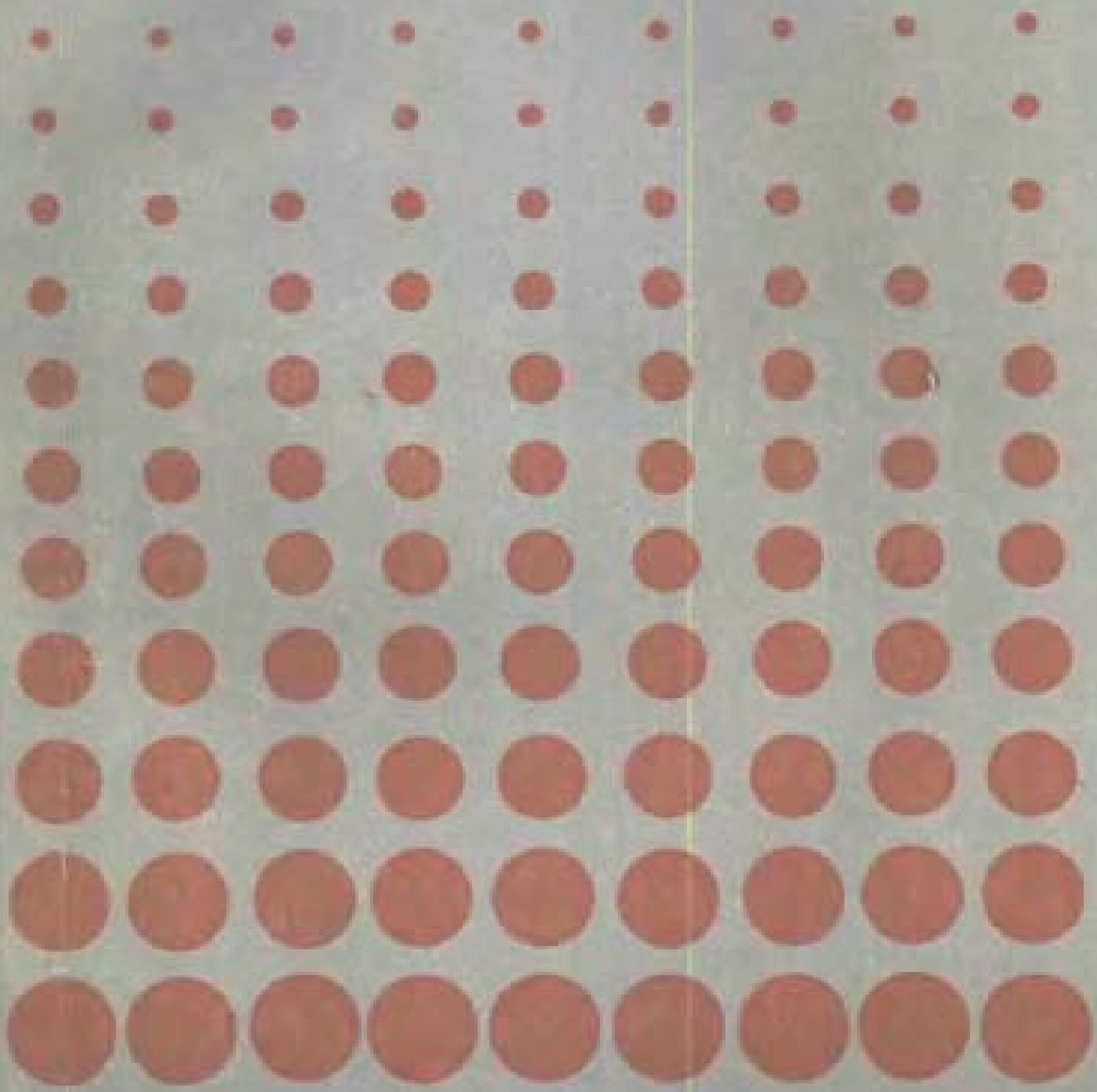
数理逻辑

（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：鲍浦

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 $0, 1, 2, e, \pi, x, y, a_{ij}$ 等等.

第二类是运算符号. 例如: $+, -, \times, \div, \Sigma, \Pi, \sin, \cos$ 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:

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

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

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

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

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

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 $C\alpha\beta$，$\alpha \to \beta$. “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 α，β 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　$p\wedge q$，　即　$KpNq$.

例 3：如果 a 大于 b，c 不大于 0，则 $a \cdot c$ 不大于 $b \cdot c$。

[解] 设 p 表示“a 大于 b”

q 表示“c 大于 0”

r 表示“$a \cdot c$ 大于 $b \cdot c$”

原句可表为 $(p \wedge q) \supset r$，即 $CKpNqr$。

例 4：铁和氧化合，但是铁和氮不化合。

[解] 设 p 表示“铁和氧化合”

q 表示“铁和氮化合”

原句可表为 $p \wedge \bar{q}$，即 $KpNq$。

应该注意的是原句决不能用如下的符号公式表示：

p_1 表示“铁化合”

q_1 表示“氧化合”

r_1 表示“氮化合”

而原句表为 $(p_1 \wedge q_1) \wedge (\bar{p}_1 \wedge \bar{r}_1)$，即 $KKp_1q_1KNp_1Nr_1$。然而这个式子的含义是“铁化合且氧化合，但是铁不化合且氮不化合”。显然这个句子与原句的含义完全不同。产生这种错误的原因是由于把原句中的“和”字错误地未加分析地作为真值联结词而引起的。仔细地分析一下就会明白，原句中“…和…化合”是紧密联在一起不能分解的谓语。同样“…和…是兄弟”，“…和…是朋友”等也是如此，其中的“和”字都不是真值联结词。读者必须善于判断一个句子中哪些是真值联结词，哪些不是真值联结词，从而判定出哪些是原子命题，哪些是复合命题，对复合命题还必须能分析出其中的成分命题与真值联结词。如果不熟悉这一点，即使会进行符号运算，会对符号进行各种变换，也只能知道一些与实践相脱离的理论，而不能把数理逻辑理论应用到实践中去，应用到计算机科学领域中去。

有一点应该提醒读者注意，上述五个真值联结词与日常使用

的相应词汇只是大体相当，并不完全一致．例如，在日常用语中“或”字有两种意义，其一是可兼的“或”，另一是不可兼的“或”．所谓可兼或是指“α或β”真当且仅当α真或β真或α、β均真．不可兼或是指“α或β”真当且仅当α真或β真．这就是说当α和β均真时，按可兼或的意义，“α或β”的值为真，按不可兼或的意义，“α或β”的值为假．人们有时按第一种意义来使用“或”，有时按第二种意义来使用“或”，这与说话人自己的态度以及上下文有关．在数理逻辑中，“或”只有唯一的意义，指可兼的或．日常用语中的“如果…则…”与数理逻辑中的蕴涵词也略有不同．日常用语中蕴涵前件和蕴涵后件一定是意义上有关联的两个语句，决不能把风马牛不相及的语句用“如果则”联系起来．乱联系的话蕴涵式将是无意义的，无所谓真假．但是数理逻辑中任意两个命题都可以用“如果则”联系起来，蕴涵式的真假完全由成分命题的真假确定，与成分命题的内容毫不相干．正因为此，所以我们把它叫做实质蕴涵．在今后的逻辑演算里，关于复合命题的真假一定要根据上面给出的五个真值联结词的定义去理解，而不能根据日常用语的意义去理解．而且今后我们只讨论复合命题的真假与其成分命题（进而与其原子命题）的真假之间的关系，而不讨论命题的内容含义．因此我们引入命题变元及真值函数两个概念．

定义 以真假为变域的变元称为*命题变元*．我们用p, q, r等表示命题变元．

定义 以真假为定义域且以真假为值域的函数称为*真值函数*．

由定义知，上述五个真值联结词都是真值函数，其中否定词是一元真值函数，其余四个是二元真值函数．除了这五个真值联结词外，还可定义各种真值函数．例如，如下定义的函数$f(p, q, r)$便是三元真值函数：

$f(p, q, r)$真当且仅当p, q, r均真或均假。该函数也可列表定

义如下：

p	q	r	$f(p,q,r)$
T	T	T	T
T	T	F	F
T	F	T	F
T	F	F	F
F	T	T	F
F	T	F	F
F	F	T	F
F	F	F	T

仿此可定义其它真值函数.

虽然真值函数无限多，但最重要的还是上面五个真值联结词. 在§1.3中，我们将证明任何真值函数均可用这五个真值联结词来表示. 所以，以后我们将主要讨论这五个真值联结词.

定义 由命题变元利用真值联结词作成的式子称为命题演算公式. 严格地说，命题演算公式是下列这些公式（本章中简称为公式）：

(1) 任何命题变元均是公式；

(2) 如果 α 是公式，则 $\bar{\alpha}$ 也是公式；

(3) 如果 α 和 β 是公式，则 $(\alpha \vee \beta)$，$(\alpha \wedge \beta)$，$(\alpha \supset \beta)$，$(\alpha \equiv \beta)$ 均是公式；

(4) 公式仅限于此.

我们用 α, β, γ 等表示公式. 若公式 α 中总共含有 n 个不同的命题变元，则说 α 是 n 元公式.

显然，如果把公式中的命题变元代以原子命题或复合命题，则该公式便是一个复合命题. 故对复合命题的研究可化为对公式的研究. 今后我们将以公式为研究对象，而不讨论复合命题.

习　题

1. 把下面的公式用括号法，点子法，括号点子法和前置法（使用N, K, A, C, E前置真值联结词）四种方法来表示：

1.1　$((p\supset r)\equiv\overline{(\bar{q}\supset r)})\vee((r\supset\bar{p})\wedge q)$；

1.2　$((p\wedge q)\wedge(p\wedge r))\equiv((\bar{p}\supset\bar{q})\supset\bar{r})$；

1.3　$p\cdot\equiv\cdot q\wedge r\supset\cdot\overline{p\vee s\equiv\cdot s\equiv t}$；

1.4　$\overline{p\supset q}\cdot\supset:r\vee q\wedge\cdot p\cdot\vee\cdot r\equiv p$；

1.5　$CKNApqrKNpEqr$；

1.6　$ENENENpqrs$.

2. 把下列句子表示成符号公式：

2.1　如果2小于3而5大于4，则2小于5；

2.2　如果a被d除尽，b也被d除尽，则d是a和b的公约数，但不一定是a和b的最大公约数；

2.3　d是a和b的公倍数当且仅当a和b都能除尽d；

2.4　若a为不等于2的质数，则b除以a的余数送给b，否则当a不等于0时把a除以b的余数送给a；

2.5　他将于明天或后天到苏州或上海去；

2.6　黄色与红色颜料合成橙色或棕色颜料；

2.7　张刚总是在图书馆看书，除非图书馆不开门或张刚生病；

2.8　只要大家都动手干，就能把事情做好；

2.9　只有大家都动手干，才能把事情做好；

2.10　说逻辑枯燥无味或毫无价值都是不对的.

3. 一元真值函数共有多少？二元真值函数共有多少？三元真值函数共有多少？n元真值函数共有多少？请列举出所有的一元和二元的真值函数，列举四个三元真值函数.

§1.2　真假性

设n元公式α中所含有的不同命题变元为$p_1, p_2, \cdots, p_n$. 我们把这些变元组成的变元组$(p_1, p_2, \cdots, p_n)$称为α的变元组. α的变

元组$(p_1, \cdots, p_n)$的任意一组确定的值都称为该公式α的关于该变元组$(p_1, p_2, \cdots, p_n)$的**完全指派**. 如果我们仅对变元组中部分变元赋以确定的值, 其余变元没有赋以确定的值, 则这样的一组值称为该公式α的关于该变元组$(p_1, p_2, \cdots, p_n)$的**部分指派**.

因为公式的真假仅与其中的命题变元的真假有关, 因此任给公式的一完全指派, 该公式就能取得一确定的值. 例如, 对于公式

$$\alpha = (p \wedge (q \supset r)) \wedge s$$

在完全指派$(p, q, r, s) = (T, F, F, T)$之下, α的值为真, 即$\alpha = T$. 对于部分指派的情形, 一般说来, 公式的真假与未赋以确定值的变元有关. 例如, 上式在部分指派$(p, q, r, s) = (T, F, F, \times)$之下(这里$\times$表示相应的变元未赋以确定的值. 以下均如此表示), α的值与s有关, s为真时α为真, s为假时α为假. 但是在某些特殊情况下, α的真假也可与未赋以确定值的变元无关. 例如, 上式在部分指派$(p, q, r, s) = (F, F, F, \times)$之下, $\alpha = F$.

定义 对于任一公式α, 凡使得α取真值(T)的指派, 不管是完全指派还是部分指派, 都称为α的**成真指派**. 凡使得α取假值(F)的指派, 也不管是完全指派还是部分指派, 都称为α的**成假指派**.

由这个定义立即可得:

公式$\bar{p}$的成真指派为$(p) = (F)$; 成假指派为$(p) = (T)$.

公式$p \wedge q$的成真指派为$(p, q) = (T, T)$, 成假指派为$(p, q) = (F, T), (T, F), (F, F)$.

公式$p \vee q$的成真指派为$(p, q) = (T, T), (T, F), (F, T)$, 成假指派为$(p, q) = (F, F)$.

公式$p \wedge \bar{p}$没有成真指派, 成假指派为$(p) = (T), (F)$, 即$(p) = (X)$.

公式$p \vee \bar{p}$没有成假指派, 成真指派为$(p) = (T), (F)$, 即(p)

$=(\times)$.

任给一公式α. 设α中含有n个命题变元$p_1, \cdots, p_n$. 显然，部分指派$(v_1, \cdots, v_{i-1}, \times, v_{i+1}, \cdots, v_n)$为$\alpha$的成真指派当且仅当$(v_1, \cdots, v_{i-1}, T, v_{i+1}, \cdots, v_n)$和$(v_1, \cdots, v_{i-1}, F, v_{i+1}, \cdots, v_n)$均为成真指派. $(v_1, \cdots, v_{i-1}, \times, v_{i+1}, \cdots, v_n)$为$\alpha$的成假指派当且仅当$(v_1, \cdots, v_{i-1}, T, v_{i+1}, \cdots, v_n)$和$(v_1, \cdots, v_{i-1}, F, v_{i+1}, \cdots, v_n)$均为成假指派. (其中诸$v_k$或为$T$或为$F$, $1 \leqslant k \leqslant n$且$k \neq i$)

定义 如果一个公式的所有完全指派均为成真指派，则该公式称为*永真公式*，或称为*重言式*.

如果一个公式有成真指派，则该公式称为*可满足公式*.

如果一个公式的所有完全指派均为成假指派，则该公式称为*永假公式*，或称为*不可满足公式*，或称为*矛盾式*.

如果一个公式有成假指派，则该公式称为*非永真公式*.

由这个定义可知，$p \wedge \bar{p}$为永假公式，当然也为非永真公式；$p \vee \bar{p}$为永真公式，当然也为可满足公式；$\bar{p}$，$p \wedge q$，$p \vee q$既为可满足公式，又为非永真公式.

由定义易证下面的定理.

定理 α永真当且仅当$\bar{\alpha}$不可满足；α可满足当且仅当$\bar{\alpha}$非永真.

上面是就一个公式进行讨论的. 现在我们讨论任意两个公式之间的关系.

定义 设有两个公式α, β. 如果关于两者的合成变元组（即这两个公式合在一起时的变元组）的任何完全指派，α和β永取相同的真假值，则α和β叫做*同真假*，或称为*逻辑等价*，记为$\alpha = \beta$.

如果关于两者的合成变元组的任何完全指派，α和β永取不同的真假值，则α和β叫做*互相矛盾*或称为*恒不同真假*.

由定义易证下面的定理.

定理 α 和 β 同真假当且仅当 $\alpha \equiv \beta$ 为永真公式.

α 和 β 互相矛盾当且仅当 α 和 $\bar{\beta}$ 同真假.

α 和 β 互相矛盾当且仅当 $\bar{\alpha}$ 和 β 同真假.

定理 设 α 为 γ 的子公式，又设在 γ 中用 β 替换 α 后所得的公式为 δ. 如果 α 和 β 同真假，则 γ 和 δ 也同真假.

[证] 设 γ 和 δ 的合成变元组为 $(p_1, \cdots, p_n)$.

任给该变元组的一个完全指派 $(p_1, \cdots, p_n) = (v_1, \cdots, v_n)$，其中诸 v_i 或为 T 或为 F, $1 \leqslant i \leqslant n$. 因为 α 和 β 同真假，所以在该指派下，α 和 β 取相同的值，设为 v，即在该指派下，$\alpha = \beta = v$，记为

(1) $\quad \alpha(p_1, \cdots, p_n)|_{(v_1, \cdots, v_n)} = \beta(p_1, \cdots, p_n)|_{(v_1, \cdots, v_n)} = v$

现把 γ 中的 α 和 δ 中的 β 看作两个新的命题变元，所得的公式分别为 γ_1 和 δ_1. 这样 γ_1 和 δ_1 的变元组分别为 $(p_1 \cdots, p_n, \alpha)$ 和 $(p_1, \cdots, p_n, \beta)$ 由(1)可得

(2) $\quad \gamma(p_1, \cdots, p_n)|_{(v_1, \cdots, v_n)} = \gamma_1(p_1, \cdots, p_n, \alpha)|_{(v_1, \cdots, v_n, v)}$

(3) $\quad \delta(p_1, \cdots, p_n)|_{(v_1, \cdots, v_n)} = \delta_1(p_1, \cdots, p_n, \beta)|_{(v_1, \cdots, v_n, v)}$

因为 δ 是把 γ 中的子公式 α 替换成子公式 β 的结果，所以 δ_1 是把 γ_1 中的命题变元 α 替换成 β 的结果. 这样当把 γ_1 中的 α 指派以 v 把 δ_1 中的 β 也指派以 v 时，γ_1 和 δ_1 便完全相同了，即有

$$\gamma_1(p_1, \cdots, p_n, \alpha)|_{(x, \cdots, x, v)} = \delta_1(p_1, \cdots, p_n, \beta)|_{(x, \cdots, x, v)}$$

特别有

(4) $\quad \gamma_1(p_1, \cdots, p_n, \alpha)|_{(v_1, \cdots, v_n, v)} = \delta_1(p_1, \cdots, p_n, \beta)|_{(v_1, \cdots, v_n, v)}$

由(2), (3), (4)得

$$\gamma(p_1, \cdots, p_n)|_{(v_1, \cdots, v_n)} = \delta(p_1, \cdots, p_n)|_{(v_1, \cdots, v_n)}$$

由定义知，γ 和 δ 同真假. 本定理得证.

下面是一些简单而又重要的同真假式，请读者验证并熟记之.

第一组 交换律

$$p \vee q = q \vee p$$

$$p \wedge q = q \wedge p$$

$$p \equiv q = q \equiv p$$

第二组　结合律

$$(p \vee q) \vee r = p \vee (q \vee r)$$

$$(p \wedge q) \wedge r = p \wedge (q \wedge r)$$

$$(p \equiv q) \equiv r = p \equiv (q \equiv r)$$

第三组　分配律

$$p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$$

$$p \wedge (q \wedge r) = (p \wedge q) \wedge (p \wedge r)$$

$$p \vee (q \wedge r) = (p \vee q) \wedge (p \vee r)$$

$$p \vee (q \vee r) = (p \vee q) \vee (p \vee r)$$

$$p \supset (q \supset r) = (p \supset q) \supset (p \supset r)$$

第四组　否定深入

$$\overline{\overline{p}} = p$$

$$\overline{p \wedge q} = \bar{p} \vee \bar{q}$$

$$\overline{p \vee q} = \bar{p} \wedge \bar{q}$$

$$\overline{p \supset q} = p \wedge \bar{q}$$

$$\overline{p \equiv q} = \bar{p} \equiv q = p \equiv \bar{q}$$

第五组　变目等同

$$p \wedge p = p \qquad p \vee p = p$$

$$p \wedge \bar{p} = F \qquad p \vee \bar{p} = T$$

$$p \supset p = T \qquad p \supset \bar{p} = \bar{p}$$

$$\bar{p} \supset p = p \qquad p \equiv p = T$$

$$p \equiv \bar{p} = \bar{p} \equiv p = F$$

第六组　部分指派

$$T \wedge p = p \qquad T \vee p = T$$

$$F \wedge p = F \qquad F \vee p = p$$

$$T \supset p = p \qquad p \supset T = T$$
$$F \supset p = T \qquad p \supset F = \bar{p}$$
$$T \equiv p = p \qquad F \equiv p = \bar{p}$$

第七组　联结词化归

$$p \wedge q = \overline{\bar{p} \vee \bar{q}}$$
$$p \vee q = \overline{\bar{p} \wedge \bar{q}}$$
$$p \supset q = \bar{p} \vee q$$
$$p \equiv q = (p \supset q) \wedge (q \supset p)$$
$$= (\bar{p} \vee q) \wedge (\bar{q} \vee p)$$
$$= (p \wedge q) \vee (\bar{p} \wedge \bar{q})$$

两个公式之间除了上面讨论的同真假性和互相矛盾性这两种关系外，还有许多其它的关系，其中比较值得研究的是同永真性和同可满足性.

定义　如果 α 永真当且仅当 β 永真，则说 α 和 β 同永真性.

如果 α 可满足当且仅当 β 可满足，则说 α 和 β 同可满足性.

因为 p 和 q 均为非永真公式，故由定义知，p 和 q 同永真性. 又因为 p 和 q 均为可满足公式，故由定义知，p 和 q 同可满足性.

容易证明下面的定理.

定理　如果 α 和 β 同真假，则 α 和 β 既同永真性又同可满足性.

请读者注意，该定理的逆定理并不成立.

定理　α 和 β 同永真性当且仅当 $\bar{\alpha}$ 和 $\bar{\beta}$ 同可满足性；α 和 β 同可满足性当且仅当 $\bar{\alpha}$ 和 $\bar{\beta}$ 同永真性.

