普通高等教育"十一五"国家级规划教材

重点大学计算机专业系列教材

计算机科学与技术概论

郭　平　朱郑州　王艳霞　编著

清华大学出版社

普通高等教育“十一五”国家级规划教材

重点大学计算机专业系列教材

计算机科学与技术概论

郭 平 朱郑州 王艳霞 编著

清华大学出版社
北京

内容简介

本书以计算技术与计算机的发展史为主线，介绍各发展阶段的计算需求以及为适应这样的需求所采用的主要计算技术、计算方法、关键人物及贡献，由此勾画出计算学科的基本轮廓，并分析了目前计算技术中存在的问题，介绍计算及计算机未来的发展方向。

本书适合作为高等院校计算机及相关专业导论性课程的教材，也可供感兴趣的读者阅读参考。本书是普通高等教育“十一五”国家级规划教材。

图书在版编目(CIP)数据

计算机科学与技术概论/郭平，朱郑州，王艳霞编著．—北京：清华大学出版社，2008.10
(重点大学计算机专业系列教材)
ISBN 978-7-302-17397-7

Ⅰ．计…　Ⅱ．①郭…　②朱…　③王…　Ⅲ．计算机科学—高等学校—教材　Ⅳ．TP3

中国版本图书馆 CIP 数据核字(2008)第 100524 号

责任编辑：付弘宇
责任校对：梁　毅
责任印制：何　芊
出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦 A 座
http://www.tup.com.cn　　**邮　　编**：100084
社　总　机：010-62770175　　**邮　　购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质　量　反　馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印　装　者：北京市清华园胶印厂
经　　销：全国新华书店
开　　本：185×260　**印　张**：13.5　**字　数**：334 千字
版　　次：2008 年 10 月第 1 版　　**印　　次**：2008 年 10 月第1次印刷
印　　数：1～3000
定　　价：21.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题，请与清华大学出版社出版部联系调换。联系电话：(010)62770177 转 3103　　产品编号：021297—01

INTRODUCTION

出版说明

随着国家信息化步伐的加快和高等教育规模的扩大，社会对计算机专业人才的需求不仅体现在数量的增加上，而且体现在质量要求的提高上，培养具有研究和实践能力的高层次的计算机专业人才已成为许多重点大学计算机专业教育的主要目标。目前，我国共有16个国家重点学科、20个博士点一级学科、28个博士点二级学科集中在教育部部属重点大学，这些高校在计算机教学和科研方面具有一定优势，并且大多以国际著名大学计算机教育为参照系，具有系统完善的教学课程体系、教学实验体系、教学质量保证体系和人才培养评估体系等综合体系，形成了培养一流人才的教学和科研环境。

重点大学计算机学科的教学与科研氛围是培养一流计算机人才的基础，其中专业教材的使用和建设则是这种氛围的重要组成部分，一批具有学科方向特色优势的计算机专业教材作为各重点大学的重点建设项目成果得到肯定。为了展示和发扬各重点大学在计算机专业教育上的优势，特别是专业教材建设上的优势，同时配合各重点大学的计算机学科建设和专业课程教学需要，在教育部相关教学指导委员会专家的建议和各重点大学的大力支持下，清华大学出版社规划并出版本系列教材。本系列教材的建设旨在“汇聚学科精英、引领学科建设、培育专业英才”，同时以教材示范各重点大学的优秀教学理念、教学方法、教学手段和教学内容等。

本系列教材在规划过程中体现了如下一些基本组织原则和特点。

1. 面向学科发展的前沿，适应当前社会对计算机专业高级人才的培养需求。教材内容以基本理论为基础，反映基本理论和原理的综合应用，重视实践和应用环节。

2. 反映教学需要，促进教学发展。教材要能适应多样化的教学需要，正确把握教学内容和课程体系的改革方向。在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养，为学生的知识、能力、素质协调发展创造条件。

3. 实施精品战略，突出重点，保证质量。规划教材建设的重点依然是专业基础课和专业主干课；特别注意选择并安排了一部分原来基础比较好的优秀教材或讲义修订再版，逐步形成精品教材；提倡并鼓励编写体现重点大学

计算机专业教学内容和课程体系改革成果的教材。

4. 主张一纲多本，合理配套。专业基础课和专业主干课教材要配套，同一门课程可以有多本具有不同内容特点的教材。处理好教材统一性与多样化的关系；基本教材与辅助教材以及教学参考书的关系；文字教材与软件教材的关系，实现教材系列资源配套。

