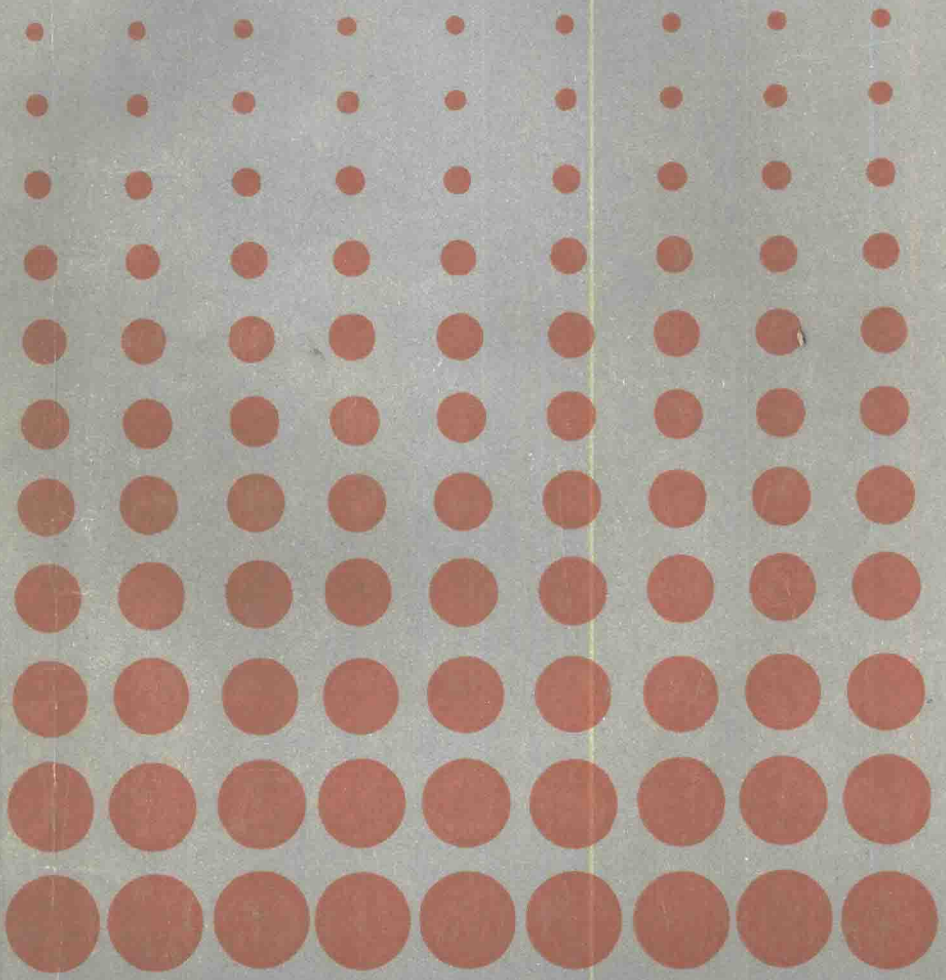
数理逻辑

（计算机类专业适用）

莫绍揆　徐永森　沈百英



SHULI LUOJI

高等教育出版社

数 理 逻 辑

（计算机类专业适用）

莫绍揆　徐永森　沈百英

高等教育出版社

提　要

数理逻辑与计算机科学有着密切的关系，是计算机科学的重要理论基础。本书介绍了数理逻辑最基本、最重要的概念，方法和理论，并且介绍了它在计算机科学中的一些应用。

本书共分三章。第一章是命题演算，第二章是谓词演算，第三章是递归函数。它们包括命题、谓词、命题联结词、量词、公理系统、算子、原始递归函数、一般递归函数、幕状函数、能行可计算性等概念、方法和理论。