[证]　设 α 和 β 同永真性. 由定义知，α 和 β 或者均永真或者均非永真. 若 α 和 β 均永真，则 $\bar{\alpha}$ 和 $\bar{\beta}$ 均永假，即 $\bar{\alpha}$ 和 $\bar{\beta}$ 均不可满足. 若 α 和 β 均非永真，则 $\bar{\alpha}$ 和 $\bar{\beta}$ 均可满足. 由定义知，$\bar{\alpha}$ 和 $\bar{\beta}$ 同可满足性.

设 $\bar{\alpha}$ 和 $\bar{\beta}$ 同可满足性. 由定义知, $\bar{\alpha}$ 和 $\bar{\beta}$ 或者均可满足或者均不可满足. 若$\bar{\alpha}$和 $\bar{\beta}$ 均可满足, 则 α 和 β 均非永真. 若 $\bar{\alpha}$ 和 $\bar{\beta}$ 均不可满足, 则 α 和 β 均永真. 由定义知, α 和 β 同永真性.

本定理的前一半得证. 同法可证得后一半.

定义 把任意一个不含蕴涵词和等价词的 α 中的所有 $\wedge$ 换为 $\vee$, 所有的 $\vee$ 换为 $\wedge$ 后所得的公式称为该 α 的对偶式, 记为 α^*.

例如, $(\overline{p \vee q} \wedge \overline{p \vee r}) \vee (p \wedge (\bar{q} \vee \bar{r}))$的对偶式为

$$(\overline{p \wedge q} \vee \overline{p \wedge r}) \wedge (p \vee (\bar{q} \wedge \bar{r}))$$

由定义易知, $(\alpha^*)^*$仍为 α. 因此, α 和 α^* 互为对偶式.

定义 把任一公式 α 中各变元的所有肯定形式的出现换为其否定, 所有否定形式的出现换为其肯定后所得的式子称为该 α 的内否式, 记为 α^-.

例如, $(\overline{p \vee q} \wedge \overline{p \vee r}) \vee (p \wedge (\bar{q} \vee \bar{r}))$的内否式为

$$(\overline{\bar{p} \vee \bar{q}} \wedge \overline{\bar{p} \vee \bar{r}}) \vee (\bar{p} \wedge (q \vee r))$$

由定义易知, $(\alpha^-)^-$ 仍为 α. 因此, α 和 α^-是互为内否式.

下面我们简单讨论对偶式和内否式的某些性质. 讨论这些性质时, 凡涉及到对偶式, 均假定公式中不含 $\supset$ 和 $\equiv$.

定理 对于任何公式α, 均有$\overline{(\alpha^*)} = (\bar{\alpha})^*$, $\overline{(\alpha^-)} = (\bar{\alpha})^-$请读者自行证明.

定理 对于任何公式 α, 均有 $\bar{\alpha} = \alpha^{*-}$.

[证] 现就 α 中的真值联结词的个数 n 施行归纳.

奠基 应证当 $n=0$ 时本定理成立.

因为 $n=0$ 时, α 必为命题变元, 设 $\alpha = p$.

所以 $$\bar{\alpha} = \bar{p}$$

而 $$\alpha^{*-} = p^{*-} = \bar{p}$$

故 $$\bar{\alpha} = \alpha^{*-}$$

归纳 应证若 $n \leqslant k$ 时本定理成立, 则 $n = k+1$ 时本定理也成

立.

因为 $n=k+1\geqslant 1$, 所以 α 中至少有一个真值联结词, 故 α 必为下列三种形式之一:

$$\alpha=\bar{\alpha}_1 \quad 或 \quad \alpha=\alpha_1\wedge\alpha_2 \quad 或 \quad \alpha=\alpha_1\vee\alpha_2$$

且 α_1 和 α_2 中的真值联结词的个数均小于等于 n. 根据归纳假设有

$$\bar{\alpha}_1=\alpha_1{}^{*^-},\quad \bar{\alpha}_2=\alpha_2{}^{*^-}$$

因此, 当 $\alpha=\bar{\alpha}_1$ 时

$$\bar{\alpha}=\overline{\bar{\alpha}_1}=\overline{(\alpha_1{}^{*^-})}=(\bar{\alpha}_1)^{*^-}=\alpha^{*^-}$$

本定理成立.

当 α 为 $\alpha_1\wedge\alpha_2$ 时

$$\begin{aligned}\bar{\alpha}=\overline{\alpha_1\wedge\alpha_2}&=\bar{\alpha}_1\vee\bar{\alpha}_2\\&=\alpha_1{}^{*^-}\vee\alpha_2{}^{*^-} && [归纳假设]\\&=(\alpha_1{}^*\vee\alpha_2{}^*)^- && [内否式的定义]\\&=(\alpha_1\wedge\alpha_2)^{*^-} && [对偶式的定义]\\&=\alpha^{*^-}\end{aligned}$$

本定理亦成立.

同法可证得当 $\alpha=\alpha_1\vee\alpha_2$ 时本定理也成立.

由归纳法, 本定理成立.

定理 α 与 α^- 既同永真性又同可满足性.

[证] 设 α 与 α^- 的变元组为 $(p_1,\cdots,p_n)$. 任给一个指派 $(p_1,\cdots,p_n)=(v_1,\cdots,v_n)$, 由内否式的定义易知

$$\alpha(p_1,\cdots,p_n)\big|_{(v_1,\cdots,v_n)}=\alpha^-(p_1,\cdots,p_n)\big|_{(\bar{v}_1,\cdots,\bar{v}_n)}$$

$$\alpha^-(p_1,\cdots,p_n)\big|_{(v_1,\cdots,v_n)}=\alpha(p_1,\cdots,p_n)\big|_{(\bar{v}_1,\cdots,\bar{v}_n)}$$

由此不难证得本定理.

定理 $\bar{\alpha}$ 与 α^* 既同永真性又同可满足性.

[证] 因为 $\bar{\alpha}$ 与 α^{*^-} 同真假, 又因为 $(\alpha^*)^-$ 与 (α^*) 既同永真性

又同可满足性，所以 $\bar{\alpha}$ 与 α^* 既同永真性又同可满足性.

对偶定理 任给公式 α 和 β，均有

$\alpha\supset\beta$ 与 $\beta^*\supset\alpha^*$ 既同永真性又同可满足性；

$\alpha\equiv\beta$ 与 $\alpha^*\equiv\beta^*$ 既同永真性又同可满足性.

[证] 因为

$$\begin{aligned}\alpha\supset\beta&=\bar{\beta}\supset\bar{\alpha}\\&=\beta^{*-}\supset\alpha^{*-}\\&=(\beta^*\supset\alpha^*)^-\end{aligned}$$

又因为 $(\beta^*\supset\alpha^*)^-$ 与 $(\beta^*\supset\alpha^*)$ 同永真性同可满足性，所以 $(\alpha\supset\beta)$ 与 $(\beta^*\supset\alpha^*)$ 同永真性同可满足性.

因为

$$\begin{aligned}\alpha\equiv\beta&=\bar{\alpha}\equiv\bar{\beta}\\&=\alpha^{*-}\equiv\beta^{*-}\\&=(\alpha^*\equiv\beta^*)^-\end{aligned}$$

所以 $(\alpha\equiv\beta)$ 与 $(\alpha^*\equiv\beta^*)$ 同永真性同可满足性.

现在我们来讨论这样一个问题：任给一个公式 α，如何确定它的成真指派和成假指派？解决这个问题的最简单的方法是把 α 的所有完全指派逐个代入 α 中，计算出 α 在各个指派之下所取的值，这样便能得出 α 的一切成真和成假指派. 这个方法虽然很简单，但是因为完全指派的总数相当多，按指数级数增长，而且对每个指派所需作的计算也相当长，所以实际上一般不用此法而用其它更为简单的方法. 下面我们介绍一种较为简便的方法——部分指派法.

部分指派法可综述如下：

第一步 否定深入.

先把否定词一直深入到变元上.

第二步 部分指派.

选定一个变元，对其作真和假两种指派，得到两个不含该变元但含真假值的公式.

第三步　化简.

根据上面关于部分指派的同真假式作变换，必能把第二步中得到的公式化简成变元较少的公式. 如果化简后的公式中还含有两个或两个以上的变元，则对这些化简了的公式再重新回到第二步，重复上列过程，否则化简后的公式只含有一个变元或者不含有变元，这时真假指派将是非常明显的，可以得到结果.

在第二步中，每次选定哪个变元对其作指派呢？大体说来可用下列准则.

1. 每次都应对出现次数最多的变元作指派.

2. 如果对某变元指派以"T"或"F"时，立即能得到结果的，则可先对该变元作指派.

当这两个准则彼此冲突时，可任用其一. 下面给出一个例子来说明.

例：试判定

$$\overline{(p \vee r) \supset (p \supset q) \equiv p \wedge q \equiv r}$$

的永真性和可满足性.

[解]　第一步：否定深入的结果为

$$(p \vee r) \supset ((p \supset q) \equiv (p \wedge (q \equiv r)))$$

第二、三步：因为 p 出现次数最多，所以先对 p 作指派，并化简.

$p=T$ 时得　$(T \vee r) \supset ((T \supset q) \equiv (T \wedge (q \equiv r)))$

$= T \supset (q \equiv (q \equiv r))$

$= q \equiv (q \equiv r)$

这个式子无需否定深入，只需对 q 或 r 指派. 因为对 r 作指派能立即得结果，故对 r 作指派.

$r=T$ 时得　$q \equiv (q \equiv \bar{T}) = q \equiv \bar{q} = F$

$r=F$ 时得　$q \equiv (q \equiv \bar{F}) = q \equiv q = T$

上面是 p 为真时的结果. 再对 p 指派以假后化简.

$p=F$ 时得 $(F\vee r)\supset((F\supset q)\equiv(F\wedge(p\equiv r)))$

$=r\supset(T\equiv F)$

$=r\supset F$

$=r$

综上所述, 可得原公式的成真指派是

$$(p,q,r)=(T,\times,F),(F,\times,T)$$

成假指派是

$$(p,q,r)=(T,\times,T),(F,\times,F)$$

所以原公式是非永真但可满足.

习　题

1. 验证上面的七组同真假式.

2. 作出下列公式的否定式, 并把它们化为中置式, 再把否定深入到变元之前:

2.1 $CNANpqNCNKNqrApNr$;

2.2 $CNKANKprqNKqNEpqNCqNEpr$.

3. 利用上面七组同真假式验证下列同真假式:

3.1 $p\equiv(p\equiv q)=q$;

3.2 $p\equiv(\bar{p}\equiv q)=\bar{q}$;

3.3 $p\wedge(p\equiv q)=p\wedge q$;

3.4 $p\wedge(\bar{p}\equiv q)=p\wedge\bar{q}$;

3.5 $p\vee(p\equiv q)=p\vee\bar{q}$;

3.6 $p\vee(\bar{p}\equiv q)=p\vee q$.

4. 利用等价变换证明下列各式永真:

4.1 $p\supset(q\supset p)$;

4.2 $p\supset(\bar{p}\supset q)$;

4.3 $((p\supset q)\supset p)\supset p$;

4.4 $((p\supset q)\supset q)\equiv((q\supset p)\supset p)$;

4.5 $p \vee (p \supset q)$;

4.6 $(p \supset q) \vee (q \supset p)$.

5. n 元公式 α 的完全指派共有多少？仅对 m 个变元 $(m<n)$ 指派以真假的部分指派共有多少？

6. 试证 α 和 β 互相矛盾当且仅当 $\overline{\alpha \equiv \beta}$ 为永真公式.

7. 写出下列各式的对偶式和内否式:

7.1 $(\overline{p \supset q} \vee \overline{q \wedge r}) \supset (p \vee \bar{r})$;

7.2 $(p \equiv q) \supset (\bar{p} \wedge (\overline{q \supset r}))$.

8. 试证若 α 与 β 同永真性，则 α 与 $\alpha \wedge \beta$ 同永真性.

9. 求下列公式的成真指派与成假指派:

9.1 $(p \equiv q) \vee ((q \wedge r) \supset p)$;

9.2 $\overline{p \wedge \bar{q}} \supset ((q \equiv p) \equiv r)$;

9.3 $\overline{p \wedge q} \wedge (r \supset \bar{p})$;

9.4 $((p \supset r) \wedge ((q \supset r) \wedge \cdot r)) \equiv (\bar{p} \vee \bar{q})$.

10. 某单位要派人去 A 地学习，但因工作关系甲、乙两人不能都去 A 地学习，且若派乙去则丙要留下工作，若派丁去则乙和丙至少要去一人. 试问甲、乙、丙、丁四人中最多能派几个人去 A 地学习？若派两人去 A 地学习的话，可派哪两个人去？

§1.3 范式和应用

上一节中我们讨论了求一个公式 α 的真假指派问题，现在来讨论该问题的逆问题：已知一公式 α 的成真和成假指派，能否把 α 的表达式求出来？

为此，我们先来研究真假指派与真值联结词之间的关系.

定义 命题变元或者命题变元的否定或者由它们利用合取词“$\wedge$”组成的公式称为*简单合取式*；命题变元或者命题变元的否定或者由它们利用析取词“$\vee$”组成的公式称为*简单析取式*.

由定义知，p, $\bar{q}$, $p \wedge \bar{q} \wedge q$, $p \wedge \bar{p} \wedge \bar{q} \wedge r \wedge p \wedge \bar{q}$ 等均为简单合取式；p, $\bar{q}$, $p \vee \bar{q} \vee q$, $\bar{p} \vee p \vee r \vee q \vee p$ 等均为简单析取式.

读者易见，根据 $p \wedge p = p$, $p \vee p = p$ 以及交换律、结合律等同真假式，恒可把相同的合取项(析取项)合并成一项. 例如:

$$p \wedge p \wedge q \wedge r \wedge p \wedge q \quad 及 \quad p \vee p \vee q \vee r \vee p \vee q$$

可分别合并成

$$p \wedge q \wedge r \quad 及 \quad p \vee q \vee r.$$

因此今后我们恒可假定，在简单合取式中各个合取项都不相同，在简单析取式中各个析取项都不相同.

读者还易知，根据 $p \wedge \bar{p} = F$, $p \vee \bar{p} = T$, 可证明某一变元及其否定均出现的简单合取式必为永假公式，例如 $\bar{p} \wedge q \wedge p$ 为永假公式；而某一变元及其否定均出现的简单析取式必为永真公式. 例如，$p \vee q \vee \bar{q}$ 为永真公式.

定义 某一变元及其否定均出现的简单合取式称为*虚合取式*，或称为*永假合取式*，反之每个变元最多出现一次(或者肯定或者否定)的简单合取式称为*实合取式*.

某一变元及其否定均出现的简单析取式称为*虚析取式*，或称为*永真析取式*，反之每个变元最多出现一次(或者肯定或者否定)的简单析取式称为*实析取式*.

由此可得下面的定理.

定理 虚合取式没有成真指派，一切虚合取式均同真假；虚析取式没有成假指派，一切虚析取式均同真假.

定理 就一确定的变元组而言，任一实合取式有也仅有一个成真指派，任一实析取式有也仅有一个成假指派；反之，任给一个指派，除$(\times, \cdots, \times)$外，有一个也仅有一个实合取式以该指派为其成真指派，有一个也仅有一个实析取式以该指派为其成假指派.

现举例说明:

$p \wedge q \wedge r \wedge \bar{s}$ 的唯一成真指派为 $(p, q, r, s) = (T, F, T, F)$.

$\bar{p} \wedge r \wedge \bar{s}$ 的唯一成真指派为 $(p, q, r, s) = (F, \times, F, F)$.

$p \vee \bar{q} \vee r \vee s$ 的唯一成假指派为 $(p, q, r, s) = (F, T, F, F)$.

$p \vee r \vee s$ 的唯一成假指派为 $(p, q, r, s) = (F, \times, T, T)$.

反之

$(p, q, r) = (F, F, T)$ 对应的实合取式为 $\bar{p} \wedge \bar{q} \wedge r$,
对应的实析取式为 $p \vee q \vee \bar{r}$.

$(p, q, r) = (T, F, \times)$ 对应的实合取式为 $p \wedge \bar{q}$,
对应的实析取式为 $\bar{p} \vee q$.

由此读者不难得出建立这种对应的方法.

以后我们将使用"与某某(成真)指派对应的简单合取式","与某某(成假)指派对应的简单析取式"这类说法,其含义是很显然的.

下列定理是容易验证的.

定理 合取式 $\alpha_0 \wedge \alpha_1 \wedge \cdots \wedge \alpha_n$ 的成真指派集为 α_0, $\alpha_1, \cdots, \alpha_n$ 的成真指派集的交集,而其成假指派集为 $\alpha_0, \alpha_1, \cdots, \alpha_n$ 的成假指派集的并集.

析取式 $\alpha_0 \vee \alpha_1 \vee \cdots \vee \alpha_n$ 的成假指派集为 α_0, $\alpha_1, \cdots, \alpha_n$ 的成假指派集的交集,而其成真指派集为 α_0, $\alpha_1, \cdots, \alpha_n$ 的成真指派集的并集.

现在我们可以回答本节开始时所提出的问题了.它可归纳为下述的基本定理.

基本定理 任一公式 α 恒可以表为简单合取式或简单合取式的析取(称为 α 的*析合范式*),也可以表为简单析取式或简单析取式的合取(称为 α 的*合析范式*).

[证] 如果 α 为永真公式,则 $\alpha = p \vee \bar{p}$. 因为 $p \vee \bar{p}$ 既是析合范式又是合析范式,故定理成立.

如果 α 为永假公式,则 $\alpha = p \wedge \bar{p}$. 因为 $p \wedge \bar{p}$ 既是析合范式又是合析范式,故定理成立.

如果α既非永真也非永假，则α既有成真指派又有成假指派．设α的成真指派为$\xi_0, \cdots, \xi_k$；α的成假指派为$\eta_0, \cdots, \eta_k$．作出与这些成真指派相对应的简单合取式，设为$\alpha_0, \cdots, \alpha_k$；又作出与这些成假指派相对应的简单析取式，设为$\beta_0, \cdots, \beta_k$．由上面的定理知，下面的析合范式

$$\alpha_0 \vee \alpha_1 \vee \cdots \vee \alpha_k$$

的成真指派恰为$\xi_0, \xi_1, \cdots, \xi_k$；而下面的合析范式

$$\beta_0 \wedge \beta_1 \wedge \cdots \wedge \beta_k$$

的成假指派恰为$\eta_0, \eta_1, \cdots, \eta_k$．故有

$$\alpha = \alpha_0 \vee \alpha_1 \vee \cdots \vee \alpha_k = \beta_0 \wedge \beta_1 \wedge \cdots \wedge \beta_k$$

于是定理得证．

现举一例来说明如何具体求析合范式和合析范式．

例：设公式α的成真指派为

$$(p, q, r, s) = (T, \times, F, T), (F, T, \times, F), (T, \times, \times, F)$$

求其析合范式和合析范式．

[解] 先求析合范式．

作出相应于各成真指派的简单合取式，它们分别为

$$p \wedge r \wedge s, \quad \bar{p} \wedge q \wedge \bar{s}, \quad p \wedge \bar{s}$$

把它们析取起来则得析合范式

$$(p \wedge r \wedge s) \vee (\bar{p} \wedge q \wedge \bar{s}) \vee (p \wedge \bar{s})$$

再求合析范式．

应先根据成真指派列出所有的成假指派，它们为

$$(T, T, T, T), (T, F, T, T), (F, F, T, F), (F, F, F, F),$$
$$(F, T, T, T), (F, T, F, T), (F, F, T, T), (F, F, F, T).$$

作出诸成假指派相应的简单析取式，它们为

$$\bar{p} \vee \bar{q} \vee r \vee \bar{s}, \quad \bar{p} \vee q \vee r \vee \bar{s}, \quad p \vee q \vee r \vee s, \quad p \vee q \vee r \vee \bar{s},$$
$$p \vee \bar{q} \vee r \vee \bar{s}, \quad p \vee \bar{q} \vee r \vee \bar{s}, \quad p \vee q \vee r \vee \bar{s}, \quad p \vee q \vee r \vee \bar{s}.$$

把它们合取起来则得合析范式

$$(\bar{p}\vee\bar{q}\vee\bar{r}\vee\bar{s})\wedge(\bar{p}\vee q\vee\bar{r}\vee\bar{s})$$
$$\wedge(p\vee q\vee\bar{r}\vee s)\wedge(p\vee q\vee r\vee s)$$
$$\wedge(p\vee\bar{q}\vee r\vee\bar{s})\wedge(p\vee\bar{q}\vee r\vee\bar{s})$$
$$\wedge(p\vee q\vee\bar{r}\vee\bar{s})\wedge(p\vee q\vee r\vee\bar{s}).$$

由这个基本定理可以得到下面的重要推论.

推论 任何真值函数均可用"$^-$", "$\wedge$", "$\vee$"三个真值联结词作出.

[**证**] 设 f 是任一 n 元真值函数. 由命题变元

$p_1, \cdots, p_n$ 利用 f 组成公式 $f(p_1, \cdots, p_n)$.

由基本定理知, 该公式可表为析合范式或合析范式, 即该公式可由命题变元 $p_1, \cdots, p_n$ 利用"$^-$", "$\wedge$", "$\vee$"三个真值联结词来表示. 故任一真值函数均可用"$^-$", "$\wedge$", "$\vee$"三个真值联结词作出.

推论 一切真值函数均可用 $^-$, $\wedge$, $\vee$, $\supset$ 和 $\equiv$ 五个真值联结词作出.

推论 一切真值函数均可用 $^-$, $\wedge$ 或者 $^-$, $\vee$ 或者 $^-$, $\supset$ 这三组联结词中任何一组作出.

[**证**] 现就 $^-$, $\wedge$ 证明之, 其它两组请读者自证.

因为 $$\alpha\vee\beta=\overline{\bar{\alpha}\wedge\bar{\beta}}$$

所以"$\vee$"可用 $^-$, $\wedge$ 表示. 故由上面的推论得知, 一切真值函数均可用 $^-$, $\wedge$ 来表示.