5. 依靠专家，择优落实。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时，要引入竞争机制，通过申报、评审确定主编。书稿完成后要认真实行审稿程序，确保出书质量。

繁荣教材出版事业，提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量，希望有志于教材建设的教师能够加入到我们的编写队伍中来。

教材编委会

FOREWORD

前言

自计算机诞生以来，计算机的应用已经渗透到了各行各业，并且应用将进一步扩展。在我国，计算机专业教育随着计算机技术的发展已成为我国高校中最大的专业，开设计算机专业的学校最多，计算机专业的在校学生人数最多。

本书作为计算学科（或称计算机学科）导论性课程的教材，在内容组织上主要考虑了3个方面的因素。首先，计算机科学技术的发展为我们提供了大量的素材，要思考如何使选择的素材既适合于导论性质又能反映学科的整体发展。其次，计算学科的每个学科方向都涉及大量的知识，一本导论性的书不应该面面俱到，成为后继课程的索引。第三，计算学科的发展非常迅速，导论性课程应该在介绍学科基本知识的同时关注学科的发展方向，提高学生学习的兴趣。为此，本教材按计算学科发展的纵横两个方面来组织内容。纵的方面，以计算学科发展的历史为线索，涉及手工计算、机械计算、电子计算、智能计算和生物计算，力求阐明计算学科发展的历史必然性。横的方面，对计算学科发展的每个历史阶段，讨论其计算方法和计算工具。计算方法是计算的思想与灵魂，计算工具用于实现计算方法，计算工具的发展又催生新的计算方法产生。因此，计算方法与计算工具的发展变化成为计算学科发展的根本动力。

本书共分7章。

第1章介绍计算的概念及计算学科的基本内容。以CC2004为基础，介绍了计算学科的专业方向和知识领域，以及我国计算机学科的专业设置。

第2章讨论手工计算。从计算需求、工具的角度来认识手工计算；从计算能力、方便性、存储特征等角度来探讨手工计算的特点，并说明它们在计算学科形成发展中的启迪、影响、作用，以及可供借鉴的思想和方法。

第3章讨论计算的机械化。概括讨论机械式计算的历史和思想方法、计算机械化的意义以及对计算学科发展的影响，指出人类对计算机械化、自动化的渴求与探索。从帕斯卡的机械式加法机到朱斯的系列计算装置来讨论计算技术与计算工具的发展变化过程。

第 4 章讨论计算的自动化——计算机硬件。从计算理论(计算技术)和计算硬件(计算工具)的角度论述数字电子计算机出现的历史必然性，指出数字电子计算机应具备的功能和性能，以及为达到这样的功能与性能而形成的计算机硬件的基本组成、工作原理与实现方法。

第 5 章介绍计算自动化过程中的计算机软件。从方便管理和使用的角度介绍计算机软件组成与开发技术的发展。从计算过程的抽象、表示和实现方法等方面阐述了计算机软件设计的思想和过程。

第 6 章讨论计算的智能化。从人类逻辑思维过程模拟的角度介绍了计算智能化的原理，以人工智能、专家系统为例介绍了计算智能化的思想与实现方法。最后讨论了第五代计算机的研究开发过程，给出了它的系统结构和软件结构，分析了第五代计算机失败的原因。

第 7 章讨论了生物计算模型。生物体的并行计算能力是现代数字电子计算机无法比拟的。本章讨论了 3 种典型的生物计算模型(人工神经网络、DNA 计算与膜计算)的思想、原理、方法及实现技术。DNA 计算与膜计算的研究历史很短，要实现通用计算硬件还有许多的困难，但是它们的思想和原理是可以借鉴的。膜计算是从 2000 年才开始研究的生物计算模型。它把细胞层面上的化学反应与物质流动性当做计算过程来抽象，膜内的化学反应就是我们通常理解的计算过程，而物质在不同膜之间流动则相对于通常意义计算系统中的消息传递，整个生物体的皮肤相当于一个计算系统。已有的研究表明，膜计算可以以极大的并行度来实现，从而获得远远超过传统电子计算机的计算能力。

使用本书需要 36~ 48 学时，第 6、第 7 章可作为选讲的内容。讲授的过程中应注意增加一些科学家的故事、重要的历史事件等内容，以提高学生学习的兴趣，同时让学生感受科学家思考问题的方法。