本书可作为高等学校计算机科学系数理逻辑课程的教材或参考书，也可供其他有关人员学习参考。

责任编辑：鲍涌

数理逻辑

（计算机类专业适用）

莫绍揆　徐永森　沈百英

高等教育出版社出版

新华书店北京发行所发行

北京新华印刷厂印装

*

开本 850×1168 1/32　印张 8　字数 189,000

1984 年 7 月第 1 版　1985 年 4 月第 1 次印刷

印数　00,001—16,700

书号 13010·01025　　定价 1.60 元

序　言

计算机及计算机科学与数理逻辑有着十分密切的关系．人们说数字电子计算机是数理逻辑与电子学结合的产物，这话不假．无论是作为数字电子计算机雏型的图灵机器，还是作为设计数字电子计算机的数学工具的布尔代数，都离不开数理逻辑．人们还说数理逻辑是计算机科学的理论基础，这话在理．无论是作为计算机科学的核心的算法，还是作为程序设计工具的语言，无论是程序设计方法学，还是计算复杂性理论，都涉及到数理逻辑的知识和理论．因此人们公认数理逻辑是计算机科学系的重要基础课程，是计算机科学系学生必须掌握的基本理论．

数理逻辑包括的内容很多，除了最基础的逻辑演算外，还包括证明论、递归论、模型论和公理集合论．证明论主要研究数学理论系统的相容性（即不矛盾性、协调性）的证明．递归论是能行可计算性的理论，它为能行可计算的函数找出各种理论上精确化的严密的类比物．自从发明电子计算机后，迫切需要在理论上弄清楚电子计算机能计算哪些函数，因此能行性理论（即递归论）更为人们所重视．模型论主要是对各种数学理论系统建立模型，并研究各模型之间的关系以及模型与数学系统之间的关系等．公理集合论是在消除已知集合论悖论的情况下用公理方法把有关集合的理论充分地发展下去．

我们认为，计算机科学系的学生没有必要也不可能去学习数理逻辑的全部内容，而是应该把数理逻辑的最重要、最基本并且跟计算机科学关系最密切的内容讲深讲透．从目前看，逻辑演算及递归论与计算机科学关系最为密切，尤其是逻辑演算．它的重要

性不仅在于它在计算机科学各个方面的广泛应用，而且在于它的一套完整的严格的形式的公理化方法．这套方法对培养学生的抽象思维能力、逻辑推理能力和严密的分析问题解决问题的能力都起着重要作用．因此本书把重点放在逻辑演算上，放在逻辑演算的推理研究上，此外还有递归函数的基本理论．

在本书的引论中，我们对符号体系作了简单讨论．目的是让读者从一开始就对符号体系问题引起足够的重视．

第一章从形式和非形式两个不同角度讨论命题、命题联结词、命题演算公式及其永真性等概念和理论．介绍了永真推理过程和日常推理过程，以及两者之间的关系．

第二章叙述谓词、量词、自由变元、约束变元、谓词演算公式及其永真性等概念和理论，给出了一个谓词演算永真公式的公理系统，并对它进行了讨论．

第三章首先介绍数学归纳法，然后介绍构造函数的方法，讨论函数的分类和各种函数集，叙述原始递归函数和一般递归函数的一些重要性质，最后简单讲解能行可计算性．

本书介绍了数理逻辑在计算机科学上的一些应用，其目的是使学生从中受到一些启示，但是本书没有详细介绍各种应用，我们认为重要的是把数理逻辑的基本内容掌握好，这样不但可以保证在学习其它课程和阅读文献资料时不致对其中的数理逻辑知识产生困难，而且可以为读者自己开拓更多更新的应用工作打下良好基础．

我们诚恳希望读者对本书的形式和内容、以及不妥之处提出批评指正．

作　　者

一九八三年七月于南京

目　录

引　论

符号体系在本课程中具有十分重要的作用. 特别是在数理逻辑中, 正是由于引进了一整套符号体系, 并用它来研究推理过程的规律, 才使得对推理过程规律的研究达到了一个新阶段, 从而形成数理逻辑这门独立的学科, 由此又得名符号逻辑. 基于符号体系在本课程中的重要作用, 因此有必要首先对符号体系作一些初步探讨.

