

<<计算复杂性>>

图书基本信息

书名：<<计算复杂性>>

13位ISBN编号：9787115224002

10位ISBN编号：7115224005

出版时间：2010-4

出版时间：人民邮电出版社

作者：Oded Goldreich

页数：603

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<计算复杂性>>

前言

The quest for efficiency is ancient and universal, as time and other resources are always in shortage. Thus, the question of which tasks can be performed efficiently is central to the human experience. A key step toward the systematic study of the aforementioned question is a rigorous definition of the notion of a task and of procedures for solving tasks. These definitions were provided by computability theory, which emerged in the 1930s. This theory focuses on computational tasks, and considers automated procedures (i.e., computing devices and algorithms) that may solve such tasks. In focusing attention on computational tasks and algorithms, computability theory has set the stage for the study of the computational resources (like time) that are required by such algorithms. When this study focuses on the resources that are necessary for any algorithm that solves a particular task (or a task of a particular type) , the study becomes part of the theory of Computational Complexity (also known as Complexity Theory) .¹ Complexity Theory is a central field of the theoretical foundations of computer science. It is concerned with the study of the intrinsic complexity of computational tasks. That is, a typical complexity theoretic study refers to the computational resources required to solve a computational task (or a class of such tasks) , rather than referring to a specific algorithm or an algorithmic schema. Actually, research in Complexity Theory tends to start with and focus on the computational resources themselves, and addresses the effect of limiting these resources on the class of tasks that can be solved. Thus, Computational Complexity is the general study of what can be achieved within limited time (and/or other limited natural computational resources) .

<<计算复杂性>>

内容概要

复杂性理论是计算机科学的理论基础的核心。

本书是著名计算机科学家Oded Goldreich的力作，书中对计算任务固有复杂性研究进行了概念性介绍，全面分析了复杂性理论的现代主题。

本书涉及复杂性理论的很多子领域（如难度放大、伪随机性及概率证明系统等），涵盖了NP完整性、空间复杂性、随机性和计数、伪随机数生成器等内容，还在附录里面介绍了现代密码学基础等。

本书内容严谨，可读性强，适合作为高年级本科生、研究生的教材。

同时，书中展示了复杂性理论的很多子领域，也适合领域专家参考。

<<计算复杂性>>

作者简介

Oded Goldreich 以色列魏茨曼科学研究所 (Weizmann Institute of Science) 计算机科学教授, Meyer W. Weisgal 讲席教授。

他是SIAM Journal on Computing、Journal of Cryptology和Computational Complexity杂志的特约编辑。