在本书的编写和出版过程中得到了清华大学出版社和重庆大学教材出版基金的大力支持。在此对所有支持本书出版的领导、老师、学生和朋友们表示衷心的感谢。

由于时间紧迫，加之笔者水平有限，错误和笔误在所难免，敬请广大读者批评指正。本书的配套课件可以从清华大学出版社网站 http://www.tup.tsinghua.edu.cn 下载。如果在本书的使用中遇到任何问题，请联系 fuhy@ tup.tsinghua.edu.cn。

作　者

2008 年 5 月

CONTENTS

目录

第 1 章　绪论 …… 1

1.1　计算的概念 …… 1
1.1.1　计算与计算模型 …… 1
1.1.2　计算的复杂性 …… 4
1.1.3　计算的要素 …… 6
1.2　计算学科 …… 8
1.2.1　计算作为一门学科 …… 8
1.2.2　计算学科的知识领域 …… 10
1.2.3　计算学科的二维定义矩阵 …… 15
1.3　计算学科与计算机学科 …… 17
1.3.1　计算学科与计算机学科的关系 …… 17
1.3.2　我国计算机学科专业设置 …… 18
1.4　本章小结 …… 19
思考题 …… 19

第 2 章　手工计算 …… 21

2.1　记数与计算的需求 …… 21
2.1.1　实物记数 …… 21
2.1.2　结绳记数 …… 22
2.1.3　刻痕记数 …… 23
2.1.4　算筹记数 …… 24
2.2　算盘 …… 26
2.2.1　算盘的发展 …… 26
2.2.2　中国算盘指法 …… 28
2.2.3　中国算盘口诀 …… 28
2.2.4　会计算盘指法 …… 28
2.2.5　算盘小结 …… 29

2.3　数与计算式 …… 29
2.3.1　中国数字与数码 …… 29
2.3.2　阿拉伯数码 …… 30
2.3.3　数制的发展 …… 31
2.3.4　十进制与二进制 …… 35
2.3.5　计算式 …… 37
2.4　本章小结 …… 41
思考题 …… 41

第 3 章　计算的机械化 …… 42

3.1　手动齿轮计算装置 …… 43
3.1.1　帕斯卡的加法器 …… 43
3.1.2　莱布尼茨的乘法器 …… 44
3.1.3　机械计算器的商业化 …… 45
3.2　程序控制计算机 …… 45
3.2.1　杰卡德提花机 …… 45
3.2.2　巴贝奇的差分机和分析机 …… 47
3.3　模拟计算机 …… 50
3.4　机电计算机 …… 51
3.4.1　机电计算机的诞生环境 …… 51
3.4.2　霍尔瑞斯的电动制表机 …… 52
3.4.3　祖斯的 Z 系列计算机 …… 54
3.4.4　艾肯的自动顺序控制计算机 …… 56
3.5　数据的存储与输入输出 …… 58
3.6　计算的复杂性思考 …… 58
3.7　本章小结 …… 60
思考题 …… 60

第 4 章　计算的自动化——计算机硬件 …… 61

4.1　可计算性与图灵机 …… 61
4.1.1　可计算性理论的研究 …… 61
4.1.2　图灵等的贡献 …… 62
4.1.3　λ 演算 …… 65
4.2　ENIAC——第一台数字电子计算机 …… 65
4.2.1　电子管的发明 …… 65
4.2.2　ABC 计算机 …… 66
4.2.3　ENIAC 的诞生 …… 67
4.3　冯·诺依曼与计算机体系结构 …… 69
4.3.1　冯·诺依曼介入 ENIAC 的改进前后 …… 69

4.3.2 冯·诺依曼体系结构要点 …… 70
4.3.3 冯·诺依曼体系结构的意义 …… 73
4.3.4 冯·诺依曼体系结构的不足 …… 73
4.4 计算机硬件体系结构的发展 …… 74
4.4.1 电子管与第一代计算机 …… 74
4.4.2 晶体管与第二代计算机 …… 76
4.4.3 集成电路与第三代计算机 …… 79
4.4.4 VLSI 和第四代计算机 …… 82
4.4.5 人工智能与第五代计算机 …… 86
4.4.6 其他计算机 …… 87
4.5 计算机系统及其相关学科 …… 88
4.5.1 数字逻辑与数字电路 …… 88
4.5.2 集成电路与光电技术 …… 89
4.5.3 量子计算与分子计算 …… 91
4.5.4 信息网络与网络计算 …… 94
4.6 本章小结 …… 95
思考题 …… 95