现在我们来定义一个新的真值联结词"与非词", 记为"$\overline{\wedge}$", 其定义如下:

p	q	$p\overline{\wedge}q$
T	T	F
T	F	T
F	T	T
F	F	T

可以证明一切真值函数均可用该联结词作出.

推论 一切真值函数均可用"$\overline{\wedge}$"作出.

[证] 因为 $$\bar{p}=p\overline{\wedge}p$$

$$p\wedge q=\overline{p\overline{\wedge}q}=(p\overline{\wedge}q)\overline{\wedge}(p\overline{\wedge}q)$$

所以 $\bar{}$ 和 $\wedge$ 均可由 $\overline{\wedge}$ 作出. 由上面的推论可知本推论成立.

上面讨论的按照成真指派或成假指派作出析合范式或合析范式的方法,在自动交换机、数字计算机以及各种自动控制系统的逻辑设计中有着重要的应用. 现作一点简单的介绍.

首先应该注意的是, 上面讨论的"真"和"假"两个概念是没有给以定义的概念. 尤其是我们使用的符号"T"和"F", 它们仅仅是两个不同的符号而已,用其它一对符号来代替,对我们的讨论没有任何影响. 例如,用"+"和"−"这一对符号来代替是完全可以的. 现在我们不妨以"0"和"1"来表示"真"和"假". 这样我们讨论的范围便限于 0 与 1 了. 通常把这样讨论 0 和 1 的命题逻辑系统称为布尔代数,又称为两值系统. 在布尔代数中,我们有下面的规则:

$$\bar{0}=1;\quad \bar{1}=0;$$

$$0\wedge 0=0;\quad 0\wedge 1=1;\quad 1\wedge 0=1;\quad 1\wedge 1=1;$$

$$0\vee 0=0;\quad 0\vee 1=0;\quad 1\vee 0=0;\quad 1\vee 1=1;$$

$$0\supset 0=0;\quad 0\supset 1=1;\quad 1\supset 0=0;\quad 1\supset 1=0;$$

$$0\equiv 0=0;\quad 0\equiv 1=1;\quad 1\equiv 0=1;\quad 1\equiv 1=0.$$

在布尔代数中相应于命题演算中的真值函数就是以 0、1 为定义域以 0、1 为值域的函数, 称之为布尔函数; 而相应于命题变元就是以 0 和 1 为变域的变元, 称之为布尔变元; 相应于命题演算公式就是由布尔变元利用布尔函数作成的式子,称之为布尔代数公式. 由上面的基本定理及其推论可知, 任何布尔代数公式以及任何布尔函数均可用 $\bar{}$、$\wedge$、$\vee$ 三个函数来表示.

设有一数论函数 f,它把自然数 a 和 b 变为自然数 c,即

$$c=f(a,b)$$

因为任何自然数均可用二进制数表示，故可将 a、b、c 表成二进制数，设它们为

$$a=a_ka_{k-1}\cdots a_1$$

$$b=b_kb_{k-1}\cdots b_1$$

$$c=c_tc_{t-1}\cdots c_1$$

其中诸 $a_i(1\leqslant i\leqslant k)$，诸 $b_j(1\leqslant j\leqslant k)$、诸 $c_t(1\leqslant t\leqslant l)$均为 0 或 1. 因为 c 由 a 和 b 而得，所以各 c_t 应由诸 a_i 和诸 b_j 的值而得，即应有

$$c_t=f_t(a_1,\cdots,a_k;b_1,\cdots,b_k),\qquad t=1,\cdots,l$$

这里各 f_t 均为以 0 和 1 为定义域以 0 和 1 为值域的布尔函数，因此均可用 $\bar{}$、$\wedge$、$\vee$ 来表示. 由此可见，任何对数论函数的研究均可化归为对 $\bar{}$，$\wedge$，$\vee$ 这三个函数的研究.

这里举一个具体例子来说明. 设数论函数 f 为加法，即

$$c=a+b$$

用二进制表示即为

$$\begin{array}{cccccccc}
 & a_n & a_{n-1} & \cdots & a_i & \cdots & a_1 \\
 & b_n & b_{n-1} & \cdots & b_i & \cdots & b_1 \\
e_{n+1} & e_n & e_{n-1} & \cdots & e_i & \cdots & e_1 \\
\hline
c_{n+1} & c_n & c_{n-1} & & c_i & & c_1 \\
 & e_{n+1} & e_n & & e_{i+1} & & e_2
\end{array}$$

这里 e_1 表示 0，e_{i+1} 表示第 i 位的进位，有进位时 $e_{i+1}=1$，无进位时 $e_{i+1}=0$，$i=1,\cdots,n$. 根据下表便可由 a_i、b_i、e_i 算出 c_i 及 e_{i+1} $(i=1,\cdots,n)$：

a_i	0	0	0	0	1	1	1	1
b_i	0	0	1	1	0	0	1	1
e_i	0	1	0	1	0	1	0	1
c_i	0	1	1	0	1	0	0	1
e_{i+1}	0	0	0	1	0	1	1	1

再根据其成真指派可得到下面的析合范式

$$c_i=(\bar{a}_i\wedge\bar{b}_i\wedge e_i)\vee(\bar{a}_i\wedge b_i\wedge\bar{e}_i)$$
$$\vee(a_i\wedge\bar{b}_i\wedge\bar{e}_i)\vee(a_i\wedge b_i\wedge e_i)$$
$$e_{i+1}=(\bar{a}_i\wedge b_i\wedge e_i)\vee(a_i\wedge\bar{b}_i\wedge e_i)$$
$$\vee(a_i\wedge b_i\wedge\bar{e}_i)\vee(a_i\wedge b_i\wedge e_i)$$

最后补以

$$e_1=0$$
$$c_{n+1}=e_{n+1}$$

这样问题便解决了.

如用电子元件来实现 $\bar{}$, $\wedge$, $\vee$ 三个函数，便能根据这些析合范式制造出由 a_i, b_i 和 e_i 来计算 c_i 和 e_{i+1} 的机器，这种机器称为加法器. 同样可以造出减法器，乘法器，除法器等等. 这就是设计数字计算机的原理.

利用析合范式或合析范式可设计一些控制系统，现举一例来说明. 设一房间有四扇门，要求设计一个电灯控制线路，使得每个门旁的开关均能开关该房内的照明电灯. 我们把四扇门旁的开关分别记为 K_1, K_2, K_3 和 K_4，"1"表示开关断开，"0"表示开关接通. L 表示室内的照明电灯，"1"表示熄，"0"表示亮. 又设开始时四个开关均断开，电灯不亮，即 $K_1=K_2=K_3=K_4=1$, $L=1$. 有人到房间来时，随便走哪个门，改变一个门旁开关的状态(注意，这是第一次改变开关状态)，电灯均应亮. 当人离开房间时，随便走哪个门改变一下门旁开关的状态(这是第二次改变开关状态)，电灯应熄. 接着第三次改变开关状态时电灯应亮，第四次改变开关状态时电灯应熄，如此继续下去. 不难看出，偶数次改变开关状态时电灯应熄，奇数次改变开关状态时电灯应亮. 如此可以列出下表:

K_1	1	0	1	1	1	1	1	1	0	0	0	1	0	0	0	0
K_2	1	1	0	1	1	1	0	0	1	1	0	0	1	0	0	0
K_3	1	1	1	0	1	0	1	0	1	0	1	0	0	1	0	0
K_4	1	1	1	1	0	0	0	1	0	1	1	0	0	0	1	0
L	1	0	0	0	0	1	1	1	1	1	1	0	0	0	0	1

再根据L为0的所有情形可得到下面的析合范式

$$\begin{aligned}L=&(K_1\wedge\overline{K}_2\wedge\overline{K}_3\wedge\overline{K}_4)\vee(\overline{K}_1\wedge K_2\wedge\overline{K}_3\wedge\overline{K}_4)\\&\vee(\overline{K}_1\wedge\overline{K}_2\wedge K_3\wedge\overline{K}_4)\vee(\overline{K}_1\wedge\overline{K}_2\wedge\overline{K}_3\wedge K_4)\\&\vee(K_1\wedge K_2\wedge K_3\wedge\overline{K}_4)\vee(K_1\wedge K_2\wedge\overline{K}_3\wedge K_4)\\&\vee(K_1\wedge\overline{K}_2\wedge K_3\wedge K_4)\vee(\overline{K}_1\wedge K_2\wedge K_3\wedge K_4)\end{aligned}$$

利用这个式子就可以设计出电灯的控制电路.

习　题

1. 把下列各式化为每个变元至多出现一次的简单合取式或简单析取式，然后对变元组(p,q,r,s,t)写出它们的成真成假指派：

1.1　$((p\wedge\bar{q})\wedge\bar{t})\wedge(p\wedge\bar{q})$；

1.2　$((q\wedge\bar{r})\wedge p)\wedge(((p\wedge\bar{r})\wedge s)\wedge(t\wedge q))$；

1.3　$r\vee(((\bar{p}\vee\bar{q})\vee r)\vee t)$；

1.4　$((r\vee(\bar{p}\vee r))\vee((r\vee q)\vee\bar{s}))\vee\bar{p}$.

2. 简单合取式的一般形式为何？试写出其全部成假指派.

3. 证明任何真值函数均可用—和⊃作出.

4. 证明任何真值函数均可用如下定义的真值联结词“”作$\overline{\vee}$出：

p	q	$p\overline{\vee}q$
T	T	F
T	F	F
F	T	F
F	F	T

5. 求下面公式的析合范式和合析范式：

5.1　$(p\supset(q\supset r))\equiv(r\supset(q\supset p))$；

5.2 $\overline{(p\supset q)\wedge(r\supset p)}\vee\overline{(r\supset\bar{q})\supset\bar{p}}$.

6. 给出一组满足下列要求的能用以设计四分邮票自动出售机的电路的布尔函数:

(1) 每次输入一个一分、二分或伍分的硬币. 电路中分别用 c_1, c_2, c_5 三个量表示一分, 二分, 伍分的硬币;

(2) 当输入值总和小于 4 分时, 等待输入, 当大于等于 4 分时, 输出一张四分邮票并找零. 电路中邮票用变量 s 表示, 找零之数用 r_1, r_2, r_2' 三个量分别表示一分, 二分, 二分的硬币, 输入值的和数用三位的二进制数 $m_3m_2m_1$ 表示.

§1.4 命题演算永真公式的公理系统

所谓命题演算永真公式的公理系统就是给出若干条命题演算永真公式(称为公理), 再给出若干条由永真公式推出永真公式的规则(称为推理规则), 使得一切永真公式均能由这些公理出发, 利用这些推理规则一步步地推出.

现在问题是这样的永真公式和规则能不能找到. 其回答是肯定的. 下面将给出命题演算永真公式的公理系统. 首先, 我们对一般公理系统作一简单说明.

每一个公理系统都包括两大部分. 第一部分是组成部分, 它是公理系统的概念部分, 用以指明该系统所讨论的对象, 也即指明该系统中的项和公式. 第二部分是推理部分, 用以指明什么样的公式被认为是该系统中的定理.

I. 组成部分

命题变元: (1) p 是命题变元;

(2) 如果 ξ 是命题变元, 则 $\xi|$ 也是命题变元;

(3) 命题变元仅限于此.

由此可知, $p, p|, p||, p|||$ 等都是命题变元. 为了方便起见, 今后把它们写为 p, q, r 等, 并可带有下标.

联结词：$\bar{}$, $\wedge$, $\vee$, $\supset$, $\equiv$是联结词.

括号：(,)是括号.

公式：(1) 命题变元是公式;

(2) 如果α是公式则$\bar{\alpha}$也是公式;

(3) 如果α, β是公式，则$(\alpha\wedge\beta)$，$(\alpha\vee\beta)$，$(\alpha\supset\beta)$，$(\alpha\equiv\beta)$也是公式;

(4) 公式仅限于此.

由此可知，公式是由p, $|$, $\bar{}$, $\wedge$, $\vee$, $\supset$, $\equiv$, (,)，这9个基本符号按一定的规则组成.

p, $p|$, $p||$等命题变元是以真假为变域的变元. $\bar{}$, $\wedge$, $\vee$, $\supset$, $\equiv$是五个命题联结词，分别为否定词、合取词、析取词、蕴函词、等价词，在这种意义之下，公式就是命题演算公式. 仿前可定义公式的永真、可满足、非永真和永假等概念.

注意，我们这里的公式，在一些数理逻辑书中称为合适公式.

II. 推理部分

一、公理

下列15种公式模式均是公理，其中α, β, γ可为任何公式.

10. $\alpha\supset\alpha$

11. $(\alpha\supset(\beta\supset\gamma))\supset(\beta\supset(\alpha\supset\gamma))$

12. $(\alpha\supset\beta)\supset((\beta\supset\gamma)\supset(\alpha\supset\gamma))$

13. $(\alpha\supset(\alpha\supset\beta))\supset(\alpha\supset\beta)$

14. $(\alpha\equiv\beta)\supset(\alpha\supset\beta)$

15. $(\alpha\equiv\beta)\supset(\beta\supset\alpha)$

16. $(\alpha\supset\beta)\supset((\beta\supset\alpha)\supset(\alpha\equiv\beta))$

17. $(\alpha\wedge\beta)\supset\alpha$

18. $(\alpha\wedge\beta)\supset\beta$

19. $\alpha\supset(\beta\supset(\alpha\wedge\beta))$

20. $\alpha \supset (\alpha \vee \beta)$

21. $\beta \supset (\alpha \vee \beta)$

22. $(\beta \supset \alpha) \supset ((\gamma \supset \alpha) \supset ((\beta \vee \gamma) \supset \alpha))$

23. $(\alpha \supset \bar{\beta}) \supset (\beta \supset \bar{\alpha})$

24. $\bar{\bar{\alpha}} \supset \alpha$

二、推理规则

本系统中只有一条推理规则，称为分离规则(简记为"分").

$$\alpha \supset \beta,\ \alpha \vdash \beta$$

(读为：对 $\alpha \supset \beta$，α 实施分离规则可得 β.)

三、定理

(1) 公理为定理；

(2) 如果 $\alpha \supset \beta$，α 为定理，则由它们实施分离规则所得的 β 也是定理；

(3) 定理仅限于此.

在对本系统进行详细推演以前，先提出几点注意.

第一，10—24 中任何一条，均是无穷多条的公式的集合，例如

$$\alpha \supset \alpha,\ \beta \supset \beta,\ (\alpha \supset \beta) \supset (\alpha \supset \beta),\ (p \wedge q) \supset (p \wedge q)$$

等等均是公理 10，而不能把它们看成是由公理 10 作代入而得.本系统没有代入规则.

第二，要证明一公式是定理，必须也只须给出它的证明过程.

什么是证明过程？设有一系列公式 $\alpha_1, \alpha_2, \cdots, \alpha_n$，如果每个 α_i $(1 \leqslant i \leqslant n)$

(1) 或者是公理之一；

(2) 或者是由前面的 $\alpha_h, \alpha_k (h, k < i)$ 实施分离规则而得；

(3) α_n 即 β.

则说公式序列 $\alpha_1, \alpha_2, \cdots, \alpha_n$ 是定理 β 的永真证明过程（也称永真推理过程）. β 便是一定理，各个 α 叫做在 β 的证明过程中的

中间结果.

一般除要求给出证明过程外还要求同时给出证明根据. 所谓证明根据是: 当某个 α_i 是公理时, 便在它旁边注明它是哪一条公理, 当某个 α_i 不是公理, 而是由 α_k, α_l 实施分离规则而得时, 便在它旁边注明"分 $\alpha_k\alpha_l$".

按上面的定义要求, 永真证明过程和证明根据都是详尽的. 现举一例:

试证: $(\alpha \vee \beta) \supset (\beta \vee \alpha)$ 为定理

证: 公理 20=(1): $\beta \supset (\beta \vee \alpha)$

公理 21=(2): $\alpha \supset (\beta \vee \alpha)$

22=(3): $(\alpha \supset (\beta \vee \alpha)) \supset ((\beta \supset (\beta \vee \alpha)) \supset ((\alpha \vee \beta) \supset (\beta \vee \alpha)))$

分(3)(2)=(4): $(\beta \supset (\beta \vee \alpha)) \supset ((\alpha \vee \beta) \supset (\beta \vee \alpha))$

分(4)(1)=(5): $(\alpha \vee \beta) \supset (\beta \vee \alpha)$

(5)即为所要证明的定理.

第三, 虽然本系统中只用一条分离规则就足够了, 但是对于今后的推导是不够方便的. 为了推导方便起见, 我们将从这条基本规则出发, 引出一些导出规则. 如何引出导出规则呢? 假设我们已证明了一条定理 $\alpha \supset \beta$, 编号为(a), 又证明了一条定理 α, 编号为(b), 由此可得

$$\text{分(a)(b)=(c)} \qquad \beta$$

这里分(a)(b)称为 β 的证明根据, (c)为其编号, "β"这条定理应该看作对(a), (b)实施分离规则的结果, 但是我们也可以看成是对(b)实施"分(a)"规则的结果, 即

$$\text{分(a)规则} \qquad \alpha \vdash \beta$$

换言之, 每逢我们有一条定理(a): $\alpha \supset \beta$, 我们便有一条相应的分(a)规则: $\alpha \vdash \beta$. 即由(a)的前件(即 α)可得出(a)的后件(即 β).

同理，假定我们证明了一条定理$\alpha\supset(\beta\supset\gamma)$，编号为(a)，又假如我们证明了两条定理：

$$(b):\ \alpha,\qquad (c):\ \beta.$$

虽然可作如下的推理

$$\text{分}(a)(b)=(d)\qquad \beta\supset\gamma,$$

$$\text{分}(d)(c)=(e)\qquad \gamma,$$

将(d)代入得

$$\text{分分}(a)(b)(c)=(e)\qquad \gamma$$

这里我们既可以把γ看作两次实施分离规则的结果，第一次是对(a)，(b)实施分离规则，第二次是对(d)，(c)实施分离规则，但也可以把它看作是对(b)，(c)实施“分分(a)”规则的结果，即

$$\text{分分}(a)\text{规则：}\ \alpha,\beta\vdash\gamma$$

即由(a)的两个前件α及β，可以得出(a)的后件γ.

同理，如果我们有定理(a)：$\alpha\supset(\beta\supset(\gamma\supset\delta))$则我们有如下规则

$$\text{分分分}(a)\text{规则：}\ \alpha,\beta,\gamma\vdash\delta$$

即由(a)的三个前件α,β,γ可以得出(a)的后件δ.

熟悉这种导出规则，对于今后的学习将有很大帮助，现举一例.

公理 22　$(\alpha\supset\gamma)\supset((\beta\supset\gamma)\supset((\alpha\vee\beta)\supset\gamma))$

故有　　分 22 规则　$\alpha\supset\gamma\vdash(\beta\supset\gamma)\supset((\alpha\vee\beta)\supset\gamma)$

　　分分 22 规则　$\alpha\supset\gamma,\beta\supset\gamma\vdash(\alpha\vee\beta)\supset\gamma$

因此一旦我们证得下列形式的定理时

(a)　　$(\cdots\cdots)\supset(\Delta\Delta\Delta)$

(b)　　$(\sim\sim\sim)\supset(\Delta\Delta\Delta)$

这里$(\cdots\cdots)$，$(\Delta\Delta\Delta)$，$(\sim\sim\sim)$表示一些相当复杂的公式. 我们便立即可应用“分分 22”规则得

分分 22(a)(b)　　((……) ∨ (~~~~)) ⊃ (ΔΔΔ)

如果不用这条“分分 22”，而用分离规则，那么只能如下进行：

22　((……) ⊃ (ΔΔΔ)) ⊃ (((~~~~) ⊃ (ΔΔΔ)) ⊃ (((……) ∨ (~~~~)) ⊃ · (ΔΔΔ)))

(a)　(……) ⊃ (ΔΔΔ)

(b)　(~~~~) ⊃ (ΔΔΔ)

分 22(a) = (c)　((~~~~) ⊃ (ΔΔΔ)) ⊃ (((……) ∨ (~~~~)) ⊃ (ΔΔΔ))

分(c)(b) = (d)　((……) ∨ (~~~~)) ⊃ (ΔΔΔ)

显然这样写的话，除增加麻烦，增加出错的机会以外，没有其它好处.

利用“分分 22”规则，上面的关系 $(\alpha \vee \beta) \supset (\beta \vee \alpha)$ 的证明可改为

20 = (1)　$\beta \supset (\beta \vee \alpha)$

21 = (2)　$\alpha \supset (\beta \vee \alpha)$

分分 22(2)(1) = (3)　$(\alpha \vee \beta) \supset (\beta \vee \alpha)$

此证明过程显然比上面的简单. 因此，今后我们将大量使用导出规则.

第四，有部分数理逻辑学者(直觉主义者)是不承认公理 24 及利用其所推出的一切定理. 直觉主义的论点是：既不承认排中律，也不否认排中律，即不承认排中律为真，但承认排中律不假. 他们认为不假和真不是一回事. 一命题为真必须证明它为真，一命题为假必须证明它为假，即必须证明由它可以导出矛盾. 直觉主义的基本观点是能行性观点，他们的工作与能行性理论相关，所以工作成果是很有价值的. 为此今后对凡要用公理 24 才能证明的定理，都特别用“·”标出来.

由公理 10～24° 所组成的系统叫做古典系统；由公理 10～23

所组成的系统叫做极小演算；由公理 10～23 以及下列公理

25* $((\alpha\supset\beta)\supset\bar{\gamma})\supset((\alpha\supset\bar{\beta})\supset\bar{\gamma})$

所组成的系统叫做直觉系统．这个直觉系统比 A. Heyting 提出的直觉系统(即用 $\bar{\alpha}\supset(\alpha\supset\beta)$ 替代 25* 的系统)要弱，但更为适合直觉主义的观点．

习　题

试写出相应于下列各式的导出规则：

1. $(p\wedge(p\equiv q))\supset(p\wedge q)$；
2. $(\alpha\equiv\beta)\supset(\alpha\supset\beta)$；
3. $(\alpha\supset(\beta\supset\gamma))\supset((\alpha\supset\beta)\supset(\alpha\supset\gamma))$；
4. $((\alpha\supset\beta)\supset\gamma)\supset((\delta\supset\varepsilon)\supset((\varepsilon\supset(\alpha\supset\beta))\supset(\delta\supset\gamma)))$.