数学中使用的符号大体上可以分成四类. 第一类是数量符号. 例如 0, 1, 2, e, π, x, y, a_{ij} 等等.

第二类是运算符号. 例如: $+$, $-$, $\times$, $\div$, Σ, Π, sin, cos 等等.

第三类是关系符号. 例如: $>$, $<$, $=$ 等等.

第四类是辅助符号. 即(,), [,], {, } 等各类括号.

前三类符号均表示一定的内容, 必要性是十分明显的。在数学式子中这三类符号除乘号外, 都是不能省略的. 后一类符号, 即各类括号, 它们比较特殊, 一般说来它们并不表示什么内容, 但却担负着重要的作用, 它们是用来确定式子中各种运算和关系的先后次序的. 例如:

$$(((a+b)+(c\times d))\div(e+(f\times h)))$$

表示先运算	$a+b$
再运算	$c\times d$
再运算	$(a+b)+(c\times d)$
再运算	$f\times h$
再运算	$e+(f\times h)$
最后运算	$((a+b)+(c\times d))\div(e+(f\times h))$

我们知道，当一个式子比较复杂，括号一层层套得很多时，不仅写起来不方便，而且读起来更是不方便，不易判定括号之间的匹配关系．因此通常人们都尽量设法减少甚至完全避免括号的使用．如何达到这个目的呢？这就是这里要讨论的问题．

减少括号的最常用方法是给出若干约定，以规定各种运算和关系的实施次序．例如，约定在任意一个不含括号的式子中，乘除运算先于加减运算(即先乘除后加减)，在同类运算中左边的运算符先运算，右边的运算符后运算(即左结合)．这样上面的例子便可改写成为

$$(a+b+c\times d)\div(e+f\times h)$$

这个式子中的括号比之原式减少了．因为这种减少括号的方法主要是通过约定运算符的实施次序，即哪种运算符优先运算，哪种运算符在后运算，所以通常把这种方法称为优先级法．

采用优先级法可以减少括号，但不能完全避免使用括号．为了阐明这一点，我们对现行符号体系作进一步的考察．

大家知道，孤零零的一个运算符或关系符是没有意义的，每个运算符和关系符都必须和它的变目(即运算分量)连在一起使用．运算符关系符与它们的变目之间的书写方式有三种：前置式，后置式和中置式．

前置式就是把运算符写在其变目之前．例如：$\sin x$，$\cos x$，$\log x$．

后置式就是把运算符写在其变目之后．例如：$n!$．

中置式就是把运算符写在其变目的中间．例如：$x+y$，$x\div y$．

容易看出，当一个式子中同时使用两种或两种以上方式时，不论对运算符的优先级作怎样的规定，括号都是不能完全避免的．例如，$\log n!$ 有两种解释：$(\log n)!$ 和 $\log(n!)$．无论规定 log 的优先级大于 ! 的优先级，还是规定 ! 的优先级大于 log 的优先级，都只

能表示其中之一，另一个必须使用括号才能表示.

即使一个式子中只使用中置式，采用优先级法，括号也是不能完全省略的。例如，$a\times b+c$也有两种解释：$(a\times b)+c$ 和 $a\times(b+c)$. 无论规定先乘除后加减还是规定先加减后乘除，都只能表示其中之一，另一个必须使用括号才能表示.

因此，采用优先级法，括号仍旧是不能缺少的.

然而仔细观察就可以看出，舍弃括号代之以其它辅助符号是可能的. 点子法便是其中的一种简明而有效的方法. 这种点子法可以概述如下：

1. 点子附在运算符的两旁(或左或右或左右均有)；

2. 运算符的运算次序按其两旁点子的总数决定，点子少的先运算，点子多的后运算。

显然按此方法便可把式子中的运算符的运算次序完全确定下来. 例如，加法结合律可以表示为

$$a+b+\cdot c\cdot=\cdot a\cdot+b+c$$

乘对加的分配律可以表示为

$$a\times\cdot b+c\cdot=\cdot a\times b\cdot+a\times c$$

根据上述方法，我们可以很容易地把使用括号的式子改用点子法表示，反之也容易把使用点子法的式子改用括号表示.