第 5 章 计算的自动化——计算机软件 …… 96

5.1 计算的抽象与表示 …… 96
5.1.1 计算的抽象 …… 96
5.1.2 算法及其表示 …… 98
5.1.3 数据的表示 …… 101
5.2 计算机程序 …… 109
5.2.1 程序设计语言 …… 109
5.2.2 程序的解释与编译 …… 114
5.2.3 调试与运行 …… 115
5.3 计算机软件 …… 115
5.3.1 软件生存周期 …… 116
5.3.2 软件开发方法 …… 118
5.3.3 软件结构模式 …… 119
5.3.4 软件工具 …… 122
5.4 关于软件的思考 …… 125
5.4.1 软件危机 …… 126
5.4.2 软件工程 …… 127
5.4.3 知识产权和软件专利 …… 132
5.5 本章小结 …… 134
思考题 …… 134

第 6 章　计算的智能化 …… 136

6.1　机器智能 …… 136
6.1.1　人工智能的提出 …… 136
6.1.2　人工智能的发展历史 …… 137
6.1.3　智能测试 …… 139
6.2　智能计算的原理 …… 141
6.2.1　思维的计算机模拟 …… 141
6.2.2　推理系统的结构 …… 143
6.2.3　产生式系统 …… 145
6.2.4　专家系统 …… 148
6.3　第五代计算机 …… 153
6.3.1　研究目标与计划 …… 153
6.3.2　系统结构 …… 156
6.3.3　软件结构 …… 158
6.3.4　关于第五代计算机的思考 …… 160
6.4　本章小结 …… 161
思考题 …… 162

第 7 章　生物计算模型 …… 163

7.1　人工神经网络 …… 163
7.1.1　神经网络与人工神经网络 …… 163
7.1.2　神经网络计算机的特点 …… 167
7.1.3　神经网络计算机的实现方式 …… 168
7.1.4　重要的神经网络模型 …… 172
7.2　DNA 计算 …… 174
7.2.1　DNA 计算基础 …… 174
7.2.2　DNA 计算模型 …… 178
7.2.3　DNA 计算的实现 …… 182
7.3　膜计算 …… 187
7.3.1　膜结构 …… 187
7.3.2　膜计算模型 …… 188
7.3.3　膜计算的发展 …… 194
7.4　本章小结 …… 195
思考题 …… 195

附录　冯·诺依曼其人 …… 197

参考文献 …… 199

CHAPTER 1

第1章 绪　论

本章从计算的技术、方法与计算工具两个方面阐述计算的基本概念；从ACM等组织关于计算学科的研究出发，介绍计算学科的存在性与科学性；从计算机与计算、计算学科的相互联系探讨计算技术、计算方法的实现方法。

1.1 计算的概念

提到计算，我们首先想到的是加、减、乘、除这样的四则运算，甚至更复杂的函数运算。例如，求圆的面积需要计算；求一元二次方程的根需要计算；等等。可以说，我们每天都在和计算打交道。然而，究竟什么是计算呢？

1.1.1 计算与计算模型

抽象地说，计算可以看作是从一个符号行到另一个符号行的变换，也即从一个已知的符号行开始，按照一定的规则，逐步改变已知的符号行，经过若干步改变之后，得到一个满足预定要求的新的符号行。

例如，将符号行 1+1 变换成符号行 2，就是一个加法计算。然而，绝大多数的计算并不像将 1+1 变换成 2 这样简单。

当改变符号行的步骤为无穷步时，称这样的计算为无穷计算，否则为有穷计算。例如，计算

$$1+\frac{1}{2!}+\frac{1}{3!}+\cdots+\frac{1}{n!}+\cdots$$

为无穷计算；而对任意的正整数 n，计算

$$1+\frac{1}{2!}+\frac{1}{3!}+\cdots+\frac{1}{n!}$$

为有穷计算。对于人类的计算能力而言，有穷计算才真正有意义。

除了上述的数值型计算外，还有另一类计算，称为符号计算或符号推导。符号推导包括代数与各种函数的恒等式、不等式的证明，几何命题的证明，文字翻译等。但无论是数值计算还是符号推导，它们在本质上是等价的、一致