试证明下列定理：

5. $\alpha\equiv\alpha$；
6. $(\alpha\vee\alpha)\supset\alpha$；
7. $\alpha\equiv(\alpha\vee\alpha)$.

§1.5　若干重要的导出规则

首先我们给出相应于公理的导出规则：

分 11:　$\alpha\supset(\beta\supset\gamma)\vdash\beta\supset(\alpha\supset\gamma)$

分分 11:　$\alpha\supset(\beta\supset\gamma),\beta\vdash\alpha\supset\gamma$　　(挖心规则)

分分 12:　$\alpha\supset\beta,\beta\supset\gamma\vdash\alpha\supset\gamma$　　(可传规则)

分 13:　$\alpha\supset(\alpha\supset\beta)\vdash\alpha\supset\beta$　　(凝缩规则)

分 14:　$\alpha\equiv\beta\vdash\alpha\supset\beta$

分 15:　$\alpha\equiv\beta\vdash\beta\supset\alpha$

分分 16:　$\alpha\supset\beta,\beta\supset\alpha\vdash\alpha\equiv\beta$　　(充要规则)

分分 19:　$\alpha,\beta\vdash\alpha\wedge\beta$　　(合取规则)

分分 22:　$\alpha\supset\gamma,\beta\supset\gamma\vdash(\alpha\vee\beta)\supset\gamma$　　(析取规则)

分 23: $\alpha\supset\bar{\beta}\vdash\beta\supset\bar{\alpha}$ （逆否规则）

分 24°: $\bar{\bar{\alpha}}\vdash\alpha$

由此容易证明加头定理和若干主要的导出规则.

加头定理

100: $(\alpha\supset\beta)\supset((\gamma_1\supset\alpha)\supset(\gamma_1\supset\beta))$

101: $(\alpha\supset\beta)\supset((\gamma_n\supset(\gamma_{n-1}\supset\cdots\supset(\gamma_1\supset\alpha)))$
$\supset(\gamma_n\supset(\gamma_{n-1}\supset\cdots\supset(\gamma_1\supset\beta))))$

其中 n 为任意的确定的自然数，且本节中的 m, i, j, k 均同此.

[证] $12=(1)$: $(\gamma_1\supset\alpha)\supset((\alpha\supset\beta)\supset(\gamma_1\supset\beta))$

分 11, $(1)=100$: $(\alpha\supset\beta)\supset((\gamma_1\supset\alpha)\supset(\gamma_1\supset\beta))$

$12=(2)$: $(\gamma_2\supset(\gamma_1\supset\alpha))\supset(((\gamma_1\supset\alpha)$
$\supset(\gamma_1\supset\beta))\supset(\gamma_2\supset(\gamma_1\supset\beta)))$

分 11, $(2)=(3)$: $((\gamma_1\supset\alpha)\supset(\gamma_1\supset\beta))$
$\supset((\gamma_2\supset(\gamma_1\supset\alpha))\supset(\gamma_2\supset(\gamma_1\supset\beta)))$

分分 12, 100, $(3)=(4)$: $(\alpha\supset\beta)\supset((\gamma_2\supset(\gamma_1\supset\alpha))$
$\supset(\gamma_2\supset(\gamma_1\supset\beta)))$

继续这个过程最后可得

101: $(\alpha\supset\beta)\supset((\gamma_n\supset(\gamma_{n-1}\supset\cdots\supset(\gamma_1\supset\alpha)))$
$\supset(\gamma_n\supset(\gamma_{n-1}\supset\cdots\supset(\gamma_1\supset\beta))))$

对加头定理实施分离规则可得下列加头规则:

分 100: $\alpha\supset\beta\vdash(\gamma\supset\alpha)\supset(\gamma\supset\beta)$

分 101: $\alpha\supset\beta\vdash((\gamma_1\supset(\gamma_2\supset\cdots\supset(\gamma_n\supset\alpha)))$
$\supset(\gamma_1\supset(\gamma_2\supset\cdots\supset(\gamma_n\supset\beta))))$

这是一条十分重要的导出规则，特记为“∇_n^1 规则”，即

$$\nabla_n^1\colon\ \alpha\supset\beta\vdash(\gamma\supset)^n\alpha\supset(\gamma\supset)^n\beta$$

其中 $(\gamma\supset)^n$ 是 $(\gamma_1\supset(\gamma_2\supset\cdots\supset(\gamma_n\supset\alpha)))$ 的缩写，表示在 α 之前

加了 n 个不同的蕴涵前件. 其余与此类似. 由此还可得“分 ∇_n^1”规则

分 ∇_n^1: $\alpha\supset\beta,(\gamma\supset)^n\alpha\vdash(\gamma\supset)^n\beta$

这也是一条常用的规则, 必须熟习之.

对 11 施行“分 ∇_n^1”规则可得

分 $\nabla_n^1 11$: $(\gamma\supset)^n(\alpha\supset(\beta\supset\delta))\vdash(\gamma\supset)^n(\beta\supset(\alpha\supset\delta))$

这就是把第 $n+1$ 个前件与第 $n+2$ 个前件对调的规则, 亦即相邻两前件对调规则, 逐次实施这个规则就可以把第 i 前件与第 j 前件对调, 因此, 今后我们允许使用下述规则

调 ij: $(\gamma_1\supset\cdots(\gamma_i\supset\cdots(\gamma_j\supset\cdots(\gamma_n\supset\alpha))))$

$\vdash(\gamma_1\supset\cdots(\gamma_j\supset\cdots(\gamma_i\supset\cdots(\gamma_n\supset\alpha))))$

由“∇_n^1”规则和“调 ij”规则可得

$$\begin{aligned}\alpha\supset(\beta\supset\gamma)&\underset{\nabla_i^1}{\vdash}(\delta\supset)^i\alpha\supset(\delta\supset)^i(\beta\supset\gamma)\\&\underset{\blacksquare}{\vdash}\beta\supset(((\delta\supset)^i\alpha)\supset(\delta\supset)^i\gamma)\\&\underset{\nabla_j^1}{\vdash}((e\supset)^j\beta)\supset(e\supset)^j(((\delta\supset)^i\alpha)\supset(\delta\supset)^i\gamma)\\&\underset{\blacksquare}{\vdash}((\delta\supset)^i\alpha)\supset(((e\supset)^j\beta)\supset(\delta\supset)^i(e\supset)^j\gamma)\end{aligned}$$

即, 调 ∇_j 调 ∇_i

$$\alpha\supset(\beta\supset\gamma)\vdash((\delta\supset)^i\alpha)\supset(((e\supset)^j\beta)\supset(\delta\supset)^i(e\supset)^j\gamma)$$

此规则也是加头规则, 特称为“∇_{ij}^2”规则, 即

∇_{ij}^2: $\alpha\supset(\beta\supset\gamma)\vdash(\delta\supset)^i\alpha\supset((e\supset)^j\beta\supset(\delta\supset)^i(e\supset)^j\gamma)$

由此可得“分分 ∇_{ij}^2”规则

分分 ∇_{ij}^2: $\alpha\supset(\beta\supset\gamma),(\delta\supset)^i\alpha,(e\supset)^j\beta\vdash(\delta\supset)^i(e\supset)^j\gamma$

这也是一条常用规则, 必须熟习之.

由“调”规则和“凝”(凝缩)规则得下列一般凝缩规则

凝：$\gamma_1\supset\cdots(\beta\supset\cdots(\gamma_i\supset\cdots(\beta\supset\cdots(\gamma_n\supset\cdot\alpha))))$

$\vdash\gamma_1\supset\cdots(\beta\supset\cdots(\gamma_i\supset\cdots(\gamma_n\supset\alpha)))$

根据这个规则，今后我们永可把相同的前件进行合并.

加头规则 ∇^1_n 和 ∇^2_{nm} 是今后经常使用的规则，使用时常常不强调 n、m 的具体数值，所以我们把 ∇^1_n 省写为 ∇^1，把 ∇^2_{nm} 省写为 ∇^2. ∇^1 是指由蕴涵式 $\alpha\supset\beta$ 可推出如果蕴涵式的(第一)前件 α 加若干个"头"，则后件 β 也加同样的"头". ∇^2 是指由蕴涵式 $\alpha\supset(\beta\supset\gamma)$ 可推出如果蕴涵式的第一前件 α 加若干个"头"，第二前件 β 加若干个"头"，则后件 γ 也加第一和第二前件所加之"头".

习　题

1. 试推导出下列规则:

∇^3_{ijk}: $\alpha\supset(\beta\supset(\gamma\supset\delta))\vdash((\xi\supset)^i\alpha)\supset(((\eta\supset)^j\beta)\supset(((\zeta\supset)^k\gamma)\supset(\xi\supset)^i(\eta\supset)^j(\zeta\supset)^k\delta))$.

2. 试写出下列导出规则:

(a) 分 12，(分 12)^{2n}，(分 12)$^{2n+1}$;

(b) 分 17，分 18;

(c) 分 20，分 21.

3. 试写出下列导出规则:

(a) 分分分 ∇^3_{ijk};

(b) 分分 $\nabla^2_{ij}11$，分分 $\nabla^2_{ij}12$;

(c) 分 ∇^1_i14，分 ∇^1_j15，分分 $\nabla^2_{ij}16$;

(d) 分分 $\nabla^2_{ij}19$;

(e) 分分 $\nabla^2_{ij}22$，分分分 $\nabla^3_{ijk}22$.

4. 证明下列定理:

102　$(\alpha_1\supset(\alpha_2\supset(\alpha_3\supset(\alpha_4\supset\beta))))\supset(\alpha_1\supset(\alpha_4\supset(\alpha_2\supset(\alpha_3\supset\beta))))$;

103　$(\alpha\supset)^n\beta\supset((\beta\supset\gamma)\supset(\alpha\supset)^n\gamma)$;

104　$((\gamma\supset)^i(\alpha\supset\beta))\supset((\delta\supset)^j\alpha\supset(\gamma\supset)^i(\delta\supset)^j\beta)$.

§1.6 假设推理过程和推理定理

上面介绍了永真推理过程，要找一个定理的永真推理过程往往是相当困难的，在日常的推理中，一般不采用这种推理方法，而采用所谓假设推理法．下面我们讨论这种证明方法．

先把推理规则的概念解释一下：

设有一如下的推理规则 R

R：　$\alpha_1, \alpha_2, \cdots, \alpha_k \vdash \beta$

则我们说，β 是由 $\alpha_1, \alpha_2, \cdots\cdots \alpha_k$ 实施规则 R 而得到的．例如，我们前面说 β 是由 $\alpha \supset \beta, \alpha$ 实施分离规则而得．

定义　如果能够作出一系列公式 $\alpha_1, \alpha_2, \cdots, \alpha_n$ 它们具有下列性质：

(i) 诸 α_i 或为公理之一；

(ii) 或为公式 $\gamma_1, \gamma_2, \cdots, \gamma_k$ 之一（诸 γ 叫做假设）；

(iii) 或由前面的 α_g, α_h（g, h 均小于 i）实施分离规则而得；

(iv) 或由前面若干个 α 实施新推理规则 $R_1, R_2, \cdots, R_l$ 之一而得；

(v) α_n 即 β．

那么，这个公式系列 $\alpha_1, \alpha_2, \cdots, \alpha_n$ 便叫做增加新规则 $R_1, R_2, \cdots, R_l$ 后，由公式 $\gamma_1, \gamma_2, \cdots, \gamma_k$ 证明 β 的**假设证明过程**，亦称**假设推理过程**；β 叫做增加新规则 $R_1, R_2, \cdots, R_l$ 后由诸 γ 推出的**结论**，今后记为

$$\gamma_1, \gamma_2, \cdots, \gamma_k \vdash (R_1, R_2, \cdots, R_l)\beta$$

如果没有增加新规则，即把定义中的(iv)删去，这时就说 β 是由诸 γ 推出的结论，并记为

$$\gamma_1, \gamma_2 \cdots, \gamma_k \vdash \beta$$

这是一条规则，因此假设推理过程实质上是规则的推广．

定义中的“(ii)或为 γ_1, γ_2, …, γ_k 之一”这句话可以有两种解释,其一是指“或为诸 γ 本身之一”,即不把诸 γ 看作是一类公式的集合,而看作是一条公式. 例如,若 $\delta=\alpha\supset(\beta\supset\gamma)$,则 $\alpha_1\supset(\beta_1\supset\gamma_1)$就不被看作是 δ;其二是指“或为诸 γ 型之一”,即把诸 γ 看作是一类形式相同的公式的集合,例如,若 $\delta=\alpha\supset(\beta\supset\gamma)$则 $\alpha_1\supset(\beta_1\supset\gamma_1)$, $(e_1\supset e_2)\supset(e_3\supset e_4)$等等均是 δ.

在下面我们采用第一种解释,不用第二种解释,而且也不增加新规则 $R_1, R_2, \cdots\cdots, R_t$.

推理定理 如果 $\gamma_1, \gamma_2, \cdots, \gamma_k\vdash\alpha$,在推理过程中对诸 γ 永不作代入,且诸 γ 至少被使用一次,则必有

$$\gamma_1, \gamma_2, \cdots, \gamma_h\vdash\gamma_{h+1}\supset(\cdots\supset(\gamma_k\supset\cdot\alpha))\ (0\leqslant h<k)$$

[**证**] 因为 $\gamma_1, \gamma_2, \cdots, \gamma_k\vdash\alpha$,所以可找出一公式系列 $\alpha_1, \alpha_2, \cdots, \alpha_n=\alpha$ 它们满足假设证明过程的定义中的条件(i)~(iii),今另作一公式系列 $\beta_1, \beta_2, \cdots, \beta_n$ 其作法如下:

如果 α_i 为公理之一,则 β_i 仍为 α_i(即仍为公理).

如果 α_i 为假设 γ_1, γ_2, …, γ_h 之一,则 β_i 仍为 α_i(即仍为假设).

如果 α_i 为假设 $\gamma_{h+1}, \cdots, \gamma_k$ 之一,则 β_i 为 $\alpha_i\supset\alpha_i$(即公理 10,可看成是由 α_i 加头“$\alpha_i\supset$”而得).

如果 α_i 由 α_{m1}, α_{m2}, (m_1, m_2 均小于 i)利用分离规则而得,可设 $\beta_{m1}=(\delta\supset)^s\alpha_{m1}$, $\beta_{m2}=(e\supset)^t\alpha_{m2}$,其中$(\delta\supset)^s(e\supset)^t$ 均是 $\gamma_{h+1}\supset, \cdots, \gamma_k\supset$中的若干个的连接,则 β_i 为$(\delta\supset)^s(e\supset)^t\alpha_i$, (即分分 $\nabla^2 10(\beta_{m1})(\beta_{m2})$可看成是 α_i 加头而得,所加之头为 α_{m1}, α_{m2} 所加之头的总和).

因此,β_n 为由 α_n(即 α)加头而得,所加之头是若干个 $\gamma_{h+1}\supset$, …, $\gamma_k\supset$的连接. 根据假设诸 γ 至少被使用一次;因此 β_n 所加之头中各 $\gamma_{h+1}\supset$, …, $\gamma_k\supset$至少出现一次. 当然同一“$\gamma_i\supset$”可能出

现多次，但利用凝缩规则可使每一“$\gamma_i\supset$”出现一次，再使用“调”规则可得

$$\gamma_{k+1}\supset(\cdots\supset(\gamma_k\supset\alpha))$$

综上所述，我们有一公式系列 $\beta_1,\beta_2,\cdots,\beta_n,\ \gamma_{k+1}\supset(\cdots\supset(\gamma_k\supset\alpha))$，其中各$\beta_i$或者为公理之一，或者为$\gamma_1,\cdots,\gamma_k$之一，或为由前面的 β_{m1},β_{m2} 使用分分 $\nabla^2 10$ 而得，而 $\gamma_{k+1}\supset(\cdots\supset(\gamma_k\supset\alpha))$是由 β_n利用“凝”和“调”规则而得，故由假设推理过程的定义得

$$\gamma_1,\cdots\gamma_k\vdash\gamma_{k+1}\supset(\cdots\supset(\gamma_k\supset\alpha))$$

定理得证.

当然 $k=0$ 时定理 仍成立，即为在没有假设的情形可推得 $\gamma_1\supset(\cdots\supset(\gamma_k\supset\alpha))$，此推理过程即为永真推理过程.

利用推理定理可使永真公式 的推导过程 得到本质上的简化，今举例如下:

例 1: $(\alpha\supset(\beta\supset\gamma))\supset((\delta\supset\alpha)\supset((\delta\supset\beta)\supset(\delta\supset\gamma)))$

[证] 把公式中的前件当作假设，并用“*”表示.

*(1) $\alpha\supset(\beta\supset\gamma)$

*(2) $\delta\supset\alpha$

*(3) $\delta\supset\beta$

*(4) δ

分(2)(4)=(5) α

分(3)(4)=(6) β

分(1)(5)=(7) $\beta\supset\gamma$

分(7)(6)=(8) γ

由定义得 $\alpha\supset(\beta\supset\gamma),\delta\supset\alpha,\delta\supset\beta,\delta\vdash\gamma$

由推理定理得 $(\alpha\supset(\beta\supset\gamma))\supset((\delta\supset\alpha)\supset((\delta\supset\beta)\supset(\delta\supset\gamma)))$.

根据推理定理的证明，可把这个假设推理过程译为永真推理过程，方法如下:

假设*(1) 改为 $10=(1)'$ $(\alpha\supset(\beta\supset\gamma))\supset(\alpha\supset(\beta\supset\gamma))$

*(2) 改为 $10=(2)'$ $(\delta\supset\alpha)\supset(\delta\supset\alpha)$

*(3) 改为 $10=(3)'$ $(\delta\supset\beta)\supset(\delta\supset\beta)$

*(4) 改为 $10=(4)'$ $\delta\supset\delta$

分(2)(4)改为分分 $\nabla^2 10(2)'(4)'=(5)'$

$(\delta\supset\alpha)\supset(\delta\supset\alpha)$

分(3)(4)改为分分 $\nabla^2 10(3)'(4)'=(6)'$

$(\delta\supset\beta)\supset(\delta\supset\beta)$

分(1)(5)改为分分 $\nabla^2 10(1)'(5)'=(7)'$

$(\alpha\supset(\beta\supset\gamma))\supset((\delta\supset\alpha)\supset(\delta\supset(\beta\supset\gamma)))$

分(7)(6)改为分分 $\nabla^2 10(7)'(6)'=(8)'$

$(\alpha\supset(\beta\supset\gamma))\supset((\delta\supset\alpha)\supset(\delta\supset((\delta\supset\beta)\supset(\delta\supset\gamma))))$

调凝调$(8)'=(9)'$

$(\alpha\supset(\beta\supset\gamma))\supset((\delta\supset\alpha)\supset((\delta\supset\beta)\supset(\delta\supset\gamma)))$

由本例可见，利用推理定理的方式可总结如下：

第一，把待证公式前件一一列出，作为假设，并在编号前面标以“*”（表示它为非永真公式）.

第二，照通常永真推理过程进行推理（这时的永真推理过程已十分简单），但只准使用分离规则和永真的导出规则，不准使用相应于各假设及中间结果的导出规则. 这是因为我们规定，不得把各假设看成是一类公式的集合，而只能看成是其本身.

第三，当推导出待证公式的后件时便得结果.

第四，根据推理定理，可把假设推理过程译为永真推理过程.

例 2：$(\gamma\supset(\alpha\supset\beta))\supset((\gamma\supset(\beta\supset\alpha))\supset(\gamma\supset(\alpha\equiv\beta)))$

[证] *(1) $\gamma\supset(\alpha\supset\beta)$

*(2) $\gamma\supset(\beta\supset\alpha)$

*(3) γ

分(1)(3)=(4)　$\alpha\supset\beta$

分(2)(3)=(5)　$\beta\supset\alpha$

分分16(4)(5)=(6)　$\alpha\equiv\beta$

由定义得　$\gamma\supset(\alpha\supset\beta),\ \gamma\supset(\beta\supset\alpha),\ \gamma\vdash\alpha\equiv\beta$

由推理定理得　$(\gamma\supset(\alpha\supset\beta))\supset((\gamma\supset(\beta\supset\alpha))\supset(\gamma\supset(\alpha\equiv\beta)))$

现把这个过程译为永真推理过程.

$*(1)\to10=(1)'$　$(\gamma\supset(\alpha\supset\beta))\supset(\gamma\supset(\alpha\supset\beta))$

$*(2)\to10=(2)'$　$(\gamma\supset(\beta\supset\alpha))\supset(\gamma\supset(\beta\supset\alpha))$

$*(3)\to10=(3)'$　$\gamma\supset\gamma$

分(1)(3)→分分 $\nabla^2$10(1)' (3)' =(4)'　$(\gamma\supset(\alpha\supset\beta))\supset(\gamma\supset(\alpha\supset\beta))$

分(2)(3)→分分 $\nabla^2$10(2)' (3)' =(5)'　$(\gamma\supset(\beta\supset\alpha))\supset(\gamma\supset(\beta\supset\alpha))$

分分 16(4)(5)→分分 $\nabla^2_{4,5}$16(4)'　(5)' =(6)'　$(\gamma\supset(\alpha\supset\beta))\supset(\gamma\supset((\gamma\supset(\beta\supset\alpha))\supset(\gamma\supset(\alpha\equiv\beta))))$

调凝(6)' =(7)'　$(\gamma\supset(\alpha\supset\beta))\supset((\gamma\supset(\beta\supset\alpha))\supset(\gamma\supset(\alpha\equiv\beta)))$

例3：$(\gamma\supset\alpha)\supset((\gamma\supset\beta)\supset(\gamma\supset(\alpha\wedge\beta)))$

[证]　为了方便起见，我们将把假设推理过程与永真推理过程分左右并列写出.

