

Compiler Construction Principles Practice Solution Manual

ETAPS 2002 was the 7th instance of the European Joint Conferences on Theory and Practice of Software. ETAPS is an annual federated conference that was established in 1998 by combining a number of existing and new conferences. This year it comprised 5 conferences (FOSSACS, FASE, ESOP, CC, TACAS), 13 satellite workshops (ACL2, AGT, CMCS, COCV, DCC, INT, LDFA, SC, SFEDL, SLAP, SPIN, TPTS, and VISS), 8 invited lectures (not including those specific to the satellite events), and several tutorials. The events that comprise ETAPS address various aspects of the system development process, including specification, design, implementation, analysis, and improvement. The languages, methodologies, and tools which support these activities are all well within its scope. Different blends of theory and practice are represented, with an inclination towards theory with a practical motivation on one hand and soundly-based practice on the other. Many of the issues involved in software design apply to systems in general, including hardware systems, and the emphasis on software is not intended to be exclusive.

This guide was written for readers interested in learning the C++ programming language from scratch, and for both novice and advanced C++ programmers wishing to enhance their knowledge of C++. The text is organized to guide the reader from elementary language concepts to professional software development, with in depth coverage of all the C++ language elements en route.

This book constitutes the proceedings of the 23rd International Conference on Compiler Construction, CC 2014, which was held as part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, which took place in Grenoble, France, in April 2014. The 10 full papers and 4 tool papers included in this volume were carefully reviewed and selected from 47 submissions; the book also contains one invited talk. The papers are organized in topical sections named: program analysis and optimization; parallelism and parsing and new trends in compilation.

This book constitutes the refereed proceedings of the 15th International Conference on Compiler Construction, CC 2006, held in March 2006 as part of ETAPS. The 17 revised full papers presented together with three tool demonstration papers and one invited paper were carefully reviewed and selected from 71 submissions. The papers are organized in topical sections.

PPDP.

Compiler Optimizations for Scalable Parallel Systems

20th International Conference, CC 2011, Held as Part of the Joint European Conference on Theory and Practice of Software, ETAPS 2011, Saarbrücken, Germany, March 26--April 3, 2011, Proceedings

Building a Modern Computer from First Principles

23rd International Conference, CC 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings

A refreshing antidote to heavy theoretical tomes, this book is a concise, practical guide to modern compiler design and construction by an acknowledged master. Readers are taken step-by-step through each stage of compiler design, using the simple yet powerful method of

Download Ebook Compiler Construction Principles Practice Solution Manual

recursive descent to create a compiler for Oberon-0, a subset of the author's Oberon language. A disk provided with the book gives full listings of the Oberon-0 compiler and associated tools. The hands-on, pragmatic approach makes the book equally attractive for project-oriented courses in compiler design and for software engineers wishing to develop their skills in system software.

The CC program committee is pleased to present this volume with the proceedings of the 13th International Conference on Compiler Construction (CC 2004). CC continues to provide an exciting forum for researchers, educators, and practitioners to exchange ideas on the latest developments in compiler technology, programming language implementation, and language design. The conference emphasizes practical and experimental work and invites contributions on methods and tools for all aspects of compiler technology and all language paradigms. This volume serves as the permanent record of the 19 papers accepted for presentation at CC 2004 held in Barcelona, Spain, during April 1–2, 2004. The 19 papers in this volume were selected from 58 submissions. Each paper was assigned to three committee members for review. The program committee met for one day in December 2003 to discuss the papers and the reviews. By the end of the meeting, a consensus emerged to accept the 19 papers presented in this volume. However, there were many other quality submissions that could not be accommodated in the program; hopefully they will be published elsewhere.

The continued success of the CC conference series would not be possible without the help of the CC community. I would like to gratefully acknowledge and thank all of the authors who submitted papers and the many external reviewers who wrote reviews.