的，即两者可以相互转化，具有共同的计算本质。

人类揭示计算的本质是在20世纪30～40年代。这期间，数学家与数理逻辑学家相继提出了4种计算模型，它们是一般递归函数、λ 可计算函数（或称 λ 演算）、图灵（A. M. Turing）机模型和波斯特（E. L. Post，1897—1954）系统。

哥德尔（Kurt Godel）首先在1931年提出了原始递归函数的概念。所谓原始递归函数，就是由初始函数出发，经过有限次地使用代入与原始递归式而获得的函数。这里所说的初始函数是指下列三种函数之一。

（1）零函数——值恒为零的函数：$0(x)=0$。

（2）投影函数——值与第 i 个自变元的值相同的函数：$f_i(x_1,x_2,\cdots,x_n)=x_i(1\leqslant i\leqslant n)$。

（3）后继函数——值为自变元的直接后继数的函数：$S(x)=x+1$。

代入（又名叠置、迭置）是最简单又最重要的算子，其一般形式是由一个 m 元函数 f 与 m 个 n 元函数 $g_1,g_2,\cdots,g_m$ 构造成新函数，如下所示。

$$f(x_1,x_2,\cdots,x_n)=f(g_1(x_1,x_2,\cdots,x_n),g_2(x_1,x_2,\cdots,x_n),\cdots,g_m(x_1,x_2,\cdots,x_n))$$

原始递归式的一般形式为

$$\begin{cases}f(u_1,u_2,\cdots,u_n,0)=g(u_1,u_2,\cdots,u_n)\\f(u_1,u_2,\cdots,u_n,s(x))=h(u_1,u_2,\cdots,u_n,x,f(u_1,u_2,\cdots,u_n,x))\end{cases}$$

最简单的形式是

$$\begin{cases}f(u,0)=g(u)\\f(u,s(x))=h(u,x,f(u,x))\end{cases}$$

其特点是，不能由 g、h 两已知函数直接计算新函数的一般值 $f(u,x)$，而只能依次计算 $f(u,0)$、$f(u,1)$、$f(u,2)$，…；但只要依次计算，必能把任何一个 $f(u,x)$ 值算出来。换句话说，只要 g、h 有定义且可计算，则新函数 f 也有定义且可计算。

哥德尔于1934年引进了一般递归函数的概念，后经克林（S. C. Kleene，1909—1994）的改进与阐明，便形成了现在普遍采用的定义。所谓一般递归函数，就是由初始函数出发，经过有限次使用代入、原始递归式和 μ 算子而构成的有定义的函数。这里的 μ 算子就是构造逆函数的算子或求根算子。

例 1.1 设 n 是自然数。函数 $f(n)$ 定义为

$$f(n)=\begin{cases}1 & 当\ n=0\\nf(n-1) & 当\ n>0\end{cases}$$

那么，$f(n)$ 为递归函数。

例 1.2 设 m、n 为自然数。函数 $A(m,n)$ 定义为

$$A(m,n)=\begin{cases}n+1 & m=0\\A(m-1,1) & n=0\\A(m-1,A(m,n-1)) & m、n>0\end{cases}$$

则 $A(m,n)$ 称为阿克曼（Ackermann）函数，它是递归函数。

这样定义的一般递归函数比原始递归函数更广，但是，人们还是会问：这样定义的函数是否已经包括了所有直观上的可计算函数？如果还有更广的可计算函数又该怎样定义？在被这类问题困惑的同时，丘奇（Alonzo Church）、克林又提出了一类可计算函数，叫做 λ 可计

算函数。但事隔不久，丘奇和克林便分别证明了 λ 可计算函数正好就是一般递归函数，即这两类可计算函数是等价的、一致的。在这一有力的证据基础上，丘奇于 1936 年公开发表了他早在两年前就孕育出的一个论点，即著名的丘奇论点：每个能行可计算的函数都是一般递归函数。所谓“能行”是一个直观的概念，它指“能机械地进行并产生一个确定的结果”。