$*(1)$	$\gamma\supset\alpha$	$10=(1)$	$(\gamma\supset\alpha)\supset(\gamma\supset\alpha)$
$*(2)$	$\gamma\supset\beta$	$10=(2)$	$(\gamma\supset\beta)\supset(\gamma\supset\beta)$
$*(3)$	γ	$10=(3)$	$\gamma\supset\gamma$
分(1)(3)=(4)	α	分分$\nabla^2$10(1)(3)=(4)	$(\gamma\supset\alpha)\supset(\gamma\supset\alpha)$
分(2)(3)=(5)	β	分分$\nabla^2$10(2)(3)=(5)	$(\gamma\supset\beta)\supset(\gamma\supset\beta)$
分分19(4)(5)=(6)	$\alpha\wedge\beta$	分分$\nabla^2$19(4)(5)=(6)	$(\gamma\supset\alpha)\supset(\gamma\supset((\gamma\supset\beta)\supset(\gamma\supset(\alpha\wedge\beta))))$
由推理定理得证.		调凝(6)=(7)	$(\gamma\supset\alpha)\supset((\gamma\supset\beta)\supset(\gamma\supset(\alpha\wedge\beta)))$

例 4: $((\alpha\wedge\beta)\gamma)\supset\gamma)\supset(\alpha\supset(\beta\supset\gamma))$

[证]

*(1)	$(\alpha\wedge\beta)\supset\gamma$	10=(1)	$((\alpha\wedge\beta)\supset\gamma)\supset((\alpha\wedge\beta)\supset\gamma)$
*(2)	α	10=(2)	$\alpha\supset\alpha$
*(3)	β	10=(3)	$\beta\supset\beta$
分分19(2)(3)=(4)	$\alpha\wedge\beta$	分分∇^{A}19(2)(3)=(4)	$\alpha\supset(\beta\supset(\alpha\wedge\beta))$
分(1)(4)=(5)	γ	分分∇^{A}10(1)(4)=(5)	$((\alpha\wedge\beta)\supset\gamma)\supset(\alpha\supset(\beta\supset\gamma))$

由推理定理得证.

下面列出一些比较重要的定理.

关于蕴涵的定理

100 $(\beta\supset\gamma)\supset((\alpha\supset\beta)\supset(\alpha\supset\gamma))$

101 $(\beta\supset\gamma)\supset((\alpha\supset)^{n}\beta\supset(\alpha\supset)^{n}\gamma)$

其中 n 为任一确定的自然数. 以下 n, i, j 均同此.

102 $(\gamma_1\supset\cdots(\gamma_i\supset\cdots(\gamma_j\supset\cdots(\gamma_n\supset\alpha))))$
$\supset(\gamma_1\supset\cdots(\gamma_j\supset\cdots(\gamma_i\supset\cdots(\gamma_n\supset\alpha))))$

103 $(\alpha\supset)^{n}\beta\supset((\beta\supset\gamma)\supset(\alpha\supset)^{n}\gamma)$

104 $((\gamma\supset)^{i}(\alpha\supset\beta))\supset((\delta\supset)^{j}\alpha\supset(\gamma\supset)^{i}(\delta\supset)^{j}\beta)$

105 $(\alpha\supset(\beta\supset\gamma))\supset((\alpha\supset\beta)\supset(\alpha\supset\gamma))$

106 $(\alpha\supset(\beta\supset\gamma))\supset((\delta\supset\alpha)\supset((\delta\supset\beta)\supset(\delta\supset\gamma)))$

107 $\alpha\supset((\alpha\supset\beta)\supset\beta)$

关于等价的定理

110 $((\gamma\supset)^{i}(\alpha\supset\beta))\supset(((\delta\supset)^{j}(\beta\supset\alpha))\supset(\gamma\supset)^{i}(\delta\supset)^{j}(\alpha\equiv\beta))$

111 $\alpha\equiv\alpha$

112 $(\alpha\equiv\beta)\supset(\beta\equiv\alpha)$

113 $(\alpha\equiv\beta)\supset((\alpha\supset\gamma)\equiv(\beta\supset\gamma))$

114 $(\alpha\equiv\beta)\supset((\gamma\supset\alpha)\equiv(\gamma\supset\beta))$

115 $(\alpha\equiv\beta)\supset((\gamma\equiv\delta)\supset((\alpha\supset\gamma)\equiv(\beta\supset\delta)))$

116 $(\alpha\equiv\beta)\supset((\alpha\cong\gamma)\supset(\beta\cong\gamma))$

117 $(\alpha\equiv\beta)\supset((\gamma\equiv\alpha)\supset(\gamma\cong\beta))$

118 $(\alpha\equiv\beta)\supset((\gamma\equiv\delta)\supset((\alpha\cong\gamma)\equiv(\beta\equiv\delta)))$

关于合取的定理

120 $(\gamma\supset)^i\alpha\supset((\delta\supset)^j\beta\supset(\gamma\supset)^i(\delta\supset)^j(\alpha\wedge\beta))$

121 $\alpha\supset(\alpha\wedge\alpha)$

122 $(\alpha\wedge\beta)\supset(\beta\wedge\alpha)$

123 $(\alpha\wedge(\beta\wedge\gamma))\supset((\alpha\wedge\beta)\wedge\gamma)$

124 $((\alpha\wedge\beta)\wedge\gamma)\supset(\alpha\wedge(\beta\wedge\gamma))$

125 $(\alpha\supset\beta)\supset((\gamma\wedge\alpha)\supset(\gamma\wedge\beta))$

126 $(\alpha\supset\beta)\supset((\alpha\wedge\gamma)\supset(\beta\wedge\gamma))$

127 $(\alpha\supset(\beta\supset\gamma))\supset((\alpha\wedge\beta)\supset\gamma)$

128 $((\alpha\wedge\beta)\supset\gamma)\supset(\alpha\supset(\beta\supset\gamma))$

§1.7 假设推理过程和推理定理(续)

在日常生活和科学研究中，除使用上节讲的那种假设推理法外，还使用所谓**额外假设推理法**，即在推理过程中除把待证公式中诸前件作为假设之外，还根据各种不同的情况引入不同的额外假设，最后设法消去额外假设而得结论.

让我们举例来说明.

例 1: $(\alpha\supset\beta)\supset((\alpha\vee\gamma)\supset(\beta\vee\gamma))$

[证] $*(1)$ $\alpha\supset\beta$

$*(2)$ $\alpha\vee\gamma$

$**(3)$ α [这是额外假设]

分(1)(3)=(4) β

分 20(4)=(5) $\beta\vee\gamma$

由推理定理消去(3)得(6)　$\alpha\supset(\beta\vee\gamma)$

　　**(7)　γ　　[这是额外假设]

　　分 21(7)=(8)　$\beta\vee\gamma$

由推理定理消去(7)得(9)　$\gamma\supset(\beta\vee\gamma)$

　　分分 22(6)(9)=(10)　$(\alpha\vee\gamma)\supset(\beta\vee\gamma)$

　　分(10)(2)=(11)　$\beta\vee\gamma$

由推理定理得证.

这个证明方法称为穷举法。一般当假设中有析取式时或者当推理过程中出现析取式时可采用这种证明方法. 所引入的额外假设是该析取式中的各析取项. 在例 1 中因假设中有析取式 $\alpha\vee\gamma$, 所以引入了额外假设 α 和 γ. 为了区别一般的假设和额外假设, 在额外假设之前标以双星号"**".

关于析取的定理

130　$(\delta\supset)^i(\alpha\supset\gamma)\supset((e\supset)^j(\beta\supset\gamma)\supset(\delta\supset)^i(e\supset)^j((\alpha\vee\beta)\supset\gamma)))$

131　$(\alpha\vee\alpha)\supset\alpha$

132　$(\alpha\vee\beta)\supset(\beta\vee\alpha)$

133　$(\alpha\vee(\beta\vee\gamma))\supset((\alpha\vee\beta)\vee\gamma)$

134　$((\alpha\vee\beta)\vee\gamma)\supset(\alpha\vee(\beta\vee\gamma))$

135　$(\alpha\supset\beta)\supset((\alpha\vee\gamma)\supset(\beta\vee\gamma))$

136　$(\alpha\supset\beta)\supset((\gamma\vee\alpha)\supset(\gamma\vee\beta))$

137　$(\alpha\supset\beta)\supset((\gamma\supset\delta)\supset((\alpha\vee\gamma)\supset(\beta\vee\delta)))$

另一种常用的额外假设推理法是反证法. 所谓反证法就是除把待证公式的前件作为假设外, 还把待证公式的后件的否定作为额外假设, 如果由这些假设及额外假设可以推出矛盾(既推出某一公式又推出该公式的否定), 则认为待证公式得证. 这种推理过程的合理性可由下列定理保证.

反证法推理定理　　如果有

(1) $\alpha_1, \alpha_2, \cdots, \alpha_n, \beta \vdash \gamma$

(2) $\alpha_1, \alpha_2, \cdots, \alpha_n, \beta \vdash \bar{\gamma}$

则可得　$\alpha_1, \alpha_2, \cdots, \alpha_n \vdash \bar{\beta}$

即　　$\vdash \alpha_1 \supset (\alpha_2 \supset \cdots \supset (\alpha_n \supset \bar{\beta}))$

[证]　根据推理定理由(1)(2)分别可得

(3) $\alpha_1 \supset (\cdots \supset (\alpha_n \supset (\beta \supset \gamma)))$

(4) $\alpha_1 \supset (\cdots \supset (\alpha_n \supset (\beta \supset \bar{\gamma})))$

分 $\nabla^1 23(4)=(5)$　$\alpha_1 \supset (\cdots \supset (\alpha_n \supset (\gamma \supset \bar{\beta})))$

凝分分 $\nabla^2 10(5)(3)=(6)$　$\alpha_1 \supset (\cdots \supset (\alpha_n \supset (\beta \supset \bar{\beta})))$

分 $\nabla^1 23、107=(7)$　$\beta \supset (\beta \supset \overline{\beta \supset \bar{\beta}})$

凝(7)=(8)　$\beta \supset \overline{\beta \supset \bar{\beta}}$

分 23(8)=(9)　$(\beta \supset \bar{\beta}) \supset \bar{\beta}$

分 ∇' (9)(6)=(10)　$\alpha_1 \supset (\cdots \supset (\alpha_n \supset \bar{\beta}))$

本定理得证.

例 2：$(\alpha \supset \beta) \supset (\bar{\beta} \supset \bar{\alpha})$

[证]　*(1)　$\alpha \supset \beta$

*(2)　$\bar{\beta}$

**(3)　α

分(1)(3)=(4)　β

(4)与(2)矛盾，故由反证法推理定理可得

$\alpha \supset \beta, \bar{\beta} \vdash \bar{\alpha}$

即　　$(\alpha \supset \beta) \supset (\bar{\beta} \supset \bar{\alpha})$

定理得证.

例 3：$(\bar{\alpha} \supset \alpha) \supset \alpha$

[证]　*(1)　$\bar{\alpha} \supset \alpha$

**(2)　$\bar{\alpha}$

分(1)(2)=(3)　α

(3)与(2)矛盾，故由反证法推理定理得

(4)　$(\bar{\alpha}\supset\alpha)\supset\bar{\bar{\alpha}}$

分 $\nabla^{1}24^{\circ}$(4)=(5)　$(\bar{\alpha}\supset\alpha)\supset\alpha$

定理得证.

这里要注意的是由反证法推理定理只能得(4)，而不能直接得(5). (5)必须利用 24° 才能得到.

关于否定的定理

140　$\alpha\supset\bar{\bar{\alpha}}$

141　$(\alpha\supset\beta)\supset(\bar{\beta}\supset\bar{\alpha})$

142°　$(\bar{\alpha}\supset\beta)\supset(\bar{\beta}\supset\alpha)$

143　$(\bar{\alpha}\supset\bar{\beta})\supset(\beta\supset\alpha)$

144°　$(\bar{\alpha}\supset\alpha)\supset\alpha$

145　$(\alpha\supset\bar{\alpha})\supset\bar{\alpha}$

146　$(\alpha\supset\beta)\supset((\alpha\supset\bar{\beta})\supset\bar{\alpha})$

147°　$(\alpha\supset\beta)\supset((\bar{\alpha}\supset\beta)\supset\beta)$

148　$(\alpha\supset\beta)\supset(\bar{\bar{\alpha}}\supset\bar{\bar{\beta}})$

149°　$\alpha\vee\bar{\alpha}$

还有一种常用的额外假设推理法——*半反证法*. 其思想是当待证公式的后件是一个析取式 $\alpha\vee\beta$ 时，除把待证公式的诸前件作为假设外，还把待证公式的后件中的一个析取项(例如 α)的否定作为额外假设，如果由此推出另一析取项 β，那么待证公式便认为得证. 这种推理过程的合理性可由下列定理得到保证.

半反证法推理定理　如果有

(1)　$\alpha_1,\cdots,\alpha_n,\bar{\beta}\vdash\gamma$

则可得　　$\alpha_1,\cdots,\alpha_n\vdash\beta\vee\gamma$

即得　　$\alpha_1\supset(\cdots\supset(\alpha_n\supset(\beta\vee\gamma))\cdots)$

[证] 根据推理定理由(1)得

(2) $\alpha_1 \supset (\cdots \supset (\alpha_n \supset (\bar{\beta} \supset \gamma)))$

分分 11, 136°, 149° = (3) $(\bar{\beta} \supset \gamma) \supset (\beta \vee \gamma)$

分 Δ^1(3)(2)=(4) $\alpha_1 \supset (\cdots \supset (\alpha_n \supset (\beta \vee \gamma)))$

定理得证.

注意, 本定理要用 149° 才能证明, 也即必须用 24° 才能证明. 所以使用本定理证明的定理都必须用"∘"标出.

例4.: $(\alpha \supset \beta) \supset (\bar{\alpha} \vee \beta)$

[证] *(1) $\alpha \supset \beta$

**(2) $\bar{\bar{\alpha}}$

分24°(2)=(3) α

分(1)(3)=(4) β

由半反证法推理定理知本定理得证.

第一章中我们曾指出蕴涵词"⊃"与日常用的"如果…则…"并不全同. 正因为这样, 使当有一部分永真公式用日常语言来解释时感觉有点"怪". 例如, 容易验证"$\alpha \supset (\beta \supset \alpha)$"是永真公式. 若把 α 代以"2+2=4", β 代以"花是香的", 则该公式可读为"如果 2+2=4, 则由花是香的能推出 2+2=4". 这种推理在日常生活中是从来不被使用的. 诸如这一类公式, 我们称之为蕴涵怪论. 现在我们来证明这个永真公式.

例5: $\alpha \supset (\beta \supset \alpha)$

[证] *(1) α

*(2) β

分分 19(1)(2)=(3) $\alpha \wedge \beta$

分 17(3)=(4) α

由推理定理得证.

请注意, 推导(4)绕了圈子, 更为简捷的办法是由"分 10(1)"

而得(4). 为什么不用这种简捷的证明方法呢？这是因为我们的推理定理要求各假设在证明过程中至少使用一次. 如果用"分 10 (1)", 则 *(2)未被使用, 故不能使用推理定理. 但有了例 5，推理定理中的这一要求(即各假设在证明过程中至少使用一次)就不必要了. 也就是说推理定理可改述如下:

推理定理 如果有

$$\gamma_1, \gamma_2, \cdots, \gamma_k \vdash \alpha$$

在推理过程中对诸 γ 永不作代入, 则必有

$$\gamma_1 \cdots, \gamma_h \vdash \gamma_{h+1} \supset (\cdots \supset (\gamma_k \supset \alpha)), \quad (0 \leqslant h < k)$$

关于蕴涵怪论的定理

150 $\alpha \supset (\beta \supset \alpha)$

151 $((\alpha \supset \beta) \supset (\alpha \supset \gamma)) \supset (\alpha \supset (\beta \supset \gamma))$

152° $\alpha \supset (\bar{\alpha} \supset \beta)$

153° $(\bar{\alpha} \vee \beta) \equiv (\alpha \supset \beta)$

154° $(\alpha \equiv \beta) \equiv ((\bar{\alpha} \vee \beta) \wedge (\alpha \vee \bar{\beta}))$

155° $\overline{\alpha \equiv \beta} \equiv (\alpha \equiv \bar{\beta})$

156° $((\alpha \vee \beta) \wedge (\bar{\beta} \vee \gamma)) \supset (\alpha \vee \gamma)$

157° $(\alpha \wedge \gamma) \supset ((\alpha \wedge \beta) \vee (\bar{\beta} \wedge \gamma))$

158° $\alpha \equiv (\beta \equiv (\alpha \equiv \beta))$

159° $((\alpha \equiv \beta) \equiv \gamma) \equiv (\alpha \equiv (\beta \equiv \gamma))$

§1.8 替换定理

外延性定理

160 $(\alpha \equiv \beta) \supset (\bar{\alpha} \equiv \bar{\beta})$

161 $(\alpha \equiv \beta) \supset ((\alpha \wedge \gamma) \equiv (\beta \wedge \gamma))$

162 $(\alpha \equiv \beta) \supset ((\gamma \wedge \alpha) \equiv (\gamma \wedge \beta))$

163 $(\alpha \equiv \beta) \supset ((\gamma \equiv \delta) \supset ((\alpha \wedge \gamma) \equiv (\beta \wedge \delta)))$

164 $(\alpha\equiv\beta)\supset((\alpha\vee\gamma)\equiv(\beta\vee\gamma))$

165 $(\alpha\equiv\beta)\supset((\gamma\vee\alpha)\equiv(\gamma\vee\beta))$

166 $(\alpha\equiv\beta)\supset((\gamma\equiv\delta)\supset((\alpha\vee\gamma)\equiv(\beta\vee\delta)))$

以上七个定理以及113～118分别叫做相应于其联结词的外延性公式. 利用外延性公式可以证明下面的替换定理.

替换定理 如果 $\varphi(\alpha)$ 是一个含有公式 α 的公式, $\varphi(\beta)$ 是将若干个 α 替换以 β 的结果, 则

$$(\alpha\equiv\beta)\supset(\varphi(\alpha)\equiv\varphi(\beta))$$

是可证公式

[证] 现就 $\varphi(\alpha)$ 中的联结词个数 n(不包括 α 中的联结词)施行归纳法.

奠基 $n=0$ 时, $\varphi(\alpha)$ 或者为 α, 或者为命题变元 p.

当 $\varphi(\alpha)=\alpha$ 时, $\varphi(\beta)=\beta$

因而 $(\alpha\equiv\beta)\supset(\varphi(\alpha)\equiv\varphi(\beta))$

就是 $(\alpha\equiv\beta)\supset(\alpha\equiv\beta)$

此式显然为可证公式.

当 $\varphi(\alpha)=p$ 时, $\varphi(\beta)=p$

因而 $(\alpha\equiv\beta)\supset(\varphi(\alpha)\equiv\varphi(\beta))$

就是 $(\alpha\equiv\beta)\supset(p\equiv p)$

此式显然也为可证公式.

归纳 当 $n=k+1$ 时, $\varphi(\alpha)$ 必是下列五种形状之一

$$\bar{\gamma},\ \gamma\vee\delta,\ \gamma\wedge\delta,\ \gamma\supset\delta,\ \gamma\equiv\delta$$

这时, $\varphi(\beta)$ 必为下列五种形状之一

$$\bar{\gamma}_1,\ \gamma_1\vee\delta_1,\ \gamma_1\wedge\delta_1,\ \gamma_1\supset\delta_1,\ \gamma_1\equiv\delta_1$$

其中 γ_1,δ_1 分别为在 γ,δ 中把 α 替换以 β 的结果. 根据归纳假设有

(1) $(\alpha\equiv\beta)\supset(\gamma\equiv\gamma_1)$

(2) $(\alpha\equiv\beta)\supset(\delta\equiv\delta_1)$

对于第一种形状

分分 12(1)160=(3) $(\alpha\equiv\beta)\supset(\bar{\gamma}\equiv\bar{\gamma}_1)$

即 $(\alpha\equiv\beta)\supset(\varphi(\alpha)\equiv\varphi(\beta))$

为可证公式.

对于第二种形状

分分分 106, 166, (1)(2)=(4) $(\alpha\equiv\beta)\supset((\gamma\vee\delta)\equiv(\gamma_1\vee\delta_1))$

即 $(\alpha\equiv\beta)\supset(\varphi(\alpha)\equiv\varphi(\beta))$

为可证公式.

对于第三、四、五种形状，可用同法证明.

故由数学归纳法本定理得证.

由替换定理易得下列替换规则:

替: $\varphi(\alpha),\alpha\equiv\beta\vdash\varphi(\beta)$

替: $\varphi(\alpha),\beta\equiv\alpha\vdash\varphi(\beta)$

下面是一些重要的等价式:

170 $\alpha\equiv(\alpha\wedge\alpha)$

171 $\alpha\equiv(\alpha\vee\alpha)$

172 $\alpha\equiv\bar{\bar{\alpha}}$

173 $(\alpha\wedge\beta)\equiv(\beta\wedge\alpha)$

174 $(\alpha\vee\beta)\equiv(\beta\vee\alpha)$

175 $((\alpha\wedge\beta)\wedge\gamma)\equiv(\alpha\wedge(\beta\wedge\gamma))$

176 $((\alpha\vee\beta)\vee\gamma)\equiv(\alpha\vee(\beta\vee\gamma))$

177 $(\alpha\supset(\beta\supset\gamma))\equiv((\alpha\wedge\beta)\supset\gamma)$

178 $(\alpha\wedge(\beta\vee\gamma))\equiv((\alpha\wedge\beta)\vee(\alpha\wedge\gamma))$

179 $(\alpha\vee(\beta\wedge\gamma))\equiv((\alpha\vee\beta)\wedge(\alpha\vee\gamma))$

180 $\overline{\alpha\vee\beta}\equiv(\bar{\alpha}\wedge\bar{\beta})$

181° $\overline{\alpha \wedge \beta} \equiv (\bar{\alpha} \vee \bar{\beta})$

182° $\overline{\alpha \supset \beta} \equiv (\alpha \wedge \bar{\beta})$

183° $(\alpha \supset \beta) \equiv (\bar{\alpha} \vee \beta)$

184° $(\alpha \equiv \beta) \equiv ((\bar{\alpha} \vee \beta) \wedge (\bar{\beta} \vee \alpha))$

§1.9 关于命题演算公理系统的讨论

本节讨论五个问题. 首先讨论古典系统、直觉系统和极小演算三者之间的关系，其次讨论具体的公理系统与形式的公理系统之间的关系，最后讨论公理系统的不矛盾性，完备性和独立性问题.