"Modern Compiler Design" makes the topic of compiler design more accessible by focusing on principles and techniques of wide application. By carefully distinguishing between the essential (material that has a high chance of being useful) and the incidental (material that will be of benefit only in exceptional cases) much useful information was packed in this comprehensive volume. The student who has finished this book can expect to understand the workings of and add to a language processor for each of the modern paradigms, and be able to read the literature on how to proceed. The first provides a firm basis, the second potential for growth. Thinking Low-Level, Writing High-Level, the second volume in the landmark Write Great Code series by Randall Hyde, covers high-level programming languages (such as Swift and Java) as well as code generation on 64-bit CPUs ARM, the Java Virtual Machine, and the Microsoft Common Runtime. Today's programming languages offer productivity and portability, but also make it easy to write sloppy code that isn't optimized for a compiler. Thinking Low-Level, Writing High-Level will teach you to craft source code that results in good machine code once it's run through a compiler. You'll learn:

- How to analyze the output of a compiler to verify that your code generates good machine code*
- The types of machine code statements that compilers generate for common control structures, so you can choose the best statements when writing HLL code*
- Enough assembly language to read compiler output*
- How compilers convert various constant and variable objects into machine data*

With an understanding of how compilers work, you'll be able to write source code that they can translate into elegant machine code. NEW TO THIS EDITION, COVERAGE OF:

- Programming languages like Swift and Java*
- Code generation on modern 64-bit CPUs*
- ARM processors on mobile phones and tablets*
- Stack-based architectures like the Java Virtual Machine*
- Modern language systems like the Microsoft Common Language Runtime*

*Managing IT in Construction/Managing Construction for Tomorrow
Engineering a Compiler*

A Complete Guide to Programming in C++

13th International Conference, CC 2004, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2004, Barcelona, Spain, March 29 - April 2, 2004, Proceedings

Thinking Low-Level, Writing High-Level

A guide to the concepts and applications of computer graphics covers such topics as interaction techniques, dialogue design, and user interface software.

Typical undergraduate CS/CE majors have a practical orientation: they study computing because they like programming and are good at it. This book has strong appeal to this core student group. There is more than enough material for a semester-long course. The challenge for a course in programming language concepts is to help practical students understand programming languages at an unaccustomed level of abstraction. To help meet this challenge, the book includes enough hands-on programming exercises and examples to motivate students whose primary interest in computing is practical

Constraints have emerged as the basis of a representational and computational paradigm that draws from many disciplines and can be brought to bear on many problem domains. This volume contains papers dealing with all aspects of computing with constraints. In particular, there are several papers on applications of constraints, reflecting the practical usefulness of constraint programming. The papers were presented at the 1998 International Conference on Principles and Practice of Constraint Programming (CP'98), held in Pisa, Italy, 26{30 - tober, 1998. It is the fourth in this series of conferences, following conferences in Cassis (France), Cambridge (USA), and Schloss Hagenberg (Austria). We received 115 high quality submissions. In addition, 7 abstracts submissions were not followed by a full paper, hence were not counted as submissions. The program committee selected 29 high quality papers after thorough refereeing by at least 3 experts and further discussion by committee members. We thank the referees and the program committee for the time and effort spent in reviewing the papers. The program committee invited three speakers: { Joxan Jarar { Peter Jeavons { Patrick Prosser Their papers are in this volume. The fundamental mathematical tools needed to understand machine learning include linear algebra, analytic geometry, matrix decompositions, vector calculus, optimization, probability and statistics. These topics are traditionally taught in disparate courses, making it hard for data science or computer science students, or professionals, to efficiently learn the mathematics. This self-contained textbook bridges the gap between mathematical and machine learning texts, introducing the mathematical concepts with a minimum of prerequisites. It uses these concepts to derive four central machine learning methods: linear regression, principal component analysis, Gaussian mixture models and support vector machines. For students and others with a mathematical background, these derivations provide a starting point to machine learning texts. For those learning the mathematics for the first time, the methods help build intuition and practical experience with applying mathematical concepts. Every chapter includes worked examples and exercises to test understanding. Programming tutorials are offered on the book's web site.