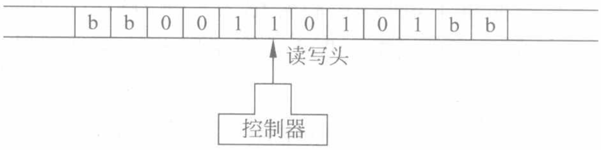

图 1.1 图灵机模型

与此同时，图灵提出了一种计算模型，如图 1.1 所示，称为图灵机模型。它由一条无限长的纸带和一个读写头构成。读写头根据其当前的状态和纸带中对应的符号来确定要完成的操作。这些操作包括：改写纸带中的符号、修改读写头的状态和确定读写头的移动方向。可以把图灵机模型抽象地描述为

$$TM = \{\sum, S, D, Q\}$$

TM 满足的条件如下。

(1) $\sum$ 为非空有限符号集，称作带符号集。

(2) S 为非空有限集，称作状态符号集。

(3) $D=\{B,R,L\}$ 为读写头移动的方向集，B 表示读写头不移动，R 表示右移，L 表示左移。

(4) $Q \subseteq S \times \sum \times \sum \times D \times S$。

(5) 若 (p,a,b,m,q)、$(p,a,c,n,r) \in Q$，则有 $b=c$，$m=n$ 且 $q=r$。

$\sum$ 的元素是写在带上的符号，称为带符号；S 的元素表示控制器的状态，称为状态符号。集合 $S \times \sum \times \sum \times D \times S$ 的元素称为基本操作，基本操作 $\{p,a,b,r,q\}$ 简记为 $pabrq$。当 $pabrq \in Q$，称 pa 属于 Q 的定义域。Q 是图灵机的指令表，基本操作 $pabrq$ 的含义是：如果当前的状态为 p 且读写带的符号为 a，则带中的符号 a 改写为符号 b，读写头按 r 的值进行移动并将状态修改为 q。

条件(4)表示指令集无矛盾，即对同一状态不能发出超过一个的动作指令。

例 1.3 M 的字母表 $A=\{0,1,b\}$，b 表示空格。状态集 $Q=\{q_1,q_2,q_3\}$，其中，指定 q_1 是开始状态，q_3 是终止状态。M 的控制器的命令如下：

$$\begin{array}{ccccc}
q_1 & 0 & 1 & \mathrm{R} & q_1; \\
q_1 & 1 & 0 & \mathrm{R} & q_1; \\
q_1 & \mathrm{b} & \mathrm{b} & \mathrm{R} & q_2; \\
q_2 & \mathrm{b} & \mathrm{b} & \mathrm{L} & q_3; \\
q_2 & 0 & 0 & \mathrm{B} & q_1; \\
q_2 & 1 & 1 & \mathrm{B} & q_1;
\end{array}$$

设 M 的输入是 1100b0011，读写头对准左边第一个 1，状态为 q_1。问：输出是什么？

根据图灵机的操作过程可知，输出状态为 0011b1100。

由图灵机模型描述的计算称为图灵机可计算函数。图灵机是一台理论计算机，它是人类对计算与机器计算的最一般、最高级的抽象。

图灵进一步证明了图灵机可计算函数与 λ 可计算函数是一致的，因此也就和一般递归函数一致、等价。于是，表面上不同的三类可计算函数在本质上就是同一类。这样一来，丘奇论点和图灵论点也就是一回事了，现将它们合称为丘奇-图灵论点，即直观的可计算函数等同于一般递归函数、λ 可定义函数、图灵机可计算函数及波斯特系统。关于 λ 可计算函数与波斯特系统的介绍可参阅相关资料。

丘奇-图灵论点的提出与确认，在数学和计算学科发展史上具有重大的理论和现实意义。正如我国数理逻辑专家莫绍揆教授所言，有了这个论点以后，就可以断定某些问题是不能能行地解决或不能能行地判定的。也就是说，丘奇-图灵论点回答了什么是可以计算的。

对于计算机学科，丘奇-图灵论点的意义在于它明确刻画了计算机的本质或计算机的计算能力，确定了计算机能计算的函数是一般递归函数，对于一般递归函数之外的函数，计算机是无能为力的。

丘奇-图灵论点的提出，标志着人类对计算本质认识的深入，它是数学与计算历史上的一块里程碑。

1.1.2 计算的复杂性

丘奇-图灵论点指出了可计算函数的本质特征，然而对应不同的可计算函数，计算它所花费的代价是不同的，先看下面两个例子。

例 1.4 哥尼斯堡七桥问题。

17 世纪的东普鲁士有一座哥尼斯堡（Konigsberg）城，城中有一座奈佛夫（Kneiphof）岛，普雷格尔（Pregol）河的两条支流环绕其旁，将整个城市分成北区、东区、南区和岛区 4 个区域。全城共有 7 座桥将 4 个城区相连起来，如图 1.2 所示。人们常通过这 7 座桥到各城区游玩，于是产生了一个有趣的数学难题：寻找走遍这 7 座桥，且只许走过每座桥一次，最后又回到原出发点的路径。该问题就是著名的“哥尼斯堡七桥问题”。