1.9.1 古典系统、直觉系统和极小演算之间的关系

§1.4 中已经讲了由公理 10～24° 所组成的系统叫做古典系统，由公理 10～23 所组成的系统叫做极小演算，由公理 10～23 以及公理

25* $((\alpha \supset \beta) \supset \bar{\gamma}) \supset ((\alpha \supset \bar{\bar{\beta}}) \supset \bar{\gamma})$

所组成的系统叫做直觉系统.

显然，极小演算是古典系统的子系统，也是直觉系统的子系统.

直觉系统与古典系统之间又有怎样的关系呢？现在来讨论它们的关系.

定理 直觉系统是古典系统的子系统.

[证] 分 100, 24°=(1) $(\alpha \supset \bar{\bar{\beta}}) \supset (\alpha \supset \beta)$

分 12(1)=25* $((\alpha \supset \beta) \supset \bar{\gamma}) \supset ((\alpha \supset \bar{\bar{\beta}}) \supset \bar{\gamma})$

故由 10～23 和 24° 可推出 10～23 和 25*. 本定理得证.

定理 α 在古典系统内可证当且仅当 $\bar{\bar{\alpha}}$ 在直觉系统内可证.

[证] (⟸)设 $\bar{\bar{\alpha}}$ 在直觉系统内可证. 因为直觉系统是古典系统的子系统，所以在古典系统内 $\bar{\bar{\alpha}}$ 可证.

对 $\bar{\bar{\alpha}}$ 实施分 24° 规则便得 α，故 α 在古典系统内可证.

(⟹)因为在直觉系统中有

(甲) $\alpha\supset\bar{\bar{\alpha}}$ (即 140)

又有公理 10～23，所以在直觉系统中有公理 10～23 的双重否定，即有

分(甲) $10=10^*$ $\overline{\overline{\alpha\supset\alpha}}$

分(甲) $11=11^*$ $\overline{\overline{(\alpha\supset(\beta\supset\gamma))\supset(\beta\supset(\alpha\supset\gamma))}}$

⋮

分(甲) $23=23^*$ $\overline{\overline{(\alpha\supset\bar{\beta})\supset(\beta\supset\bar{\alpha})}}$

24° 的双重否定也可在直觉系统中推出，其推导过程如下：

$25^*=(1)$ $((\bar{\bar{\alpha}}\supset\alpha)\supset\overline{\bar{\bar{\alpha}}\supset\alpha})\supset((\bar{\alpha}\supset\bar{\alpha})\supset\overline{\bar{\bar{\alpha}}\supset\alpha})$

分分(1)(甲) $10=24^*$ $\overline{\overline{\bar{\bar{\alpha}}\supset\alpha}}$

在直觉系统中还可推出

(乙) $\overline{\overline{\alpha\supset\beta}}\supset(\bar{\bar{\alpha}}\supset\bar{\bar{\beta}})$

其推导过程如下：

$10=(1)$ $(\alpha\supset\beta)\supset(\alpha\supset\beta)$

分 $11(1)=(2)$ $\alpha\cdot\supset((\alpha\supset\beta)\supset\cdot\beta)$

分 $\nabla^1 148(2)=(3)$ $\alpha\supset(\overline{\overline{\alpha\supset\beta}}\supset\overline{\overline{\beta}})$

分 $11(3)=(4)$ $\overline{\overline{\alpha\supset\beta}}\supset(\alpha\supset\overline{\overline{\beta}})$

分 $\nabla^1 23(4)=(5)$ $\overline{\overline{\alpha\supset\beta}}\supset(\bar{\beta}\supset\bar{\alpha})$

分 $\nabla^1 141(5)=$(乙) $\overline{\overline{\alpha\supset\beta}}\supset(\overline{\overline{\alpha}}\supset\overline{\overline{\beta}})$

所以在直觉系统中有导出规则

分分(乙)规则 $\overline{\overline{\alpha\supset\beta}},\ \overline{\overline{\alpha}}\vdash\overline{\overline{\beta}}$

记为“分*”规则.

设在古典系统中，α 可证. 必然在古典系统中有 α 的证明过程. 设其证明过程是

$$\beta_1,\beta_2,\cdots,\beta_n=\alpha$$

由定义知，诸 β_i 或为公理 10～24 之一，或由在前的 β_h 和 β_k 实施

"分"而得. 由 10～24 及分与 10*～24* 及分*的对应关系知,在直觉系统中必有一相应的证明过程

$$\overline{\overline{\beta_1}},\ \overline{\overline{\beta_2}},\cdots,\overline{\overline{\beta_n}}=\overline{\overline{\alpha}}$$

其中诸$\overline{\overline{\beta_i}}$或为 10*～24* 之一,或由在前的$\overline{\overline{\beta_k}}$和$\overline{\overline{\beta_{k'}}}$实施"分*"而得. 故在直觉系统中, $\overline{\overline{\alpha}}$可证. 本定理得证.

定理 $\bar{\alpha}$在古典系统中可证当且仅当$\bar{\alpha}$在直觉系统中可证.

[证] (⟸)显然成立.

(⟹) 设$\bar{\alpha}$在古典系统中可证. 由上一定理知 $\overline{\overline{\alpha}}$在直觉系统中可证. 因为在直觉系统中有

$$\overline{\overline{\alpha}}\supset\bar{\alpha}$$

所以在直觉系统中,$\bar{\alpha}$可证. 本定理得证.

由以上这些定理知道,若在古典系统中 α 可证,则在直觉系统中$\overline{\overline{\alpha}}$可证, 但 α 并不一定可证. 又因为由 α 可证或$\overline{\overline{\alpha}}$可证, 可以推出 $\bar{\alpha}$, $\overline{\overline{\alpha}}$, $\overline{\overline{\overline{\alpha}}}$等不可证(见本节 1.9.3), 所以可得如下的结论:

直觉系统和古典系统承认可证的公式虽不一样多, 但否认可证的公式却是一样的多.

1.9.2 具体的公理系统与形式的公理系统

在 §1.4 中给出了命题演算永真公式的公理系统. 该公理系统可以分成两部分来描述,一部分称为语法部分,另一部分称为语义部分, 或称为解释部分. 为了明确起见, 现在再具体地描述如下:

I. 组成部分

语法:

命题变元 1. p 是命题变元;

2. 如果 ξ 是命题变元,则 ξ' 也是命题变元;

3. 命题变元仅限于此.

联结词　$\neg, \vee, \wedge, \supset, \equiv$ 都是联结词.

括号　(,)是括号.

公式　1. 命题变元是公式;

2. 如果 α 是公式, 则 $\bar{\alpha}$ 也是公式; 如果 α, β 是公式, 则 $(\alpha \vee \beta)$, $(\alpha \wedge \beta)$, $(\alpha \supset \beta)$, $(\alpha \equiv \beta)$ 也是公式;

3. 公式仅限于此.

语义(即解释):

命题变元是以真假为变域的变元.

联结词是以真假为定义域以真假为值域的函数, 其中否定词"¬"是一元函数, 其它为二元函数. 具体含义是:

α	β	$\bar{\alpha}$	$\alpha \vee \beta$	$\alpha \wedge \beta$	$\alpha \supset \beta$	$\alpha \equiv \beta$
T	T	F	T	T	T	T
T	F	F	T	F	F	F
F	T	T	T	F	T	F
F	F	T	F	F	T	F

其中 T 表示真, F 表示假.

公式是由命题变元利用联结词组成, 称为命题演算公式. 公式有值, 其值是真或假. 公式的真假值计算法则是根据命题变元的实际值, 按指定的联结词进行计算, 括号内的公式先计算. 如果不论公式中的命题变元取何值, 公式均取得真(假)值, 则该公式称为永真(假)公式, 否则称为非永真(可满足)公式.

II. 推理部分

语法:

公理　10～24°为公理.

其中 α, β, γ 是任意的公式, 下同.

规则　分离规则是规则.

定理　1. 公理是定理;

2. 如果 $\alpha\supset\beta$ 和 α 都是定理，则由它们实施分离规则所得的 β 也是定理；

3. 定理仅限于此.

语义(即解释):

公理都是永真公式.

规则指明如何由旧的永真公式推出新的永真公式. 分离规则指明如果 $\alpha\supset\beta$ 和 α 都是永真公式则 β 也是永真公式.

定理都是永真公式，它们是由公理出发利用分离规则一步步推出来的公式.

我们之所以把这个公理系统称为命题演算永真公式公理系统是因为我们对这个公理系统作了如上的语义解释. 不难知道，系统中推出的一切定理都是永真公式. 而且可以证明（见本节的1.9.4)，所有永真公式都是本系统中的定理.

其实上述的语义解释并不是唯一的. 完全可以对上述语法成分作其它的解释. 例如，我们把命题变元解释成以 0 和 1 为变域的变元. 把联结词解释成以 0、1 为定义域以 0、1 为值域的代数运算符，具体为:

a	b	$\bar{a}$	$a\vee b$	$a\wedge b$	$a\supset b$	$a\equiv b$
0	0	1	0	0	0	0
0	1	1	0	1	1	1
1	0	0	0	1	0	1
1	1	0	1	1	0	0

公式有值，其值是 0 或 1. 公式的值的计算法则是根据变元的实际值按指定的代数运算符进行计算，括号内的公式先计算. 这样的公式称为布尔代数公式. 如果不论公式中的变元取何值，公式的值均为 0(1)，则该公式称为永为 0(1)的公式，否则称为可为 1(可为 0)的公式. 公理都是其值永为 0 的公式. 分离规则指明，

如果 $\alpha \supset \beta$ 和 α 都是值永为 0 的公式，则 β 也是值永为 0 的公式. 定理都是其值永为 0 的公式，它们是由公理出发利用分离规则一步步地推出来的公式.

按这种解释，上面的公理系统便是其值永为 0 的布尔代数公式的公理系统.

由此可知，公理系统有语法和语义两个方面. 同一语法定义可以有两个或多个不同的语义解释，并构成两个或多个具体的公理系统.

既有语法定义又有语义解释的公理系统称为*具体的公理系统*.

但是在讨论公理系统时，我们往往完全可以不管语义解释而只注意语法定义. 这种讨论也是很重要的，同样可以得到许多重要的结果. 通常把这种不涉及语义只涉及语法的公理系统称为*形式的公理系统*，简称为*形式系统*.

无论是具体的还是形式的公理系统都十分注意三大性质，不矛盾性、完备性和独立性. 尤其是前面两个性质更为重要. 下面分别叙述之.

1.9.3 不矛盾性(又称相容性和协调性)

不矛盾性问题也就是公理系统是否矛盾的问题. 通常对系统是否矛盾有多种不同的说法. 下面我们讨论三种不同的说法.

设 $\mathscr{A}$ 是一个公理系统，α 是 $\mathscr{A}$ 中任一公式.

第一种说法是，若 α 和 $\bar{\alpha}$ 至少有一个不是 $\mathscr{A}$ 中的定理，则说 $\mathscr{A}$ 是不矛盾的.

显然这种说法中要求 $\mathscr{A}$ 中包含有否定词，否则无法讨论它的不矛盾性. 有人把这种不矛盾性称为*古典不矛盾性*.

第二种说法是，若不是 $\mathscr{A}$ 中一切的公式都是 $\mathscr{A}$ 中的定理，则说 $\mathscr{A}$ 是不矛盾的.

因为一公式是不是$\mathscr{A}$中的定理仅与语法规则有关，与语义解释无关，所以有人把这种不矛盾性称为*语法不矛盾性*.

第三种说法是，如果$\mathscr{A}$中的定理都是永真公式，则说$\mathscr{A}$是不矛盾的.

因为这种说法涉到永真公式的概念，与语义解释有关，所以有人把这种不矛盾性称为*语义不矛盾性*.

下面我们来证明我们的公理系统既是语义不矛盾的，又是古典不矛盾的，又是语法不矛盾的.

可靠性定理 本命题演算公理系统是语义不矛盾的，即本系统中所有定理都是永真公式.

〔证〕 容易验证本系统中的 15 条公理都是永真公式，分离规则也保持永真性，即只要$\alpha\supset\beta$永真，α永真，则由它们实施分离规则而得的β也一定永真. 因此，根据数学归纳法，本系统中一切定理都是永真公式. 本定理得证.

可靠性定理 本公理系统是古典不矛盾的.

[证] 因为从语义角度看，对于任何公式α和$\bar{\alpha}$，α和$\bar{\alpha}$至少有一个为非永真公式，因此根据上面定理知，至少有一个不是本系统中的定理. 本定理得证.

可靠性定理 本公理系统是语法不矛盾的.

[证] 由上定理知，对于任何公式α，α和$\bar{\alpha}$中至少有一个不是定理，而这两个式子均是本系统中的公式，所以并非一切公式均是定理，即本系统是语法不矛盾的.

1.9.4 完备性(又称完全性)

和不矛盾性一样，完备性也有多种说法. 这里讨论相对完备性和绝对完备性两种.

一个公理系统，如果该系统是为某一具体理论而建立的，且该理论中所有的定理均能在该系统中推出，则说系统相对于该理论

是完备的.

对于一个公理系统,如果把系统中任何非定理的公式(以及与它同类型的一切公式)作为新公理加到该系统中所得的新的系统均为矛盾的系统,则说该系统是绝对完备的.

定理 本命题演算公理系统是相对完备的,即所有的永真公式均可在本系统中推出.

[证] 设 δ 是任一永真公式

$\tilde{\delta}$ 是 δ 的合析范式

利用 183°, 184°, 172°, 180°, 181°, 179 和替换规则不难证明在本系统中

$$\delta \equiv \tilde{\delta}$$

是可证的,即它是本系统中的定理.

设 $\tilde{\delta}=\delta_1 \wedge \delta_2 \wedge \cdots \wedge \delta_n$

$$\delta_i = x_{i1} \vee x_{i2} \vee \cdots \vee x_{im} \qquad (i=1, \cdots, n)$$

其中诸 x_{ij} 或是命题变元,或是命题变元的否定. 由前可知

δ 永真 当且仅当 $\tilde{\delta}$ 永真

当且仅当 诸 δ_i 永真

当且仅当 诸 δ_i 是虚析取式

即对于任何 i 均有 u, v 使得 δ_i 中的 x_{iu} 与 x_{iv} 互为否定,即

$$\bar{x}_{iu} = x_{iv}$$

利用 149°,公理 20、21和关于析取词的交换律结合律(即174, 176),容易证明在本系统中诸 δ_i 均可证,因而 $\tilde{\delta}$ 可证,因而 δ 可证. 本定理得证.

定理 本命题演算公理系统是绝对完备的.

[证] 设 δ 是本系统中任一非定理的公式.

现把该公式(以及与它同类型的一切公式)作为新公理加入本系统,所得的系统暂称为新系统. 我们将证明新系统是矛盾的.

因为本系统是相对完备的，而 δ 又不是本系统中的定理，所以 δ 不是永真公式，因而在 δ 的合析范式 $\tilde{\delta}$ 中必有合取项为实析取式.

设　　$\tilde{\delta}=\delta_1 \wedge \delta_2 \wedge \cdots \wedge \delta_n$

又设 δ_1 为实析取式，且

$$\delta_1 = x_1 \vee x_2 \vee \cdots \vee x_m$$

则有诸 x_i 或为命题变元式或为命题变元的否定，且有

$$\bar{x}_i \neq x_j \quad (j=1, \cdots, m)$$

因为在本系统中下列公式是可证的：

(1) $\delta \equiv \tilde{\delta}$

(2) $\tilde{\delta} \supset \delta_1$

所以在新系统中(1)，(2)也是可证的. 因为在新系统中 δ 是新公理，所以在新系统中 δ_1 可证. 因为在新系统中，凡与 δ 同类型的一切公式均是公理，所以在新系统中，凡与 δ_1 同类型的一切公式均可证. 今把 δ_1 中诸 x_i 作如下改动：

当 x_i 为命题变元时，该命题变元代之以 α（α 为任一公式）；

当 x_i 为命题变元的否定时，该命题变元代之以 $\bar{\alpha}$.

把这样改动后所得的公式记为 δ_1°. 易见 δ_1° 与 δ_1 同类型. 从而 δ_1° 可证.

因为　　$\delta_1^\circ = \alpha^\circ \vee \alpha^\circ \vee \cdots \vee \alpha^\circ$

其中 α° 或为 α 或为 $\bar{\alpha}$，所以由 δ_1° 极易推出 α 可证. 即在新系统中任何公式均可证. 由定义知，新系统是矛盾的. 本定理得证.

必须注意，一个公理系统的相对完备性是必要的. 因为既然要对某个理论建立公理系统，当然要求它能推出该理论内的一切定理. 但是绝对完备性不是必要的. 实际上我们需要研究公理系统的子系统，显然子系统不是绝对完备的.

1.9.5　独立性

独立性可分为简单独立性和完全独立性两种.

设有一个公理系统$\mathscr{A}$，又设α是$\mathscr{A}$中的一条公理. 如果由$\mathscr{A}$中的规则及$\mathscr{A}$中的其它公理不能推出α，则说α在$\mathscr{A}$中是简单独立的. 如果由$\mathscr{A}$中的规则及$\mathscr{A}$中的其它公理或这些公理的否定均不能推出α及其否定，则说α在$\mathscr{A}$中是完全独立的.

一个公理系统，如果该系统中所有的公理都是简单独立的，则说该公理系统是简单独立的. 如果该系统中所有的公理都是完全独立的，则说该公理系统是完全独立的.

具体地说，设$\mathscr{A}$是一个公理系统，$\mathscr{A}$中有三条公理α_1，α_2，α_3，两条规则R_1，R_2，记为

$$\mathscr{A}=(\{\alpha_1\alpha_{,2},\alpha_3\},\{R_1,R_2\}).$$

此外

$\mathscr{A}\vdash\beta$　表示　在$\mathscr{A}$中可推出公式β

$\mathscr{A}\overline{\vdash}\beta$　表示　在$\mathscr{A}$中推不出公式β

所谓α_1是$\mathscr{A}$中简单独立的公理是指

$$(\{\alpha_2,\alpha_3\},\{R_1,R_2\})\overline{\vdash}\alpha_1$$

所谓α_1是$\mathscr{A}$中完全独立的公理是指

$$(\{\bar{\alpha}_2,\alpha_3\},\{R_1,R_2\})\overline{\vdash}\alpha_1\text{ 或 }\bar{\alpha}_1$$

且　$$(\{\bar{\alpha}_2,\alpha_3\},\{R_1,R_2\})\overline{\vdash}\alpha_1\text{ 或 }\bar{\alpha}_1$$

且　$$(\{\bar{\alpha}_2,\bar{\alpha}_3\},\{R_1,R_2\})\overline{\vdash}\alpha_1\text{ 或 }\bar{\alpha}_1$$

且　$$(\{\bar{\alpha}_2,\bar{\alpha}_3\},\{R_1,R_2\})\overline{\vdash}\alpha_1\text{ 或 }\bar{\alpha}_1$$

显然，完全独立性的要求比简单独立性的要求强得多.

一个公理系统中各公理之间相互独立是很好的，但并不是必需的. 事实上本公理系统中公理 10 就不是简单独立的，因而本公理系统不是简单独立的，更不是完全独立的.

习　题

1. 证明本命题演算公理系统中的公理 10 不是简单独立的.

2. 设公理系统 $\mathscr{A}$, $\mathscr{B}$和$\mathscr{C}$是在本命题演算永真公式公理系统之上分别加入下列类型的公式(A), (B)和(C)后的新系统. 试证在这些新系统中公式 α 均为可证公式, 即 $\mathscr{A}$, $\mathscr{B}$和$\mathscr{C}$均是矛盾的.

(A) $\alpha \supset \beta$

(B) $((\alpha \wedge \beta) \supset (\bar{\alpha} \vee \gamma)) \supset \beta$

(C) $(\alpha \vee \beta) \supset (\alpha \supset \beta)$

其中 α、β 为任何公式.

第二章　谓词演算

§2.1　个体与谓词

在命题演算中，我们的讨论只是根据真值联结词把命题分析成原子命题.原子命题是不能利用真值联结词来再行分析的整体.然而原子命题还是可以作进一步分析的.本章中我们将把原子命题进一步分析成个体和谓词.

所谓个体是指可以独立存在的东西.它可以是具体的，也可以是抽象的.例如，小王，老张，3，4，××代表团等等.由个体组成的集合称为个体域.个体域中的个体个数可以有限也可以无限，例如{1，3，5，7}是四个个体组成的个体域，{小王，小张，小李}是三个个体组成的个体域，{1，2，3，…}是无限多个个体的个体域(通常称为自然数域).所有个体聚合在一起所组成的个体域称为全总个体域.以某个个体域 I 中的个体为变域的变元叫做个体域 I 上的个体变元.

所谓谓词是指个体、命题所具有的性质，或者若干个体、命题之间的关系.例如“小王是学生”，“5为质数”，“每天早晨做广播操是好习惯”，“5大于3”，“哥白尼指出地球绕着太阳转”，这些语句中的“是学生”，“为质数”，“是好习惯”，“大于”，“指出”都是谓词，前三个谓词是指明了个体或命题的性质的一元谓词，后二个谓词是指明二个个体或一个个体和一个命题之间的关系的二元谓词.从这几个例子中，可以知道，谓词实质上是一个函数，它以个体或命题为变域以命题为值.我们知道真值函数是以命题为变目，以命题为值的函数，故也是谓词，第一章中已经研究了五个真值联结

词，今后将对其余的谓词进行讨论，不过为了简便起见，我们将只讨论以个体为变域以命题为值的谓词.