15th International Conference, CC 2006, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2006,

Download Ebook Compiler Construction Principles Practice Solution Manual

*Vienna, Austria, March 30–31, 2006, Proceedings
18th International Conference, CC 2009, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2009, York, UK, March 22–29, 2009, Proceedings
Principles and Practice Using C++
Write Great Code, Volume 2, 2nd Edition
Papers Presented at the Symposium, Albuquerque, New Mexico, January 19–22, 1992*

This book constitutes the refereed proceedings of the 14th International Conference on Compiler Construction, CC 2005, held in Edinburgh, UK in April 2005 as part of ETAPS. The 21 revised full papers presented together with the extended abstract of an invited paper were carefully reviewed and selected from 91 submissions. The papers are organized in topical sections on compilation, parallelism, memory management, program transformation, tool demonstrations, and pointer analysis.

Data Mining: Concepts and Techniques provides the concepts and techniques in processing gathered data or information, which will be used in various applications. Specifically, it explains data mining and the tools used in discovering knowledge from the collected data. This book is referred as the knowledge discovery from data (KDD). It focuses on the feasibility, usefulness, effectiveness, and scalability of techniques of large data sets. After describing data mining, this edition explains the methods of knowing, preprocessing, processing, and warehousing data. It then presents information about data warehouses, online analytical processing (OLAP), and data cube technology. Then, the methods involved in mining frequent patterns, associations, and correlations for large data sets are described. The book details the methods for data classification and introduces the concepts and methods for data clustering. The remaining chapters discuss the outlier detection and the trends, applications, and research frontiers in data mining. This book is intended for Computer Science students, application developers, business professionals, and researchers who seek information on data mining. Presents dozens of algorithms and implementation examples, all in pseudo-code and suitable for use in real-world, large-scale data mining projects Addresses advanced topics such as mining object-relational databases, spatial databases, multimedia databases, time-series databases, text databases, the World Wide Web, and applications in several fields Provides a comprehensive, practical look at the concepts and techniques you need to get the most out of your data

This book constitutes the proceedings of the 22nd International Conference on Compiler Construction, CC 2013, held as part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, which took place in Rome, Italy, in March 2013. The 13 papers presented in this book were carefully reviewed and selected from 53 submissions. They have been organized into five topical sections on register allocation, pointer analysis, data and information flow, machine learning, and refactoring.

Compiler Construction Principles and Practice Course Technology Ptr
22nd International Conference, CC 2013, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013, Proceedings

Principles of Compiler Design

19th International Conference, CC 2010, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2010, Paphos, Cyprus, March 20-28, 2010. Proceedings

16th International Workshop, LCPC 2003, College Station, TX, USA, October 2-4, 2003, Revised Papers

17th International Conference, CC 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29 - April 6, 2008. Proceedings

This book constitutes the refereed conference proceedings of the 21st International Conference on Principles and Practice of Constraint Programming, CP 2015, held in Cork, Ireland, in August/September 2015. This edition of the conference was part of George Boole 200, a celebration of the life and work of George Boole who was born in 1815 and worked at the

Download Ebook Compiler Construction Principles Practice Solution Manual

University College of Cork. It was also co-located with the 31st International Conference on Logic Programming (ICLP 2015). The 48 revised papers presented together with 3 invited talks and 16 abstract papers were carefully selected from numerous submissions. The scope of CP 2014 includes all aspects of computing with constraints, including theory, algorithms, environments, languages, models, systems, and applications such as decision making, resource allocation, scheduling, configuration, and planning.

This new, expanded textbook describes all phases of a modern compiler: lexical analysis, parsing, abstract syntax, semantic actions, intermediate representations, instruction selection via tree matching, dataflow analysis, graph-coloring register allocation, and runtime systems. It includes good coverage of current techniques in code generation and register allocation, as well as functional and object-oriented languages, that are missing from most books. In addition, more advanced chapters are now included so that it can be used as the basis for a two-semester or graduate course. The most accepted and successful techniques are described in a concise way, rather than as an exhaustive catalog of every possible variant. Detailed descriptions of the interfaces between modules of a compiler are illustrated with actual C header files. The first part of the book, *Fundamentals of Compilation*, is suitable for a one-semester first course in compiler design. The second part, *Advanced Topics*, which includes the advanced chapters, covers the compilation of object-oriented and functional languages, garbage collection, loop optimizations, SSA form, loop scheduling, and optimization for cache-memory hierarchies. This book constitutes the refereed proceedings of the 20th International Conference on Compiler Construction, CC 2011, held in Saarbrücken, Germany, March 26–April 3, 2011, as part of ETAPS 2011, the European Joint Conferences on Theory and Practice of Software. The 15 revised full papers presented together with the abstract of one invited talk were carefully reviewed and selected from 52 submissions. The papers are organized in topical sections on JIT compilation and code generation, program analysis, reversible computing and interpreters, parallelism and high-performance computing, and task and data distribution. This book constitutes the refereed proceedings of the 12th International Conference on Compiler Construction, CC 2003,