关于该问题的求解，1736 年数学家列昂纳德·欧拉（L. Euler）发表了关于“哥尼斯堡七桥问题”的论文 *Solutio Problematis ad Geomertriam Situs Pertinentis*。他将问题进行了如下抽象。

（1）用顶点表示区域。如 A 表示岛区，B 表示东区，C 表示北区，D 表示南区。

（2）用边表示桥。如岛区到东区有一座桥相连，就建立一条边 AB；北区与岛区有两座桥相连，建立两条边 AC；等等。

经抽象化后，哥尼斯堡七桥与各区域间的连接关系可表示为图 1.3 的形式。问题的描

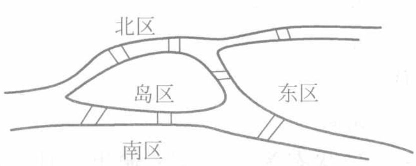

图 1.2　哥尼斯堡七桥问题示意图

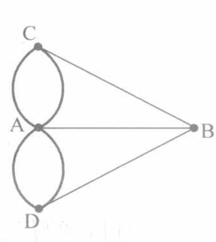

图 1.3　哥尼斯堡七桥问题的欧拉图表示

述抽象为在图 1.3 中找一条经过每条边一次且仅一次又回到出发顶点的路径。欧拉在论文中论证了哥尼斯堡七桥问题的解不存在。进一步地，对类似问题，欧拉得到了一般性的结论如下。

（1）如果与奇数条边相连的顶点数超过两个，那么不存在经过每条边一次且仅一次又回到出发顶点的路径。

（2）如果只有两个顶点与奇数条边相连，可以从这两个顶点之一出发，都能找到经过每条边一次且仅一次的路径。

（3）如果没有顶点与奇数条边相连，可从任意顶点出发，都能找到经过每条边一次且仅一次又回到出发顶点的路径。

后来，人们把这样的路径称为欧拉路径。当路径的起始顶点与终止顶点相同时，称为欧拉回路。有欧拉路径的图被称为欧拉图。

假设图的边数为 n，顶点数为 m，根据欧拉的工作，确定一个图是否为欧拉图，最多需要 $2\times m\times n$ 次运算。从欧拉图中找出欧拉路径需要的运算次数不超过 $m\times n$ 次。另一方面，图中的边的数目一般多于顶点的数目。因此，从一个图中找出欧拉路径（如果存在），运算次数不超过 n^2 的一个常数倍，记为 $O(n^2)$。如果用计算机来完成这个计算问题，假设计算机执行任意一次基本运算的时间上限为 c（c 是一个常数），那么计算机所花的时间将不超过 n^2 的一个常数倍，仍记为 $O(n^2)$。

例 1.5　旅行商问题。

对寻找欧拉路径的问题稍做修改为：有若干个城市，任意两城市间的交通费已知。某销售员要去每个城市推销产品，他需要寻找一条经过每个城市一次且仅一次并最后回到出发城市的费用最小的路径。这个问题被称为巡回销售员问题或旅行商问题（Traveling Salesman Problem，TSP）。所求出的费用最小的路径称为哈密尔顿回路。

巡回销售员问题可类似于哥尼斯堡七桥问题，可抽象为图的问题。例如，图 1.4 为有 5 个城市的情况。图 1.4 中的顶点 A、B、C、D、E 为城市，顶点间连接的边为交通线，其上的数字为相应两个城市间的交通费。

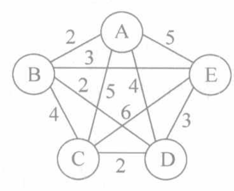

图 1.4　5 个城市交通图

寻找哈密尔顿回路的一种简单方法是：对给定的起始顶点，列出所有可能的路径，然后比较它们的费用，从中选出费用最少的路径。假设有 n 个城市，可能的路径条数为 $(n-1)!$，计算每条路径的费用至少需要 n 次计算，因此，计算的总次数不少于 $n!$。随着城市数目的增加，寻找哈密尔顿回路的计算次数呈指数级数增长，以至于达到无法计算的程度，这就是所谓的“组合爆炸”问题。例如，$n=20$ 时，可能的路径约为 1.216×10^{17} 条。若计算机每秒能找出 1000 万条路径，找出哈密尔顿回路约需要 386 年！