以谓词为变域的变元叫做谓词变元.

为了使用方便，我们约定以 p, q, r 等表示命题或命题变元，以 a, b, c 等表示特定的个体，以 x, y, z 来表示个体变元，以 A, B, C 等表示特定的谓词，以 X, Y, Z 等表示谓词变元. x 具有性质 A 表为 Ax. x, y 间具有关系 B 表为 Bxy. x, y, z 间具有关系 C 表为 $Cxyz$ 等等.

孤零零一个谓词不是完整的语句，不表达任何完整的意思，必须填以个体后才是完整的语句，才有完整的意思. 例如，孤零零一个"是兄弟"就不是一个完整的语句，不表达任何完整意思. 因此谓词和填以个体后的谓词是二个完全不同的东西. 我们特别把谓词后填以个体所得的式子称为谓词填式. 谓词和谓词填式是两个截然不同的概念.

上面我们已约定用 A, B, C, X, Y, Z 等表示谓词，但这样有一个缺点，不能知道它们是几元的，为此我们在谓词后填以变元 e_1, e_2, e_3 等，以表示谓词的元数. 例如，若 A 是二元谓词，则表成 Ae_1e_2，这种 e 叫做命名变元，谓词后填以命名变元的式子叫做谓词命名式. 应该注意，谓词命名式与谓词是完全相同的概念，区别仅仅在于谓词没有明显指出元数，而谓词命名式明显指出了元数. 因此谓词命名式与谓词填式是截然不同的概念，不能混淆.

明确了这些概念后，就可以把许多日常语句写成逻辑式子.

例 1: 美国位于加拿大与拉丁美洲之间.

[解] 设 $Ae_1e_2e_3$ 表 e_1 位于 e_2 与 e_3 之间.

a 表美国，b 表加拿大，c 表拉丁美洲.

全句可译为: $Aabc$

例 2: 这位小弟弟抱住了那只大的红汽球.

[解] 设 Ae_1e_2 表 e_1 抱住了 e_2, Be 表 e 为小的, Ce 表 e 为弟弟, De 表 e 为大的, Ee 表 e 为红的, Fe 表 e 为汽球.

a 表这位, b 表那只.

全句可译为: $Ba \wedge Ca \wedge Db \wedge Eb \wedge Fb \wedge Aab$.

例 3: 如果你不出去, 我就不关灯.

[解] 设 Ae 表 e 出去, Be 表 e 关灯.

a 表你, b 表我.

全句可译为: $\bar{A}a \supset \bar{B}b$.

例 4: 金陵就是南京.

[解] 设 a 表金陵, b 表南京.

全句可译为: $a=b$.

应该注意, 谓词与个体域是密切相关的. 例如谓词"为质数"通常都是就自然数域而论的, 谓词"大于"通常都是就实数域而论的, 谓词"是学生"通常都是就人类这个个体域而论的. 以某个个体域 I 为定义域, 以真假为值域的谓词叫做个体域 I 上的谓词. 由此可知, 按下表定义的二元谓词 Ae_1e_2 是个体域 $\{a, b\}$ 上的谓词:

e_1	e_2	Ae_1e_2
a	a	T
a	b	F
b	a	F
b	b	T

显然, h 个个体组成的个体域 I 上的一元谓词共有 2^h 个, 二元谓词共有 2^{h^2} 个, m元谓词共有 2^{h^m} 个.

以个体域 I 上的谓词为变域的变元叫做个体域 I 上的谓词变元.

习　题

1. 试翻译下列各句为逻辑式子：

1.1　秦岭隔开黄河与汉水；

1.2　我送他这本书；

1.3　王华和李建一道去北京和天津；

1.4　泰山没有昆仑山高，喜马拉雅山更高；

1.5　哥伦布认为他已经到了印度；

1.6　这是书，不是练习本，而那本书是新的；

1.7　这本《矛盾论》是老张送给小王的；

1.8　他不知道你不在家，否则他就不去找你了。

2. 试列出个体域$\{1, 2, 3\}$上的全部一元谓词。

3. 试在个体域$\{1, 2, 3, 4\}$上用列表法定义下列谓词：

3.1　$e_1=e_2$；

3.2　e_1+e_2 为质数；

3.3　$e_1+e_2=e_3$。

§2.2　量词

如果只使用上节的概念和技巧，是无法把日常一切语句都表达清楚的。

例如，“9 或者大于 0，或者等于 0，或者小于 0”，当然可译为：$9>0 \vee 9=0 \vee 9<0$ 但是，“每一数或者大于 0，或者等于 0，或者小于 0”却不能仿上法把“每一数”看成为一个个体，而译为

$a>0 \vee a=0 \vee a<0$（a 表“每一数”）因为如果这样翻译，那么结果成为“每一数大于 0 或每一数等于 0 或每一数小于 0”，显然与原意大不相同。

又例如，“老张来了”，当然可译为

Aa　（a 表“老张”，Ae 表“e 来了”）

但是“一个中国人来了”却不能仿上法把“一个中国人”看成为一个

个体而译为

Ab （b 表“一个中国人”，Ae 表“e 来了”）

因为如果这样翻译，那么“老张是中国人”可译为“老张$=b$”，“小王是中国人”也可译为“小王$=b$”，这样岂不是推出“老张＝小王”来吗？其谬误是显然的.

诸如上面这两种语句必须另行处理，为此必须利用变元的概念.

例如对于“每一数或者大于 0 或者等于 0 或者小于 0”，如果约定变元 x 代表实数域中任意一个实数，则可译为

$$x>0 \vee x=0 \vee x<0$$

这个式子的意思是“对于实数域中任意一个实数 x，或者 $x>0$ 或者 $x=0$ 或者 $x<0$”这与原意一致.

凡表示“任意一个”的变元叫做全称性变元. 语句中出现“凡”、“一切”、“每个”、“任何”等词时，均可用全称性变元翻译.

对于“一个中国人来了”，如果假定变元 t 表示一个适当选定的中国人，Ae 表“e 来了”，则可译为 At. 这个式子的意思是“某一个中国人来了”，这与原意一致.

凡表示确定的但目前尚未知道的或不必明白指出的个体的变元叫做存在性变元. 语句中出现“某个”、“一个”、“某些”、“一些”等词时均可用存在性变元翻译.

由此可知，利用变元概念已能解决上面不能解决的问题. 但是还有很多不方便不完善的地方.

第一，各变元的变域（即个体域）须作临时约定；

第二，各变元的性质（全称性或存在性）须作临时约定.

下面对这两个问题作一些详细的讨论.

第一，关于个体域问题. 如上面的例子所示，引入变元前须约定变元在什么个体域中变化，如“x 代表实数域中任意一个实数”，

“t 代表一个适当选定的中国人”. 有时甚至可能出现同一个符号在不同的地方代表不同个体域中的变元，显然这个临时约定的办法很不好. 为此我们把个体域统一起来，使得一切变元均是同一个体域中的变元，从而无须临时约定. 其办法是规定一切变元(全称性的，存在性的)都以全总个体域为其变域.

采用这种办法后，特殊个体域中的变元应该怎样表示呢？例如，设 x 是以实数为变域的全称性变元，并有

$$x>0\vee x=0\vee x<0$$

如果改用全总个体域为变域的全称性变元 y，可引入谓词 Ae，表“e 为实数”，并把上式表为

$$Ay\supset(y>0\vee y=0\vee y<0)$$

此式的含义是“对任何个体 y，只要 y 为实数便有或者 $y>0$ 或者 $y=0$ 或者 $y<0$。”易见，这句话与原意完全一样.

对于“一个中国人来了”可改译如下

$Bu\wedge Au$(Be 表“e 为中国人”，Ae 表“e 来了”，u 表示以全总个体域为变域的存在性变元)

此式的含义即“有一个适当的个体 u，u 为中国人并且 u 来了。”易见，这句话也与原意一样.

从这两个例子容易推得：

要把全称性变元的变域由特殊个体域改为全总个体域，只须在全句前加以蕴函前件，以限定其变域.

要把存在性变元的变域由特殊个体域改为全总个体域，只须在全句前加以合取项，以限定其变域.

第二，关于变元的性质问题. 如上所述，引入变元除须约定变元的变域外，还须约定变元是全称性的还是存在性的，若不事先约定清楚，结果式的意义就不明确. 这种事先约定实在不方便. 此外，如仅用这两种变元还有很多句子不大容易翻译. 例如，“并非

一切数均大于 0 "既不能译为"$\overline{x>0}$", (x 为实数域中的变元), 也不能译为 "$Ax\supset\overline{x>0}$"(Ae 表"e 为实数", x 为全总个体域中的变元), 因为这两种式子的意思均为"对于任何实数 x, x 均不大于 0", 显然与原意不同.

因此我们引进下列符号:

$\forall x$ 读为对一切 x 均使得…

$\exists x$ 读为有一个 x 使得…

有了这两个符号, 上面三个例子可如下翻译:

第一例为

$$\forall x(Ax\supset(x>0\vee x=0\vee x<0))$$

这个式子的意思即为"对一切 x, 均使得只要 x 为实数, 便或 $x>0$ 或 $x=0$ 或 $x<0$", 这句话与原意相同.

第二例为

$$\exists x(Bx\wedge Ax)$$

其意为"有一个 x 使得 x 为中国人且 x 来了", 也与原意相同.

第三例为

$$\overline{\forall x}(Ax\supset x>0)$$

其意为"并非对一切 x 均使得若 x 为实数, x 就大于 0", 也与原意相同.

这样, 我们就有下列两种公式: $\forall x\alpha(x)$, $\exists x\alpha(x)$. 对于这两种公式有下列几个概念, $\forall$, $\exists$ 分别称为全称量词和存在量词. $\forall x$ 和 $\exists x$ 中的 x 称为相应量词的指导变元, 而 $\alpha(x)$称为相应量词的作用域, 在作用域中不与指导变元同名的其它变元一律叫做参数.

有了量词的概念后, 大部分语句都能表达成逻辑公式了.

例 1: 对于任何数, 均有一数比它大.

[解] 设 Ae 表 e 为数, 则全句可译为

$$\forall x(Ax\supset(\exists y(Ay\wedge y>x)))$$

例 2: 有一数比任何数都大.

[**解**] 设 Ae 表 e 为数，则全句可译为

$$\exists x(Ax \wedge \forall y(Ay \supset x > y))$$

例 3: 并非一切劳动都能用机器代替.

[**解**] 设 Ae 表 e 为劳动，Be 表 e 为机器，Ce_1e_2 表 e_1 能被 e_2 代替，则全句可译为

$$\overline{\forall} x(Ax \supset \exists y(By \wedge Cxy))$$

例 4: 没有不犯错误的人.

[**解一**] 把"犯错误"看作一谓词.

设 Ae 表 e 为人，Be 表 e 犯错误，则全句可译为

$$\overline{\exists} x(Ax \wedge \bar{B}x)$$

[**解二**] 把"错误"看作一谓词.

设 Ae 表 e 为人，Be 表 e 为错误，Ce_1e_2 表 e_1 犯 e_2，则全句可译为

$$\overline{\exists} x(Ax \wedge \forall y(By \supset \bar{C}xy))$$

习　题

把下列各句译为逻辑公式:

1. 群山之中喜马拉雅山最高;
2. 任何金属均溶于某种液体中;
3. 有一液体可溶任何金属;
4. $f(x)$在区间(a, b)内每点均连续(根据其数学定义翻译);
5. $f(x)$在区间(a, b)内一致连续(根据其数学定义翻译);
6. 除 2 以外的所有质数都是奇数;
7. 并非所有的质数都是奇数;
8. 有不是奇数的质数.

§2.3 自由变元与约束变元

由命题变元和谓词填式利用真值联结词和量词如下作成的式子称为谓词演算公式(本章中简称为公式)

(i) 命题变元是公式;

(ii) 填以个体变元的谓词变元填式是公式;

(iii) 如果 α 是公式, 则 $\bar{\alpha}$ 也是公式;

(iv) 如果 α 和 β 是公式, 则$(\alpha\vee\beta)$, $(\alpha\wedge\beta)$, $(\alpha\supset\beta)$和$(\alpha\equiv\beta)$也都是公式;

(v) 如果 α 是公式, x 是个体变元, 则$(\forall x\alpha)$, $(\exists x\alpha)$也都是公式;

(vi) 公式仅限于此.

设 α 为一谓词演算公式, $Qx\beta$(其中 Q 或为 $\forall$ 或为 $\exists$, β 为公式)为 α 的子公式, 则该 $Qx\beta$ 中变元 x 的一切出现都叫做 x 在 α 中的约束出现, α 中 x 的除约束出现外的一切出现都叫做 x 在 α 中的自由出现. 例如:

(1) $$\forall y((Xxy\wedge\exists x\bar{Y}x)\supset\exists x(Xxy\equiv\exists xYx))$$

在该式中, 除 x 的第一个出现是自由出现外, 其余的个体变元的出现均是约束出现.

设有一量词它以 x 为指导变元. 变元 x 在该量词的作用域中的一切自由出现以及该指导变元都叫做受该量词约束. 在(1)中, 个体变元 y 的一切出现均受(1)中开头的全称量词的约束, 变元 x 的第二和第三出现受第一个存在量词的约束, 变元 x 的第四和第五出现受第二个存在量词的约束, 变元 x 的第六和第七出现受第三个存在量词的约束, 而变元 x 的第一出现不受任何量词约束, 是(1)的自由出现。公式中各个量词与各个变元的出现之间这种约束上的关系称为公式的约束关系。确定公式中哪个变元的哪些出

现受哪个量词的约束的过程叫做确定约束关系的过程。

如果变元 x 在 α 中有自由出现，则说 x 是 α 的自由变元，如果 x 在 α 中有约束出现，则说 x 是 α 的约束变元。因为同一变元在一公式中可能既有自由出现，又有约束出现，所以同一变元在一公式中可能既是自由变元又是约束变元。(1)中的 x 就是如此。

量词的主要用途在于引入约束变元。量词的指导变元是个体变元，作用域是公式(取真假值)，值域是真假。命题变元和谓词变元能否作为量词的指导变元呢？能够的．但本书中不讨论这方面的问题，因而在本书中将不涉及约束命题变元和约束谓词变元的概念．今后公式中的命题变元和谓词变元均是自由变元．

现在我们讨论改名和代入．

改名和代入都是对变元而言的．所谓改名就是把一变元改为另一变元，并要求改名后的式子(结果式)与原式意义相同，即同真假．所谓代入就是把一变元代以式子（式子的值的变域应与变元的变域相同)，并要求代入后的式子(结果式)为原式的特例．

先看改名．

第一，改名是对约束变元而言，不是对自由变元而言的，即可以对约束变元施行改名，不能对自由变元施行改名．例如：

(2) $\qquad \forall x(x=y \vee x \neq y)$

该式中的约束变元 x 可改名为 u、v、w 等．结果式分别为：

$$\forall u(u=y \vee u \neq y),\ \forall v(v=y \vee v \neq y),\ \forall w(w=y \vee w \neq y)$$

等．显然这些式子的意义与原式意义相同，都表示“任何个体都或者等于 y 或者不等于 y”．

第二，改名必须“处处”进行．所谓“处处”进行的含义是当对公式中受某个量词约束的变元改名时，必须对原式中该变元的一切受该量词约束的约束出现均改名，否则改名后将改变原式的约束关系，改变原式的意义．例如若把(2)中 x 的第一第二出现改为

u, 第三出现不改, 则有

$$\forall u(u=y\vee x\neq y)$$

其意是"任何个体 u 都或者等于 y 或者 x 不等于 y", 与原意不同. 其错误的根源在于改名后的结果式的约束关系与原式不同了.

第三, 对受某个量词约束的变元改名时新名决不能与该量词的作用域中的其它自由变元同名, 否则改名后将改变原式的约束关系, 改变原式的意义. 例如若把(2)中 x 改名为 y, 则有

$$\forall y(y=y\vee y\neq y)$$

其意是"任何个体都或者与自己相等或者与自己不等", 这与原意不同.

第四, 对受某个量词约束的变元改名时新名能否与该量词的作用域中的约束变元同名呢?回答是有时行有时不行. 例如:

(3) $$\forall x(x^2\geqslant 0\wedge\forall y(x=y\wedge\exists z(z>y)))$$

该式中的约束变元 x 不能改名为 y(y 是作用域中的约束变元), 但可改名为 z (z 也是作用域中的约束变元), 因为 (3) 式的意思是: "对于任何个体, 其平方必$\geqslant$0, 并且对于任何新个体, 原个体都与新个体相等, 并且存在一个个体, 它大于该新个体". 而改名为 y 后的结果式为

$$\forall y(y^2\geqslant 0\wedge\forall y(y=y\wedge\exists z(z>y)))$$

其意思是"对于任何个体, 其平方必$\geqslant$0, 并且对于任何新个体, 新个体必与自己相等并且存在一个个体, 它大于该新个体", 显然它们的意思不相同. 从约束关系看, 结果式的约束关系显然与原来的约束关系不相同. 但是

$$\forall z(z^2\geqslant 0\wedge\forall y(z=y\wedge\exists z(z=y)))$$

意思与原式相同. 从约束关系看, 结果式的约束关系与原来一样. 是否不改变原约束关系的改名都是正确的改名呢?确实如此. 例如在上面这个例子中, 如果先将原式中的约束变元 y 改名为其它变

元，譬如改名为 t，再将约束变元 x 改名为 y，结果式为

$$\forall y(y^2\geqslant 0\wedge \forall t(y=t\wedge \exists z(z>t)))$$

显然这时约束关系完全相同．结果式的意思也与原式相同，所以是正确的改名．

第五，由上可知，正确改名的要求是：改名前与改名后的约束关系不发生变化．

为了确保改名的正确，我们规定，改名时一定要改成其作用域中所没有的变元．显然，按此规定施行改名，结果一定正确．而且一切正确的改名均可按此规定得到．

再看代入．

第一，代入是对自由变元而言的，就是说对自由变元可施行代入．

第二，应注意代入也是有条件的，不能盲目地代入，否则将发生错误．例如，$\forall x(x=y\cdot z)$ 中的自由变元 y 可代以 2，也可代以 z，也可代以 $y+z$，也可代以 $\sin t$ 等等，因为代入后所得的结果式 $\forall x(x=2z)$，$\forall x(x=z\cdot z)$，$\forall x(x=(y+z)\cdot z)$，$\forall x(x=(\sin t)\cdot z)$ 均是原式的特例，约束关系没有变化．但是自由变元 y 决不能代以 x，也不能代以含 x 的式子，如 $x+t$，$\sin x$ 等等，因为代入后所得的结果式

$$\forall x(x=x\cdot z),\ \forall x(x=(x+t)\cdot z),\ \forall x(x=(\sin x)\cdot z)$$

均不是原式的特例，这时的约束关系已与原来的不同了．如果一定要代以 x，或代以含 x 的项，必须先把原式中约束变元改名，使之与代入项无关，譬如改为 u，而后施行代入．因为这时的结果式为

$$\forall u(u=x\cdot z),\ \forall u(u=(x+t)\cdot z),\ \forall u(u=(\sin x)\cdot z)$$

均为原式的特例，约束关系已与原式一致，由此可知，对于自由个体变元，正确代入的条件是代入前后的约束关系应保持不变．

第三，代入必须“处处”进行．所谓“处处”进行是指当对某自由变元施行代入时，必须对该公式中该变元的一切自由出现均施行代入，否则将是错误的．

第四，为了确保对自由个体变元的正确代入，我们规定，代入前先对原式施行改名，使得原式中所有约束变元名与代入式中所有变元名互不相同，然后再施行代入．显然，按此规定施行代入，结果一定正确．而且一切正确的关于自由个体变元的代入均可按此规定得到．

第五，对于命题变元和谓词变元也可施行代入，而且代入也是有条件的，其条件仍就是代入前后约束关系必须保持不变．在举例说明这个问题之前，有一点需要说明，如上所见，个体变元的代入式是项（参见 §2.8.4），而命题的代入式应为公式，谓词变元的代入式应为谓词，而且代入式的元数应与原谓词相同．下面我们举例说明对命题变元和谓词变元的代入．

例1：试在 $\forall y(p\supset Ay)\supset(p\supset\forall xAx)$ 中把 p 代以 $\exists yBxy$，和把 p 代以 $\exists xBxy$．

［解］ 因为对 p 用 $\exists yBxy$ 代以后约束关系不发生变化，所以可立即施行代入得

$$\forall y(\exists yBxy\supset Ay)\supset(\exists yBxy\supset\forall xAx)$$

对于用 $\exists xBxy$ 代 p 的情形，如盲目代入的话，约束关系就将发生变化，从而出现错误，所以必须先将原式中的约束变元 y 改名，使之与代入式中的变元无关，如将 y 改为 t 得

$$\forall t(\exists xBxy\supset At)\supset(\exists xBxy\supset\forall xAx)$$

这个结果式显然是原式的特例，故代入正确．

为了确保对命题变元的代入正确，我们规定，代入前先对原式施行改名，使得式中所有约束元名与代入式中所有变元名均互不相同，然后施行代入．显然，按此规定施行代入，结果一定正确．

而且一切正确的关于命题变元的代入均可按此规定得到.

谓词变元的代入比较复杂. 因为在公式中谓词变元均是以谓词填式的形式出现, 因此在代入前应先将代入谓词变成相应的填式, 然后再把代入好的填式代到原式中去. 因此欲使整个代入正确, 必须保证二次代入均正确, 也即二次代入过程中的约束关系均不应发生变化.

例 2: 试在 $\forall x(Xxy \supset Yx) \supset \exists y Xty$ 中, 把 Xe_1e_2 代以 $\forall t\alpha(e_1, e_2, x, t)$.

[解] 因为代入式 $\forall t\alpha(e_1, e_2, x, t)$ 中 x 是自由变元, t 是约束变元, 而原式中谓词填式 Xxy 中 x 是约束变元. 所以若盲目代入的话, 必然发生变元混乱, 产生错误. 为此必须先对原式及代入式进行改名, 使之相互间的变元均不同名, 然后施行代入.