Download Ebook Compiler Construction Principles Practice Solution Manual

held in Warsaw, Poland, in April 2003. The 20 revised full regular papers and one tool demonstration paper presented together with two invited papers were carefully reviewed and selected from 83 submissions. The papers are organized in topical sections on register allocation, language constructs and their implementation, type analysis, Java, pot pourri, and optimization.

Modern Compiler Design

Data Mining: Concepts and Techniques

14th International Conference, CC 2005, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2005

Conference Record of the Nineteenth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages Proceedings of the ... International ACM SIGPLAN Conference

on Principles and Practice of Declarative Programming *Designed for an introductory course, this text encapsulates the topics essential for a freshman course on compilers. The book provides a balanced coverage of both theoretical and practical aspects. The text helps the readers understand the process of compilation and proceeds to explain the design and construction of compilers in detail. The concepts are supported by a good number of compelling examples and exercises.*

This entirely revised second edition of Engineering a Compiler is full of technical updates and new material covering the latest developments in compiler technology. In this comprehensive text you will learn important techniques for constructing a modern compiler. Leading educators and researchers Keith Cooper and Linda Torczon combine basic principles with pragmatic insights from their experience building state-of-the-art compilers. They will help you fully understand important techniques such as compilation of imperative and object-oriented languages, construction of static single assignment forms, instruction scheduling, and graph-coloring register allocation. In-depth treatment of algorithms and techniques used in the front end of a modern compiler Focus on code optimization and code generation, the primary areas of recent research and development Improvements in presentation including conceptual overviews for each chapter, summaries and review questions for sections, and prominent placement of definitions for new terms Examples drawn from several different programming languages

This compiler design and construction text introduces students to the concepts and issues of compiler design, and features a comprehensive, hands-on case study project for constructing an actual, working compiler Immersing students in Java and the Java Virtual Machine (JVM), Introduction to Compiler Construction in a Java World enables a deep understanding of the Java programming language and its implementation. The text focuses on design, organization, and testing, helping students learn good software

engineering skills and become better programmers. The book covers all of the standard compiler topics, including lexical analysis, parsing, abstract syntax trees, semantic analysis, code generation, and register allocation. The authors also demonstrate how JVM code can be translated to a register machine, specifically the MIPS architecture. In addition, they discuss recent strategies, such as just-in-time compiling and hotspot compiling, and present an overview of leading commercial compilers. Each chapter includes a mix of written exercises and programming projects. By working with and extending a real, functional compiler, students develop a hands-on appreciation of how compilers work, how to write compilers, and how the Java language behaves. They also get invaluable practice working with a non-trivial Java program of more than 30,000 lines of code. Fully documented Java code for the compiler is accessible at <http://www.cs.umb.edu/j-/>

4th International Conference, CP98, Pisa, Italy, October 26-30, 1998, Proceedings

Programming

Principles and Practice of Constraint Programming

Introduction to Compiler Construction in a Java World

Introduction to Compiler Construction with UNIX

This title gives students an integrated and rigorous picture of applied computer science, as it comes to play in the construction of a simple yet powerful computer system.

An Introduction to Programming by the Inventor of C++

Preparation for Programming in the Real World The book

assumes that you aim eventually to write non-trivial programs, whether for work in software development or in some other technical field. Focus on Fundamental Concepts and Techniques The book explains fundamental concepts and techniques in greater depth than traditional introductions. This approach will give you a solid foundation for writing useful, correct, maintainable, and efficient code.