括号法和点子法各有优缺点. 对括号法而言，因为括号是成对地使用，所以当括号对数少时一部分一部分分得比较清楚，但当括号一层一层套得很多时，括号之间的匹配关系就不易看清楚. 对点子法而言，不存在匹配问题，运算的先后次序看得比较清楚，但当点子太多时也会模糊，而且该方法不适合于前置式和后置式. 因此，在数学式子里最好点子法和括号法并用，在前置运算符和后置运算符旁边不用点子，专用括号，点子太多时，可以内层先使用点子，中间兼用括号，括号外边再使用点子，而且括号外的点子

重新从一个点子开始逐步增加.

括号也好点子也好都是数学式子中的辅助符号，都是用来指明运算符的运算次序的. 问题是式子中的运算次序是不是非要括号点子这些辅助符号来指明不可呢？不是的，不用任何辅助符号还是能够唯一地确定运算符的运算次序的.

先看只含前置运算符的式子. 例如，sin cos log$x$，这个式子中有三个前置运算符，它们的运算次序显然只有唯一一种：先运算log，再运算cos，最后运算sin. 不存在其它解释. 同样任何只含前置运算符的式子的运算次序都是唯一确定的.

对于只含后置运算符的式子，同只含前置运算符的式子一样，运算次序也是唯一确定的.

由此可知，只要对现行数学中使用的符号作一番改造，把中置运算符和后置运算符全部改成前置运算符（或者把中置运算符和前置运算符全部改成后置运算符），式子中便可以不用辅助符号. 例如，若把

$$a+b \quad \text{改写成} \quad +ab$$
$$a\times b \quad \text{改写成} \quad \times ab$$
$$a=b \quad \text{改写成} \quad =ab$$

则加法结合律

$$(a+(b+c))=((a+b)+c)$$

可改写成 $=(a+(b+c))((a+b)+c)$

又可改写成 $=+a(b+c)+(a+b)c$

最后改写成 $=+a+bc++abc$

这个式子里不含有任何括号也不含有任何其它辅助符号，而各运算的次序是唯一确定的.

乘对加的分配律

$$(a\times(b+c))=((a\times b)+(a\times c))$$

可改写成　　$=(a\times(b+c))((a\times b)+(a\times c))$

又可改写成　$=\times a(b+c)+(a\times b)(a\times c)$

最后改写成　$=\times a+bc+\times ab\times ac$

这个式子里也没有任何辅助符号，而各运算符的运算次序是唯一确定的.

这种使用前置式来书写表示数学式子的方法称为前置法. 同样可以使用后置式来书写表示数学式子. 用后置式来书写表示数学式子的方法称为后置法. 因为前置法和后置法是波兰逻辑学家鲁卡塞维茨(J. Lukasiewicz)首先提出并采用的,所以人们又把前置法称为波兰表示法，而把后置法称为逆波兰表示法. 由于用后置法表示的数学式子便于计算机进行运算处理，所以在计算机里常常采用后置法.

下面举几个例子来说明各种表示法之间的"翻译"过程.