从例 1.4 和例 1.5 可以看出，完成不同的计算问题其计算的代价是不相同的，即计算的复杂程度是不同的。

一般而言，一个问题的计算复杂程度（或称计算复杂度）包括时间复杂度与空间复杂度两个方面。所谓时间复杂度指完成计算任务所花时间与问题规模的关系。例 1.4 和例 1.5 中的问题的规模分别是图的边数和图的顶点数。

在时间复杂度研究中，将所有可以在多项式时间内求解的问题称为 P 类问题，将所有

在多项式时间内可以验证的问题称为NP类问题(Non-deterministic Polynomial,多项式复杂程度的非确定性问题)。

显然,按例1.4的方法确定一个图是否为欧拉图是P类问题,因为它可以在$O(n^2)$时间内求解。按照例1.5中的方法,在一个图中找哈密尔顿回路所花的时间不是问题规模的多项式,但是却可以在$O(n)$时间内验证一条回路是否为哈密尔顿回路,因此旅行商问题不是P类问题而是NP类问题。

对NP类问题的进一步研究发现,NP类中某些问题的复杂性与整个类的复杂性有关,如果这些问题中的任意一个能在多项式时间内求解,则所有NP类问题都能在多项式时间内求解,这些NP类问题被称为NP完全性问题或NP难问题。

P类、NP类与NP完全类三类问题之间有如下关系:

$$\text{P类问题} \subseteq \text{NP完全类问题} \subseteq \text{NP类问题}$$

至今为止,一直没有证明P类与NP类是否相等。也就是说,例1.5中的方法是非多项式时间的,但还不知道该问题是否有多项式时间算法。已经确定的是哈密尔顿回路问题是NP类完全问题。

计算的空间复杂度指计算中使用的存储资源数量与问题规模的关系。例如,判定一个图是否为欧拉图需要的存储资源与n^2成正比。哈密尔顿回路问题中,如果计算中仅保存到当前为止费用最少的路径和图,那么需要的存储资源与n^2成正比。由此可以看出,时间复杂度与空间复杂度从不同的方面反映了待求解问题的复杂程度,它们之间没有必然的联系。对待求解问题来讲,时间复杂度是进行问题复杂度分析的主要方面。

丘奇-图灵论点确定了可计算问题的范围,计算复杂度研究确定了求解可计算问题需要花费的代价。问题的可计算性和计算的能行性是计算学科研究的基本问题。

1.1.3 计算的要素

在人类发展的长河中,虽然计算的本质从20世纪30年代起才被人类所认识,但是人类进行各种计算的历史却是源远流长的。

古人利用石头、手指、绳结等进行简单的记数与计算,而后算筹则是最简单的计算工具。古语曰:"运筹策于帷幄之中,决胜败于千里之外"。筹策即算筹,它是中国古代普遍采用的一种计算工具,通常是用竹、木或骨制成的颜色不同的小棍。算筹不仅可以替代手指来帮助计数,而且能做加减乘除等数学运算。计算每一个数学问题时,通常编出一套歌诀形式的计算方法,一边计算,一边不断地重新布棍。中国古代数学家祖冲之,就是用算筹计算出圆周率在3.1415926和3.1415927之间。由此,布棍方法与算筹成为计算的基本要素。

中国古代在计算领域的另一项发明是珠算盘及其计算规则。珠算盘最早记录于汉朝人徐岳撰写的《数术记遗》一书里,大约在宋元时期开始流行,算盘最终彻底淘汰筹算是在明代完成的。由于珠算具有"随手拨珠便成答数"的优点,一时间风靡海内,并且逐渐传入日本、朝鲜、越南、泰国、印度和美国等地,此后又经一些商人和旅行家带到欧洲,逐渐向西方传播,受到广泛欢迎。

到17世纪初,计算工具在西方也呈现了较快的发展。闻名于世的英国数学家纳皮尔(J. Napier),在他所著的一本书里,介绍了一种新工具,即后来被称为"纳皮尔算筹"的器具。据说,纳皮尔的这种器具发明于1612年,它由一些长条状的木棍组成,木棍的表面雕刻