Programming with Today's C++ (C++11 and C++14) The book is an introduction to programming in general, including object-oriented programming and generic programming. It is also a solid introduction to the C++ programming language, one of the most widely used languages for real-world software. The book presents modern C++ programming techniques from the start, introducing the C++ standard library and C++11 and C++14 features to simplify programming tasks. For

Beginners--And Anyone Who Wants to Learn Something New The book is primarily designed for people who have never programmed before, and it has been tested with many thousands of first-year university students. It has also been extensively used for self-study. Also, practitioners

Download Ebook Compiler Construction Principles Practice Solution Manual

and advanced students have gained new insight and guidance by seeing how a master approaches the elements of his art. Provides a Broad View The first half of the book covers a wide range of essential concepts, design and programming techniques, language features, and libraries. Those will enable you to write programs involving input, output, computation, and simple graphics. The second half explores more specialized topics (such as text processing, testing, and the C programming language) and provides abundant reference material. Source code and support supplements are available from the author's website.

ETAPS'99 is the second instance of the European Joint Conferences on Theory and Practice of Software. ETAPS is an annual federated conference that was established in 1998 by combining a number of existing and new conferences. This year it comprises five conferences (FOSSACS, FASE, ESOP, CC, TACAS), four satellite workshops (CMCS, AS, WAGA, CoFI), seven invited lectures, two invited tutorials, and six contributed tutorials. The events that comprise ETAPS address various aspects of the system - development process, including specification, design, implementation, analysis and improvement. The languages, methodologies and tools which support these - activities are all well within its scope. Different blends of theory and practice are represented, with an inclination towards theory with a practical motivation on one hand and soundly-based practice on the other. Many of the issues involved in software design apply to systems in general, including hardware systems, and the emphasis on software is not intended to be exclusive.

This book constitutes the proceedings of the 26th International Conference on Principles and Practice of Constraint Programming, CP 2020, held in Louvain-la-Neuve, Belgium, in September 2020. The conference was held virtually due to the COVID-19 pandemic. The 55 full papers presented in this volume were carefully reviewed and selected from 122 submissions. They deal with all aspects of computing with constraints including theory, algorithms, environments, languages, models, systems, and applications such as decision making, resource allocation, scheduling, configuration, and planning. The papers were organized according to the following topics/tracks: technical track; application track; and CP and data science and machine learning.

Download Ebook Compiler Construction Principles Practice Solution Manual

Modern Programming Languages

Compilers: Principles, Techniques and Tools (for Anna University), 2/e

15th International Conference, CP 2009 Lisbon, Portugal, September 20-24, 2009 Proceedings

Modern Compiler Implementation in C

Computer Graphics

This book constitutes the refereed proceedings of the 19th International Conference on Compiler Construction, CC 2010, held in Paphos, Cyprus, in March 2010, as part of ETAPS 2010, the Joint European Conferences on Theory and Practice of Software.

Following a thorough review process, 16 research papers were selected from 56 submissions. Topics covered include optimization techniques, program transformations, program analysis, register allocation, and high-performance systems. Language definition. Word recognition. Language recognition. Error recovery. Semantic restrictions. Memory allocation. Code generation. A load-and-go system. "sampleC compiler listing.

Scalable parallel systems or, more generally, distributed memory systems offer a challenging model of computing and pose fascinating problems regarding compiler optimization, ranging from language design to run time systems. Research in this area is foundational to many challenges from memory hierarchy optimizations to communication optimization. This unique, handbook-like monograph assesses the state of the art in the area in a systematic and comprehensive way. The 21 coherent chapters by leading researchers provide complete and competent coverage of all relevant aspects of compiler optimization for scalable parallel systems. The book is divided into five parts on languages, analysis, communication optimizations, code generation, and run time systems. This book will serve as a landmark source for education, information, and reference to students, practitioners, professionals, and researchers interested in updating their knowledge about or active in parallel computing.