例 1：将下式改用前置法表示:

$$(((a+b)\times c)\div d)-(e\times f)$$

[解] "翻译"过程如下:

$$-(((a+b)\times c)\div d)(e\times f)$$

$$-\div((a+b)\times c)d\times ef$$

$$-\div\times(a+b)cd\times ef$$

$$-\div\times+abcd\times ef$$

例 2：将下式改用括号法和点子法表示:

$$\times a-+b\div cd\div -ab\times -ac-ad$$

[解] "翻译"过程如下:

$$\times a-+b(c\div d)\div(a-b)\times(a-c)(a-d)$$

$$\times a-(b+(c\div d))\div(a-b)((a-c)\times(a-d))$$

$$\times a-(b+(c\div d))((a-b)\div((a-c)\times(a-d)))$$

$$\times a-((b+(c\div d))-((a-b)\div((a-c)\times(a-d))))$$

$$a\times((b+(c\div d))-((a-b)\div((a-c)\times(a-d))))$$

此式采用括号法表示．由它易改为点子法表示

$$a{:}\times{:}b+\cdot c\div d\cdot-{:}a-b\cdot\div\cdot a-c\times\cdot a-d$$

习　题

把下列各式用括号法，点子法，括号点子法和前置法四种方法来表示．

1. $((a\div(b\times(((c+d)-e)\times f)))-g)$

2. $(2d+4\times(ab-4xy))\div\dfrac{4xyz}{x+2y-z}$

注意，这里采用了先乘除后加减的约定，而且有些乘号省略了．

3. $2+\cdot4-y{:}+{:}u-4\cdot+{:}4+5\cdot\times\cdot7-5\div\cdot3$

4. $d+\cdot a\times b+{:}c\times f\cdot+{:}g{:}\times{:}h-e$

5. $+-+\div u3\times\div uv\times-\div2w3v5t$

6. $-+\div+2v\times4t\div\times4uvw$

第一章 命题演算

§1.1 命题与真值联结词

凡是可分辨真假的语句均称为命题. 例如:"银是白的"(真),"9 为质数"(假),"5 大于 3"(真), 均是命题. 又例如:"滚出来","祝您健康","多么香啊",这些语句无真假可言,所以不是命题. 有些语句至今尚无法确定其真假, 例如:"太阳系外有宇宙人","$2^{\sqrt{\pi}}$为代数数","在π的小数展式中,符号串 12345 出现偶数次"等等. 人们从日常经验中知道,这些语句本身是具有真假的,只是目前人们尚无法确定其真假而已, 而且可以说在人类知识发展的历史长河中, 总有一天可以确定这些语句的真假. 这类目前尚不知道其真假, 但本身必可分辨真假的语句也称为命题. 真的语句称为真命题,或说命题的值为真,假的语句称为假命题, 或说命题的值为假. 这就是说命题的值指的是命题的真假性. "银是白的"和"5大于 3" 在内容上是两个不同的命题, 但是它们值是相同 的, 都是"真". 今后,我们一般用 α, β, γ 等小写希腊字母表示命题, 用 T 表示"真"值,用 F 表示"假"值.

命题可以分为两类,一类是原子命题,一类是复合命题.

所谓复合命题是指由旧命题组成的新命题, 而且新命题的真假完全由旧命题的真假决定. 旧命题称为新命题的成分 命 题. 由旧命题组成新命题时所用的东西称为真值联结词, 也称为命题联结词, 有时简称为联结词. 例如, 由"银是白的"利用"不"可以组成复合命题"银不是白的"; 由 "昨天下雨" 和"昨天打雷"利用"或者"可以组成复合命题"昨天下雨或者(昨天)打雷",利用"且"可以

组成复合命题“昨天下雨且(昨天)打雷”. 显然这三个复合命题的真假完全由其成分命题决定, 其中的“不”, “或者”, “且”便是真值联结词.

最重要最常用的真值联结词有五个:

1. 非

利用该真值联结词可以由成分命题 α 组成复合命题“非 α”, 记为 $\bar{\alpha}$, 或记为 $N\alpha$, $\neg\alpha$, $\sim\alpha$. “非 α”的真假与 α 的真假的关系定义如下:

$\bar{\alpha}$ 真　当且仅当　α 假

也可以列表定义如下:

α	$\bar{\alpha}$
T	F
F	T

$\bar{\alpha}$ 称为 α 的否定式. 从逻辑的角度看, 日常用语中的“不”, “无”, “没有”等否定词汇均与“非”相当.

