

<<自动机理论、语言和计算导论>>

图书基本信息

书名：<<自动机理论、语言和计算导论>>

13位ISBN编号：9787111240358

10位ISBN编号：7111240359

出版时间：2008-7

出版时间：机械工业出版社

作者：（美）霍普克罗夫特（Hopcroft, J.E） 等著；孙家X 等译

页数：366

译者：孙家骥

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>

<<自动机理论、语言和计算导论>>

前言

理论计算机科学是推动计算机技术向前发展的强大动力。

自动机、形式语言、可计算性和相关方面内容构成的计算理论，是理论计算机科学的基础内容之一。学习、研究这些内容，不仅为进一步学习、研究理论计算机科学所必需，而且对增强形式化能力和推理能力有重要作用，这些能力对从事计算机技术中的软件形式化等研究，是不可缺少的。

本书是由John E. Hopcroft、Rajeev Motwani和Jeffrey D. Ullman三位计算机学者合作编写的，是最著名的理论计算机科学著作之一，是世界各国广泛采用的计算机理论专业和计算机工程专业的优秀教材之一。

它主要介绍形式语言、自动机、可计算性和相关方面内容。

它特别注意定义、定理的准确性和严格性，这有利于培养学生形式化和严格的数学推理的能力。

本书第1版于1979年发行。

它的第3版有一个新的特色，那就是增加了一套由Gradiance公司开发的在线作业帮助系统。

教师可以利用它给学生安排课后作业，或者给那些没有选课的学生提供一个特殊的综合课程（不是由老师申请开设的课程），使得他们可以利用这些课后作业作为练习和指导。

然而遗憾的是，Gradiance在线作业系统只适用于购买原版教材的北美地区学生，中国学生还不能使用这个系统，因此，我们在翻译第3版的过程中，对涉及Gradiance系统的内容进行了删减。

引进国外的优秀计算机教材，无疑会对我国的计算机教育事业的发展起积极推动作用，也是与世界接轨、建立世界一流大学不可缺少的条件。

我们把本书介绍给国内从事计算机教育事业的同行们，以供参考。

这个译本是根据原书第3版翻译的。

参加本书翻译的有：孙家骥同志，负责各章译稿的详细修改和全书的统稿；刘明浩同志，负责第1~3章及前言、目录的翻译；孙自然同志，负责第4、5章的翻译；王啸吟同志，负责第6、7章的翻译；王华同志，负责第8、9章的翻译；侯姗姗同志，负责第10、11章及索引的翻译。

为了与本书第2版中译本用的名词、术语保持一致，在翻译过程中我们参考了由刘田、姜晖和王捍贫同志完成的本书第2版中译本。

由于我们的能力有限，难免有不当之处，敬请读者不吝赐教。

<<自动机理论、语言和计算导论>>

内容概要

本书是关于形式语言、自动机理论和计算复杂性方面的经典之作，是国际上得到广泛认可的计算机理论和计算机工程专业的优秀教材。

书中涵盖了有穷自动机、正则表达式与语言、正则语言的性质、上下文无关文法及上下文无关语言、下推自动机、上下文无关语言的性质、图灵机、不可判定性以及难解问题等内容。

本书注重定义、定理的准确性和严格性，注重学生形式化和严格的数学推理能力的培养，同时在定义和证明中运用直观的方法说明抽象概念，借助许多图表帮助传达思想，并包含大量难度各异的示例和习题，便于读者加深对内容的理解。

本书适合作为计算机专业高年级本科生及研究生计算理论课程的教材和教学参考书。

<<自动机理论、语言和计算导论>>

作者简介

Hopcroft, J.E, 地斯坦福大学获得博士学位, 现为康奈尔大任康奈尔大学工程学院院长。
他是1986年图灵奖获得者。
他的研究兴趣集中在计算理论方面, 尤其是算法分析、自动机理论等。