<<计算复杂性>>

书籍目录

1 Introduction and Preliminaries 1 1.1 Introduction 1 1.1.1 A Brief Overview of Complexity Theory 2 1.1.2 Characteristics of Complexity Theory 6 1.1.3 Contents of This Book 8 1.1.4 Approach and Style of This Book 12 1.1.5 Standard Notations and Other Conventions 16 1.2 Computational Tasks and Models 17 1.2.1 Representation 18 1.2.2 Computational Tasks 18 1.2.3 Uniform Models (Algorithms) 20 1.2.4 Non-uniform Models (Circuits and Advice) 36 1.2.5 Complexity Classes 42 Chapter Notes 432 P, NP, and NP-Completeness 44 2.1 The P Versus NP Question 46 2.1.1 The Search Version: Finding Versus Checking 47 2.1.2 The Decision Version: Proving Versus Verifying 50 2.1.3 Equivalence of the Two Formulations 54 2.1.4 Two Technical Comments Regarding NP 55 2.1.5 The Traditional Definition of NP 55 2.1.6 In Support of P Different from NP 57 2.1.7 Philosophical Meditations 58 2.2 Polynomial-Time Reductions 58 2.2.1 The General Notion of a Reduction 59 2.2.2 Reducing Optimization Problems to Search Problems 61 2.2.3 Self-Reducibility of Search Problems 63 2.2.4 Digest and General Perspective 67 2.3 NP-Completeness 67 2.3.1 Definitions 68 2.3.2 The Existence of NP-Complete Problems 69 2.3.3 Some Natural NP-Complete Problems 71 2.3.4 NP Sets That Are Neither in P nor NP-Complete 81 2.3.5 Reflections on Complete Problems 85 2.4 Three Relatively Advanced Topics 87 2.4.1 Promise Problems 87 2.4.2 Optimal Search Algorithms for NP 92 2.4.3 The Class coNP and Its Intersection with NP 94 Chapter Notes 97 Exercises 993 Variations on P and NP 108 3.1 Non-uniform Polynomial Time (P/poly) 108 3.1.1 Boolean Circuits 109 3.1.2 Machines That Take Advice 111 3.2 The Polynomial-Time Hierarchy (PH) 113 3.2.1 Alternation of Quantifiers 114 3.2.2 Non-deterministic Oracle Machines 117 3.2.3 The P/poly Versus NP Question and PH 119 Chapter Notes 121 Exercises 1224 More Resources, More Power 127 4.1 Non-uniform Complexity Hierarchies 128 4.2 Time Hierarchies and Gaps 129 4.2.1 Time Hierarchies 129 4.2.2 Time Gaps and Speedup 136 4.3 Space Hierarchies and Gaps 139 Chapter Notes 139 Exercises 1405 Space Complexity 143 5.1 General Preliminaries and Issues 144 5.1.1 Important Conventions 144 5.1.2 On the Minimal Amount of Useful Computation Space 145 5.1.3 Time Versus Space 146 5.1.4 Circuit Evaluation 153 5.2 Logarithmic Space 153 5.2.1 The Class L 154 5.2.2 Log-Space Reductions 154 5.2.3 Log-Space Uniformity and Stronger Notions 155 5.2.4 Undirected Connectivity 155 5.3 Non-deterministic Space Complexity 162 5.3.1 Two Models 162 5.3.2 NL and Directed Connectivity 164 5.3.3 A Retrospective Discussion 171 5.4 PSPACE and Games 172 Chapter Notes 175 Exercises 1756 Randomness and Counting 1847 The Bright Side of Hardness 2418 Pseudorandom Generators 2849 Probabilistic Proof Systems 34910 Relaxing the Requirements 416 Epilogue 461 Appendix A: Glossary of Complexity Classes 463 Appendix B: On the Quest for Lower Bounds 469 Appendix C: On the Foundations of Modern Cryptography 482 Appendix D: Probabilistic Preliminaries and Advanced Topics in Randomization 523 Appendix E: Explicit Constructions 545 Appendix F: Some Omitted Proofs 566 Appendix G: Some Computational Problems 583 Bibliography 589 Index 60

<<计算复杂性>>

章节摘录

插图：A key step toward the systematic study of the aforementioned question is a rigorous definition of the notion of a task and of procedures for solving tasks. These definitions were provided by computability theory, which emerged in the 1930s. This theory focuses on computational tasks, and considers automated procedures (i.e., computing devices and algorithms) that may solve such tasks. In focusing attention on computational tasks and algorithms, computability theory has set the stage for the study of the computational resources (like time) that are required by such algorithms. When this study focuses on the resources that are necessary for any algorithm that solves a particular task (or a task of a particular type) , the study becomes part of the theory of Computational Complexity (also known as Complexity Theory) .¹ Complexity Theory is a central field of the theoretical foundations of computer science. It is concerned with the study of the intrinsic complexity of computational tasks. That is, a typical complexity theoretic study refers to the computational resources required to solve a computational task (or a class of such tasks) , rather than referring to a specific algorithm or an algorithmic schema. Actually, research in Complexity Theory tends to start with and focus on the computational resources themselves, and addresses the effect of limiting these resources on the class of tasks that can be solved. Thus, Computational Complexity is the general study of what can be achieved within limited time (and/or other limited natural computational resources) .

<<计算复杂性>>

媒体关注与评论

“这是一本非常值得关注的书……Goldreich的观点让我耳目一新……本书特别注重概念性问题，是研究人员及专家不可或缺的参考文献。

”——M. Bona，佛罗里达大学

<<计算复杂性>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>