2. 且

利用该真值联结词可以由成分命题 α 和 β 组成复合命题“α 且 β”, 记为 $\alpha\wedge\beta$, 或记为 $K\alpha\beta$, $\alpha\cdot\beta$, $\alpha\&\beta$. “α 且 β”的真假与 α, β 的真假之间的关系定义如下:

$\alpha\wedge\beta$ 真　当且仅当　α 与 β 均真

也可以列表定义如下:

α	β	$\alpha\wedge\beta$
T	T	T
T	F	F
F	T	F
F	F	F

$\alpha\wedge\beta$ 称为 α 与 β 的合取式, 而 α 与 β 称为该合取式的合取项. 从

逻辑的角度看，日常用语中的“并且”，“以及”，“和”，“不仅…而且…”，“虽然…但是…”，“尽管…仍然…”等词汇均与“且”相当.

3. 或

利用该真值联结词可以由成分命题 α 和 β 组成复合命题“α 或 β”，记为 $\alpha \vee \beta$，或记为 $A\alpha\beta$，$\alpha+\beta$. “α 或 β”的真假与 α，β 的真假之间的关系定义如下：

$\alpha \vee \beta$ 假　当且仅当　α 与 β 均假

也可以列表定义如下：

α	β	$\alpha \vee \beta$
T	T	T
T	F	T
F	T	T
F	F	F

$\alpha \vee \beta$ 称为 α 与 β 的析取式，而 α 与 β 称为该析取式的析取项.

4. 如果…则…

利用该真值联结词可以由成分命题 α 和 β 组成复合命题“如果 α 则 β”，记为 $\alpha \supset \beta$，或记为 $C\alpha\beta$，$\alpha \rightarrow \beta$. “如果 α 则 β”的真假与 α，β 的真假之间的关系定义如下：

$\alpha \supset \beta$ 假　当且仅当　α 真且 β 假

也可以列表定义如下：

α	β	$\alpha \supset \beta$
T	T	T
T	F	F
F	T	T
F	F	T

$\alpha \supset \beta$ 称为 α 与 β 的(实质)蕴涵式，α 称为该蕴涵式的前件，β 称为该蕴涵式的后件. 从逻辑角度看，日常用语中的“如果…必须…”，“必须…以便…”等词汇均与“如果…则…”相当.

5. 等价

利用该真值联结词可以由成分命题α和β组成复合命题“α等价于β”，记为“$\alpha\equiv\beta$”，或记为$E\alpha\beta$，$\alpha\leftrightarrow\beta$.“$\alpha$等价于$\beta$”的真假与$\alpha$，$\beta$的真假之间的关系定义如下：

$\alpha\equiv\beta$真　当且仅当　α与β均真或均假

也可以列表定义如下：

α	β	$\alpha\equiv\beta$
T	T	T
T	F	F
F	T	F
F	F	T

$\alpha\equiv\beta$称为α与β的(实质)等价式，α称为该等价式的左端，β称为该等价式的右端. 从逻辑角度看，日常用语中的“当且仅当”，“相当于”，“…和…一样”等词汇与“等价”相当.

任一命题，若其中不再含有真值联结词，则说该命题是原子命题. 就其意义来说，原子命题是不能再行分析的命题.“银是白的”，“9是质数”等均是原子命题. 今后我们一般用p，q，r等小写拉丁字母表示原子命题.

利用上面介绍的这些符号可以把许多日常语句用符号公式来表示. 现举几个例子来说明.

例1：昨天下雨并且打雷.

[**解**] 设p表示“昨天下雨”

q表示“昨天打雷”

原句可表为　$p\wedge q$，即Kpq.

例2：他虽有理论知识但无实践知识.

[**解**] 设p表示“他有理论知识”

q表示“他有实践知识”

原句可表为　$p\wedge\bar{q}$，　即　$KpNq$.