others to encourage building group perspectives. The volume is organized in terms of essays on (a) structured hypermedia, (b) personalizable software, (c) software perspectives and (d) applications to health care, education *and publishing.*

Theories of Programming and Formal Methods Nov 30 2019 This Festschrift volume, dedicated to He Jifeng on the occasion of his 70th birthday in September 2013, includes 24 refereed papers by leading researchers, current and former colleagues, who congratulated at a celebratory symposium held in Shanghai, China, in the course of the 10th International Colloquium on Theoretical Aspects of Computing, ICTAC 2013. The papers cover a broad spectrum of subjects, from foundational and theoretical topics to programs and systems issues and to applications, comprising formal methods, software and systems modeling, semantics, laws of programming, specification and verification, as well as logics. He Jifeng is known for his seminal work in the theories of programming and formal methods for software engineering. He is particularly associated with Unifying Theories of Programming (UTP) , the theory of data refinement and the laws of programming, and the rCOS formal method for object and component system construction. His book on UTP with Tony Hoare has been widely read and followed by a large number of researchers, and it has been used in many postgraduate courses. He was a senior researcher at Oxford during 1984-1998, and then a senior research fellow at the United Nations University International Institute for Software Technology (UNU-IIST) in Macau during 1998-2005. He has been a professor and currently the Dean of the Institute of Software Engineering at East China Normal University, Shanghai, China. In 2005, He Jifeng was elected as an academician to the Chinese Academy of Sciences. He also received an honorary doctorate from the University of York. He won a number of prestigious science and technology awards, including a 2nd prize of Natural Science Award from the State Council of China, a 1st prize of Natural Science Award from the Ministry of Education of China, a 1st prize of Technology Innovation from the Ministry of Electronic Industry, and a number awards from Shanghai government.

The Design of Design Aug 20 2021 Making Sense of Design Effective design is at the heart of everything from software development to engineering to architecture. But what do we really know about the design process? What leads to effective, elegant designs? The Design of Design addresses these questions. These new essays by Fred Brooks contain extraordinary insights for designers in every discipline. Brooks pinpoints constants inherent in all design projects and uncovers processes and patterns likely to lead to excellence. Drawing on conversations with dozens of exceptional designers, as well as his own experiences in several design domains, Brooks observes that bold design decisions lead to better outcomes. The author tracks the evolution of the design process, treats collaborative and distributed design, and illuminates what makes a truly great designer. He examines the nuts and bolts of design processes, including budget *constraints of many kinds, aesthetics, design empiricism, and tools, and grounds this discussion in his own real-world examples—case studies ranging from home construction to IBM’s Operating System/360. Throughout, Brooks reveals keys to success that every designer, design project manager, and design researcher should know.*

Amplifying Your Effectiveness Aug 08 2020 Explore the People-Oriented Challenges That Software Engineers Must Master Gerald M. Weinberg, James Bach, Naomi Karten, and a group of successful software consultants present powerful ideas on how software engineers and managers can amplify their professional effectiveness--as individuals, as members of teams, and as members of organizations. The collected essays address diverse topics in personal empowerment, interpersonal interaction, mastering projects, and changing the organization. Contributors include James Bach, Marie Benesh, Rick Brenner, Esther Derby, Kevin Fjelsted, Don Gray, Naomi Karten, Bob King, Pat Medvick, Brian Pioreck, Ken Roberts, Sharon Marsh Roberts, Johanna Rothman, Steve Smith, Eileen Strider, Gerald M. Weinberg, and Becky Winant. The idea for this collection arose out of a brainstorming session for the inaugural Amplifying Your Effectiveness Conference (AYE), in 2000, for which the contributing authors served as hosts. Like the book, this annual conference is designed to help technical people become more effective individually, within a team, and within an organization. For details on the next AYE Conference, visit www.ayeconference.com. The variety of techniques and perspectives represented in the book will help you amplify your effectiveness--whether or not you attend the live event.

Mathematical Foundations of Software Engineering. Essays in Honour of Tom Maibaum on the Occasion of His 70th Birthday and Retirement Nov 10 2020 This Festschrift is dedicated to Tom Maibaum on the occasion of his 70th birthday and on the occasion of his retirement. The Festschrift is also a follow up to the International Symposium on the Mathematics of Software Engineering organized by the Department of Computing of the University of Rio Cuarto, in February 2019. The Festschrift contains contributions by Tom's former students, colleagues, and friends. These contributions address research questions and problems in the general area of Formal Methods for Software Engineering - an area which Tom has spent most of his academic life working on. There are many reasons for publishing a Festschrift. Tom's academic merits alone would suffice for publishing this one. This Festschrift is also a token of appreciation. We wanted to make use of the opportunity to thank Tom for the impact he has had in the academic careers of many of his colleagues and collaborators, and especially in those of his former students.

How To Be a Geek Jun 29 2022 Computer software and its structures, devices and processes are woven into our everyday life. Their significance is not just technical: the algorithms, programming *languages, abstractions and metadata that millions of people rely on every day have far-reaching implications for the way we understand the underlying dynamics of contemporary societies. In this innovative new book, software studies theorist Matthew Fuller examines how the introduction and expansion of computational systems into areas ranging from urban planning and state surveillance to games and voting systems are transforming our understanding of politics, culture and aesthetics in the twenty-first century. Combining historical insight and a deep understanding of the technology powering modern software systems with a powerful critical perspective, this book opens up new ways of understanding the fundamental infrastructures of contemporary life, economies, entertainment and warfare. In so doing Fuller shows that everyone must learn 'how to be a geek', as the seemingly opaque processes and structures of modern computer and software technology have a significance that no-one can afford to ignore. This powerful and engaging book will be of interest to everyone interested in a critical understanding of the political and cultural ramifications of digital media and computing in the modern world.*

Programming on Purpose III Jul 31 2022 This collection of essays drawn from Plauger's popular "Programming on Purpose" column in the magazine Computer Language, focuses on the technology of writing computer software. Plauger's style is clear without being simplistic, reducing complex themes to bite-size chunks. KEY TOPICS: Covers a number of important technical themes such as computer arithmetic, approximating math functions, human perception and artificial intelligence, encrypting data and clarifying documentation.