Managing IT in Construction/Managing Construction for Tomorrow presents new developments in:- Managing IT strategies - Model based management tools including building information modeling- Information and knowledge management- Communication and collaboration - Data acquisition and storage- Visualization and simulation- Architectural design and

Languages, Compilation Techniques, and Run Time Systems

8th International Conference, CC'99, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS'99, Amsterdam, The Netherlands, March 22-28, 1999, Proceedings

21st International Conference, CP 2015, Cork, Ireland, August 31 -- September 4, 2015, Proceedings

Principles and Practice of Constraint Programming - CP98

This book constitutes the refereed proceedings of the 18th International Conference on Compiler Construction, CC 2009, held in York, UK, in March 2009 as part of ETAPS 2009, the European Joint Conferences on Theory and Practice of Software. Following a very thorough review process, 18 full research papers were selected from 72 submissions. Topics covered include traditional compiler construction, compiler analyses, runtime systems and tools, programming tools, techniques for specific domains,

and the design and implementation of novel language constructs. This book constitutes the thoroughly refereed post-proceedings of the 16th International Workshop on Languages and Compilers for Parallel Computing, LCPC 2003, held in College Station, Texas, USA, in October 2003. The 35 revised full papers presented were selected from 48 submissions during two rounds of reviewing and improvement upon presentation at the workshop. The papers are organized in topical sections on adaptive optimization, data locality, parallel languages, high-level transformations, embedded systems, distributed systems software, low-level transformations, compiling for novel architectures, and optimization infrastructure.

This volume contains the papers presented at CP 2009: The 15th International Conference on Principles and Practice of Constraint Programming. It was held from September 20–24, 2009 at the Rectory of the New University of Lisbon, Portugal. Everyone involved with the conference thanks our sponsors for their support. There were 128 submissions to the research track, of which 53 were accepted for a rate of 41.4%. Each submission was reviewed by three reviewers, with a small number of additional reviews obtained in exceptional cases. Each review was either by a Programme Committee member, or by a colleague invited to help by a committee member thanks to their particular expertise. Papers submitted as long papers were accepted at full length or not at all. It is important to note that papers submitted as short papers were held to the same high standards of quality as long papers. There is thus no distinction in these proceedings between long and short papers, except of course the number of pages they occupy. As it happens, the acceptance rates of short and long papers were very similar indeed.

There were 13 submissions to the application track, of which 8 were accepted, for a rate of 61.5%. Papers underwent the same review process as regular papers, and there was not a separate committee for reviewing application track papers. However, papers in the application track were not required to be original or novel research, but to be original and novel as an application of constraints.

Compilers and operating systems constitute the basic interfaces between a programmer and the machine for which he is developing software. In this book we are concerned with the construction of the former. Our intent is to provide the reader with a firm theoretical basis for compiler construction and sound engineering principles for selecting alternate methods, implementing them, and integrating them into a reliable, economically viable product. The emphasis is upon a clean decomposition employing modules that can be re-used for many compilers, separation of concerns to facilitate team programming, and flexibility to accommodate hardware and system constraints. A reader should be able to understand the questions he must

ask when designing a compiler for language X on machine Y, what tradeoffs are possible, and what performance might be obtained. He should not feel that any part of the design rests on whim; each decision must be based upon specific, identifiable characteristics of the source and target languages or upon design goals of the compiler. The vast majority of computer professionals will never write a compiler. Nevertheless, study of compiler technology provides important benefits for almost everyone in the field . • It focuses attention on the basic relationships between languages and machines. Understanding of these relationships eases the inevitable transitions to new hardware and programming languages and improves a person's ability to make appropriate tradeoffs in design and implementation .

Principles and Practice of Constraint Programming - CP 2009

Compiler Construction

The Elements of Computing Systems

Languages and Compilers for Parallel Computing

12th International Conference, CC 2003, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2003, Warsaw, Poland, April 7-11, 2003, Proceedings

This book constitutes the proceedings of the 17th International Conference on Compiler Construction, CC 2008. It covers analysis and transformations, compiling for parallel architectures, runtime techniques and tools, analyses, and atomicity and transactions.

Principles and Practice

26th International Conference, CP 2020, Louvain-la-Neuve, Belgium, September 7-11, 2020, Proceedings

Mathematics for Machine Learning

A Practical Introduction

11th International Conference, CC 2002, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2002, Grenoble, France, April 8-12, 2002, Proceedings