<<自动机理论、语言和计算导论>>

书籍目录

出版者的话 译者序 前言 第1章 自动机：方法与体验 1.1 为什么研究自动机理论 1.1.1 有穷自动机简介 1.1.2 结构表示法 1.1.3 自动机与复杂性 1.2 形式化证明简介 1.2.1 演绎证明 1.2.2 求助于定义 1.2.3 其他定理形式 1.2.4 表面上不是“如果-则”命题的定理 1.3 其他的证明形式 1.3.1 证明集合等价性 1.3.2 逆否命题 1.3.3 反证法 1.3.4 反例 1.4 归纳证明 1.4.1 整数上的归纳法 1.4.2 更一般形式的整数归纳法 1.4.3 结构归纳法 1.4.4 互归纳法 1.5 自动机理论的中心概念 1.5.1 字母表 1.5.2 串 1.5.3 语言 1.5.4 问题 1.6 小结 1.7 参考文献 第2章 有穷自动机 2.1 有穷自动机的非形式化描述 2.1.1 基本规则 2.1.2 协议 2.1.3 允许自动机忽略动作 2.1.4 整个系统成为一个自动机 2.1.5 用乘积自动机验证协议 2.2 确定型有穷自动机 2.2.1 确定型有穷自动机的定义 2.2.2 DFA如何处理串 2.2.3 DFA的简化记号 2.2.4 把转移函数扩展到串 2.2.5 DFA的语言 2.2.6 习题 2.3 非确定型有穷自动机 2.3.1 非确定型有穷自动机的非形式化观点 2.3.2 非确定型有穷自动机的定义 2.3.3 扩展转移函数 2.3.4 NFA的语言 2.3.5 确定型有穷自动机与非确定型有穷自动机的等价性 2.3.6 子集构造的坏情形 2.3.7 习题 2.4 应用：文本搜索 2.4.1 在文本中查找串 2.4.2 文本搜索的非确定型有穷自动机 2.4.3 识别关键字集合的DFA 2.4.4 习题 2.5 带 ϵ 转移的有穷自动机 2.5.1 ϵ 转移的用途 2.5.2 ϵ -NFA的形式化定义 2.5.3 ϵ 闭包 2.5.4 ϵ -NFA的扩展转移和语言 2.5.5 消除 ϵ 转移 2.5.6 习题 2.6 小结 2.7 参考文献 第3章 正则表达式与正则语言 第4章 正则语言的性质 第5章 上下文无关文法及上下文无关语言 第6章 下推自动机 第7章 上下文无关语言的性质 第8章 图灵机导引 第9章 不可判定性 第10章 难解问题 第11章 其他问题类索引

章节摘录

插图：第1章 自动机：方法与体验自动机理论研究抽象计算装置或“机器”。

在20世纪30年代计算机出现之前，图灵研究过一种抽象机器，这种机器具备了今天计算机的所有能力，至少在计算能力上是这样的。

图灵的目标是精确地描述一条界线，这条界线区分计算机能做什么和不能做什么；图灵的结论不仅适用于抽象的图灵机，也适用于今天的真实机器。

在20世纪40和50年代，许多研究者研究过更简单类型的机器，今天称为“有穷自动机”。

起初建议用这些自动机来为人脑功能建立模型，后来发现这些自动机对于1.1节提到的各种其他目的也极为有用。

在20世纪50年代后期，语言学家乔姆斯基（N.Chomsky）开始研究形式“文法”。

尽管这些文法不是严格意义上的机器，但与抽象自动机有密切关系，而且目前是一些重要软件部件（包括部分编译器在内）的基础。

在1969年，库克（s.Cook）扩展了图灵对什么能被计算和什么不能被计算的研究。

库克设法分离出了两类问题：一类是计算机能有效解决的；另一类是计算机理论上能解决，但实际上要花费太长时间，以致除了非常小的问题实例以外，计算机是毫无用处的。

后一类问题称为“难解的”或“NP-难的”。

计算机硬件一直遵循着计算速度呈指数增长的规律（摩尔定律），但这也不太可能显著地影响解决难解问题大实例的能力。

所有这些理论进展都直接影响了计算机科学家今天的工作。

有些概念，比如有穷自动机和某些类型的形式文法，用于设计和构造重要类型的软件。

另一些概念，比如图灵机，则帮助我们理解能期待软件做什么。

特别是，难解问题的理论允许推断是否有可能“正面”处理一个问题，编写解决这个问题的程序（因为这个问题不属于难解的一类），或者是否需要找到某种方法来迂回处理这个难解的问题：找近似算法、用启发式算法或者用某种其他方法来限制程序解决这个问题所花费的时间。

<<自动机理论、语言和计算导论>>

编辑推荐

《自动机理论、语言和计算导论》已被世界许多著名大学采用为计算机理论课程的教材或教学参考书，适合作为国内高校计算机专业高年级本科生或研究生的教材，还可供从事理论计算工作的研究人员参考。

以简洁和易理解的方式讲述理论概念。

强调理论的现代应用。

使用大量的图来帮助表达概念。

提供定义和证明的更多细节。

每章提供大量难易程度不同的练习。

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>