先把代入式改名为

$$\forall u\alpha(e_1, e_2, x, u)$$

再把原式改名为

$$\forall z(Xzy \supset Yz) \supset \exists y Xty$$

再作关于 Xzy 的代入式的填式

$$\forall u\alpha(z, y, x, u)$$

再作关于 Xty 的代入式的填式

$$\forall u\alpha(t, y, x, u)$$

最后将这二个填式代到原式去得

$$\forall z(\forall u\alpha(z, y, x, u) \supset Yz) \supset \exists y \forall u\alpha(t, y, x, u)$$

同样, 为了确保对谓词变元的代入正确, 我们规定, 在代入前, 先对原式施行改名, 使得原式中的所有约束变元名与代入式中所有变元名互不相同, 然后施行二次代入, 显然按此规定施行代入, 结果一定正确. 而且一切正确的对于谓词变元的代入均可按此规定得到.

习　题

1. 指出下列公式中的自由变元，约束变元以及约束关系：

1.1　$\forall x(x=y+x)\supset y\leqslant x$；

1.2　$((x=x)\equiv(x<y))\supset(\forall x(x=x)\equiv\forall x(x<y))$；

1.3　$\forall x\exists y((x=y)\supset\exists x(\forall y((x=y)\wedge(x<y)\vee\exists x(x<y))))$.

2. 指出下列公式中各量词不能改用什么变元作指导变元，并对各式施行改名：

2.1　$\forall x(Xx\supset Yxt)\supset\exists y(Xy\wedge Yxy)$；

2.2　$\forall x\exists yXxyt\supset\exists y\forall xXxyt$；

2.3　$\forall x(\exists y(\forall tXxt\supset Xyt)\wedge Xxy)$.

3. 试对下列公式施行代入：

3.1　在 $x=y^2+3\wedge\exists y(y\neq x)$ 中，把 x 代以

(a) y,　　(b) $y+x^2$；

3.2　在 $\forall y\forall z(\forall x(Xyz\supset(P\wedge Yxy))\supset(P\wedge Xyz))$ 中

(a) P 代以 $\forall yXyz$, Xe_1e_2 代以 $\forall z(Xe_1z\supset Ye_1e_2)$；

(b) P 代以 Xyz, Xe_1e_2 代以 $Ye_1z\equiv\forall yXye_2$.

§2.4　永真性与可满足性

要讨论谓词演算公式永真与否，首先应该弄清任一谓词演算公式的真假与公式中哪些成分有关，我们用具体例子来分析这个问题.

先讨论 Ax, $\forall xAx$, $\exists xAx$ 这三个公式的真假情况.

设常谓词"Ae"表示"e 为偶数"，则 Ax 意为"x 为偶数". 显然真假值随 x 而变化，也即 Ax 的真假与自由个体变元 x 有关. 公式 $\forall xAx$ 和 $\exists xAx$ 的意思分别为"所有 x 均为偶数"，"有一个个体为偶数". 显然它们的真假与个体域有关，而与约束个体变元无关，因为一当个体变元的变域——个体域给定后，它们的真假就确定了. 例如，对个体域$\{1, 2, 3, \cdots\}$，$\forall xAx$ 的意为"所有的自然数均为

偶数”，其值为 F；$\exists xAx$ 的意为“有一自然数为偶数”，其值为 T. 对于个体域$\{1,3,5,7,9\}$，$\forall xAx$ 的意为“1, 3, 5, 7, 9 这五个自然数均为偶数”，其值为 F；$\exists xAx$ 的意为“1, 3, 5, 7, 9 中有一个为偶数”，其值为 F.

由此可知，一公式的真假与个体域有关，与自由个体变元有关，而与约束个体变元无关.

再进一步讨论下面的例子，$\forall xXx \wedge Yy \wedge p$ 当个体域 I 指派以实数域，谓词变元 Xe 指派以常谓词“e 平方$\geqslant 0$”，谓词变元 Ye 指派以常谓词“e 的绝对值$=1$”，命题变元 p 指派以 T 时，原式的意思便为“所有的实数，其平方均$\geqslant 0$，并且 y 的绝对值$=1$，并且 T”也即为“$|y|=1$”，显然这时的值与自由个体变元有关. 若把个体域改为复数域其余均不变，原式的意思变为“所有复数，其平方均$\geqslant 0$，并且$|y|=1$，并且 T”，显然这两个语句的真假不同. 所以该公式的真假与个体域有关. 又若把谓词变元 Xe 改指派以“e 的平方<0”，其余仍不变，原式的意思为“所有的实数其平方均<0，并且$|y|=1$，并且 T”，显然与第一种意思不同，所以该公式的真假与谓词变元 Xe 有关. 同样可知，该公式的真假与谓词变元 Ye 有关，与命题变元 p 有关.

综上所述可得，任一谓词演算公式的真假与谓词变元、命题变元、自由个体变元以及个体域有关，而与约束个体变元无关.

设有一谓词演算公式 α，其自由个体变元为 $x_1, \cdots, x_h$；命题变元为 $p_1 \cdots, p_k$；谓词变元为 $X_1 \cdots, X_r$，则我们将把 α 表为

$$\alpha(x_1, \cdots, x_h; p_1, \cdots, p_k; X_1, \cdots X_r)$$

(注意，这不是谓词填式，因为 α 不是谓词，而是公式)如果对个体域 I 指派以 I°，(即其约束变元以 I° 为变域)对 $x_1, \cdots, x_h$ 分别指派以 I° 中的个体 $a_1, \cdots, a_h$；对 $p_1, \cdots, p_k$ 分别指派以 $p_1^\circ, \cdots, p_k^\circ$；对 $X_1, \cdots, X_r$ 分别指派以 I° 上的谓词 $A_1, \cdots, A_r$，则说对 α 作了一个

个体域 I° 中的指派$(a_1, \cdots, a_k; p_1^\circ, \cdots, p_k^\circ; A_1, \cdots, A_r)$.

因为 α 的值与个体域、自由变元、命题变元、谓词变元有关，所以给定一指派后，α 的真假值是确定的. 为了讨论方便起见，今后我们将把 α 在 I° 中的指派$(a_1, \cdots, a_k;\ p_1^\circ, \cdots, p_k^\circ; A_1, \cdots, A_r)$之下所取值记为

$$\alpha(a_1, \cdots, a_k; p_1^\circ, \cdots, p_k^\circ; A_1, \cdots, A_r)$$

如果该值为真，则该指派称为 α 的**成真指派**，如果该值为假则该指派称为 α 的**成假指派**.

如果仅对 α 中的部分自由变元（包括自由个体变元、命题变元、谓词变元）给以指派，则称该指派为 α 的**有缺指派**. 一般说，α 在有缺指派下，不一定能得出确定的真假值，而是关于那些未作指派的自由变元的函数.

决定谓词演算公式的真假，关键在于决定 $\forall x\alpha x$，$\exists x\alpha x$ 的真假. $\forall x\alpha x$，$\exists x\alpha x$ 的真假的决定方法如下：对于个体域 I，

$\forall x\alpha x$ 真，当且仅当 I 中各个体均使得 αx 真；$\exists x\alpha x$ 真，当且仅当 I 中有一个体使得 αx 真.

下面我们举例说明，给定一指派之后，如何确定公式 α 在该指派之下所取的值.

例：求 $\exists x\forall y((Xxz \wedge Xyz \wedge p) \supset \forall u(Xxu \supset Xyu))$ 在$(I,\ z,\ p,\ Xe_1e_2) =$（自然数域，$2, T, e_1 < e_2$）之下的值.

[**解**] 第一步，将指派代入，化简得

$$\exists x\forall y((x<2 \wedge y<2 \wedge T) \supset \forall u(x<u \supset y<u))$$
$$= \exists x\forall y((x<2 \wedge y<2) \supset \forall u(x<u \supset y<u))$$

第二步，自内向外逐层求出由量词带头的子公式的值，这里，先求 $\forall u(x<u \supset y<u)$ 的值（一般说，该值与自由变元 x, y 有关），再求出整个公式的值.

在求子公式或整个公式的值时，应对约束变元作各种可能的

代入，并求出作用域在各种情况下的真假，但是必须注意，作各种的可能的代入时，应该是有次序的，不能乱来．否则，公式的真假仍不能决定，次序是什么呢？是公式头上指导变元的次序．对于我们这个例子，子公式 $\forall u(x<u\supset y<u)$ 只有一个指导变元，故不存在次序问题，只须对约束变元的各种可能情形作代入，求值即可；但整个公式的头上有二个指导变元 x 和 y，因此存在一个次序问题，其次序就是先 x 后 y，即按 x 来分各种情形，再按 y 来分各种情形，这样就可求出作用域的真假了．

下面我们先求 $\forall u(x<u\supset y<u)$ 的真假情形．

① $x<u$ 时

$$\begin{aligned}\text{作用域}&=x<u\supset y<u\\&=T\supset y<u\\&=y<u\end{aligned}$$

$y<u$ 时，作用域 $=T$

$y\geqslant u$ 时，作用域 $=F$

② $x\geqslant u$ 时

$$\begin{aligned}\text{作用域}&=x<u\supset y<u\\&=F\supset y<u\\&=T\end{aligned}$$

故知，当 $x<u\leqslant y$ 时，作用域 $=F$

此外情形时，作用域 $=T$

因此，当 $x<y$ 时，存在 u 使得作用域 $=F$，所以

$$\forall u(x<u\supset y<u)=F$$

当 $x\geqslant y$ 时，不存在 u 使得 $x<u\leqslant y$ 成立，所以

$$\forall u(x<u\supset y<u)=T$$

故得

$$\forall u(x<u\supset y<u)=x\geqslant y$$

将此结果代入到原式中去，得

$$\exists x\forall y((x<2\wedge y<2)\supset x\geqslant y)$$

再求该公式的真假.

(i) 当 $x\geqslant 2$ 时, 作用域 $=(x<2\wedge y<2)\supset x\geqslant y$

$=(F\wedge y<2)\supset x\geqslant y$

$=T$

(ii) 当 $x<2$ 时

作用域 $=(x<2\wedge y<2)\supset x\geqslant y$

$=(T\wedge y<2)\supset x\geqslant y$

$=y<2\supset x\geqslant y$

(2.1) 当 $x=1$ 时, 作用域 $=y<2\supset 1\geqslant y$

$=T$

(2.2) 当 $x=0$ 时, 作用域 $=y<2\supset 0\geqslant y$

(2.2.1)当 $y=0$ 时, 作用域 $=0<2\supset 0\geqslant 0=T$

(2.2.2)当 $y=1$ 时, 作用域 $=1<2\supset 0\geqslant 1=F$

(2.2.3)当 $y\geqslant 2$ 时, 作用域 $=F\supset 0\geqslant y=T$

由此可列出下表:

x	y	作用域	$\forall y$	$\exists x$
$x\geqslant 2$	全	T	T	T
$x=1$	全	T	T	
$x=0$	$y=0$	T	F	
	$y=1$	F		
	$y\geqslant 2$	T		

由此可知, 在所给的指派之下, 该公式取得真值.

现在我们可以来讨论同真假性, 永真性和可满足性了.

设有公式 α 及 β, 如果对个体域 I 中每一指派, α 与 β 均取得相同的真假值, 则说 α 与 β 在 I 上同真假, 如果 α 与 β 在每一个非

空域上均同真假，则说 α 与 β 同真假.

定理 如果 α 与 β 同真假，则 $\varphi(\alpha)$ 与 $\varphi(\beta)$ 同真假，其中 $\varphi(\alpha)$ 是任一含 α 的公式，$\varphi(\beta)$ 是在 $\varphi(\alpha)$ 中用 β 替换若干个 α 的结果。

定理 在任何域中下列各对公式均同真假，其中 γ 是不含自由变元 x 的公式.

(1) $\overline{\forall}x\alpha(x)$ 与 $\exists x\bar{\alpha}(x)$

(2) $\overline{\exists}x\alpha(x)$ 与 $\forall x\bar{\alpha}(x)$

(3) $\forall x(\alpha(x)\wedge\gamma)$ 与 $\forall x\alpha(x)\wedge\gamma$

(4) $\forall x(\alpha(x)\vee\gamma)$ 与 $\forall x\alpha(x)\vee\gamma$

(5) $\exists x(\alpha(x)\wedge\gamma)$ 与 $\exists x\alpha(x)\wedge\gamma$

(6) $\exists x(\alpha(x)\vee\gamma)$ 与 $\exists x\alpha(x)\vee\gamma$

请读者自行证明这两条定理.

利用这两个定理可以证明下面很重要的定理，为此先给出一个定义.

定义 一公式如果其量词均在全式的开头，它们的作用域均延伸到整个公式的末尾，则该公式叫做*前束形公式*；由前束形的公式利用其真值联结词作成的公式叫作*准前束形公式*.

前束范式定理 任意一个谓词演算公式均和一个具有前束形的公式同真假(该前束形公式称为原公式的前束范式).

[证] 首先利用(1)和(2)两组同真假式，把否定深入到命题变元和谓词填式的前面.

其次，利用(3),(4),(5)和(6)四组同真假式，把量词移到全式的最前面，这样便得到前束形公式，定理得证.

具体求法见下例:

例: 试求 $\overline{\forall}x(\exists yXxy\supset\exists x\forall y(Yxy\wedge\vee y(Xyx\supset Yxy)))$ 的前束范式.

[解] 第一步: 否定深入.

原式 $=\overline{\forall x}(\overline{\exists y}Xxy \vee \exists x\forall y(Yxy \wedge \forall y(Xyx \supset Yxy)))$

(化去$\supset$)

$=\exists x(\exists yXxy \wedge \overline{\exists x}\forall y(Yxy \wedge \forall y(Xyx \supset Yxy)))$

(根据(1))

$=\exists x(\exists yXxy \wedge \forall x\exists y(\overline{Yxy} \vee \exists y(\overline{Xyx \supset Yxy})))$

(根据(1)(2))

第二步: 改名(以便为根据(3)、(4)、(5)、(6)把量词提到前面作准备).

上式 $=\exists x(\exists yXxy \wedge \forall u\exists v(\overline{Yuv} \vee \exists z(\overline{Xzu \supset Yuz})))$

第三步: 利用(3)、(4)、(5)、(6)把上式化为前束范式.

上式 $=\exists x\exists y(Xxy \wedge \forall u\exists v(\overline{Yuv} \vee \exists z(\overline{Xzu \supset Yuz})))$

(根据(5))

$=\exists x\exists y\forall u\exists v(Xxy \wedge (\overline{Yuv} \vee \exists z(\overline{Xzu \supset Yuz})))$

(根据(4),(5))

$=\exists x\exists y\forall u\exists v\exists z(Xxy \wedge (\overline{Yuv} \vee \overline{Xzu \supset Yuz}))$

(根据(6)(4))

如果一公式 α 对域 I 中任何指派均取得真值, 则说 α 在 I 中永真, 如果均取得假值, 则说 α 在 I 中永假, 如果至少有一个指派取得真值, 则说 α 在 I 中可满足, 如果至少有一个指派取得假值, 则说 α 在 I 中非永真.

如果 α 在每个非空域中均永真(永假), 则说 α 永真(永假), 如果 α 在某个非空域中可满足(非永真), 则说 α 可满足(非永真).

由此定义易得下面的定理.

定理 若 α 永真, 则 α 在 I 中永真; 若 α 在 I 中可满足, 则 α 可满足.

和命题演算相仿, 有下列定理成立.

定理 α 在 I 中永真当且仅当 $\bar{\alpha}$ 在 I 中不可满足(即永假). α 在 I 中可满足当且仅当 $\bar{\alpha}$ 在 I 中非永真. α 永真当且仅当 $\bar{\alpha}$ 不可满足. α 可满足当且仅当 $\bar{\alpha}$ 非永真.

定理 α 与 β 同真假当且仅当 $\alpha\equiv\beta$ 永真. $\alpha\wedge\beta$ 永真当且仅当 α 和 β 均永真. $\alpha\vee\beta$ 永假当且仅当 α 和 β 均永假.

除掉这些与命题演算相仿的定理外, 还有一些命题演算中不具备的性质.

定理 如果 I, J 为具有相同个数的个体域(个体可不相同), 则任一公式 α, α 在 I 中永真当且仅当 α 在 J 中永真, α 在 I 中可满足当且仅当 α 在 J 中可满足.

[**证**] 因为 I、J 具有相同个数个体, 所以可在它们之间建立一一对应. 设 I 中的个体 a 与 J 中的个体 a' 一一对应.

现在把 I 上的谓词作如下对应: 设 $Ae_1e_2\cdots e_n$ 为 I 上的 n 元谓词, 则令满足下列性质的 J 中 n 元谓词 $A'e_1\cdots e_n$ 为其对应谓词.

$Aa_1\cdots a_n$ 真 当且仅当 $A'a_1'\cdots a_n'$ 真

其中各 a 在 I 中取值, 各 a' 在 J 中取值.

再把 I 中指派与 J 中指派作如下对应: 设有一个 I 中指派

$$(x_1,\cdots,x_h;p_1,\cdots,p_k;X_1,\cdots,X_r)$$
$$=(a_1,\cdots,a_h;p_1^\circ,\cdots,p_k^\circ;A_1,\cdots,A_r)$$

省记为 $(x;p;X)=(a;p^\circ;A)$

则命 J 中的下列指派为其对应指派

$$(a_1',\cdots,a_h';p_1^\circ,\cdots,p_k^\circ;A_1',\cdots,A_r')$$

省记为 $(a';p^\circ;A')$

利用归纳法可证

(甲) $$\alpha(a;p^\circ;A)=\alpha(a';p^\circ;A')$$

如果 α 为命题变元, 显然(甲)成立.

如果 α 为谓词填式 $X(x_{11},\cdots,x_{1n})$, 则有 $\alpha(a;p^\circ;A)=A(a_{11},$

$\cdots, a_{1n}) = A'(a'_{11}, \cdots, a'_{1n}) = \alpha(a'; p^\circ; A')$, 故(甲)成立.

如果 α 呈下面五种形式之一: $\bar{\beta}$, $\beta_1 \vee \beta_2$, $\beta_1 \wedge \beta_2$, $\beta_1 \supset \beta_2$, $\beta_1 \equiv \beta_2$, 利用归纳假设(即 β_1, β_2 均满足(甲)), 易证 α 也满足(甲).

如果 α 呈 $\exists y\beta(x; p; A; y)$ 之形, 依归纳假设有

(乙) $\beta(a; p^\circ; A; y) = \beta(a'; p^\circ; A'; y')$

但 $\exists y\beta(a; p^\circ; A; y)$ 真, 即在 I 中有个体 b 使得 $\beta(a; p^\circ; A; b)$ 真, 根据(乙) $\beta(a'; p^\circ; A'; b')$ 真, 即 $\exists y\beta(a'; p^\circ; A'; y)$ 真, 所以(甲)成立.

如果 α 呈 $\forall y\beta(x; p; X; y)$ 之形, 同法可证(甲)成立.

因此归纳法(甲)得证.

由(甲)可知, α 在 I 中永真(可满足)当且仅当 α 在 J 中永真(可满足), 定理得证.

由本定理可见, 永真性和可满足性, 与个体域中的个体性质无关, 仅与个体域中的个体数有关. 因此今后在讨论永真性可满足性时, 我们永远以 $\{1, 2, \cdots, k\}$ 作为具有 k 个个体的个体域的代表, 特名 k 域, α 在 k 域上永真叫做 k 永真, α 在 k 域上可满足叫做 k 可满足.

定理 如果 $h < k$, 则由 α 为 k 永真可推得 α 为 h 永真, 由 α 为 h 可满足可推得 k 可满足. 或叙为由大永真可得小永真, 由小可满足可得大可满足.

[证] 设 $(a_1, \cdots, a_n; p_1^\circ, \cdots, p_m^\circ; A'_1, \cdots, A'_t)$ 是 h 域上 α 的任一指派, 记为 π. 今在 k 域作如下指派:

$$(a_1, \cdots, a_n; p_1^\circ, \cdots, p_m^\circ; A'_1, \cdots, A'_t)$$

其中诸 A'_i 如下定义: 对于 k 上的任一 r 元矢量 $(y_1, \cdots, y_r)$

$$A'_i(y_1, \cdots, y_r) = A_i(y'_1, \cdots, y'_r)$$

其中 $y'_j (j = 1, \cdots, r)$, 当 y_j 属于 h 域时有 $y'_j = y_j$, 否则 $y'_j = 1$.

这个指派称为指派 π 的导出指派, 可以证明:

$$\alpha(a_1, \cdots, a_n; p_1^\circ, \cdots, p_m^\circ; A_1', \cdots, A_t')$$
$$=\alpha(a_1, \cdots, a_n; p_1^\circ, \cdots, p_m^\circ; A_1, \cdots, A_t)$$

由此可知,若 α 为 h 可满足,则在 h 域上有指派(设为 π)使得 α 为真,因而在 k 域上,π 的导出指派也使得 α 为真,所以 α 为 k 可满足.

若 α 为 h 永真,则对于 h 域上的任一指派 π,其 k 域的导出指派必为 α 的成真指派,因而 π 也是 α 的成真指派,所以 α 为 h 永真.

定义 如果公式 α 在域 I 中永真(可满足)当且仅当公式 β 在 I 域中永真(可满足),则说 α 与 β 在域 I 中同永真性(同可满足性). 如果公式 α 永真(可满足)当且仅当公式 β 永真(可满足),则说 α 与 β 同永真性(同可满足性).

定义 把关于公式 α 中的一切自由个体变元的全称(存在)量词置在 α 的首部,所得的公式称为 α 的全称(存在)封闭式.

例如,设 $\alpha = \exists x X(x, y) \wedge \forall y X(x, y)$,

则 α 的全称封闭式是 $\forall x \forall y(\exists x X(x, y) \wedge \forall y X(x, y))$;

α 的存在封闭式是 $\exists x \exists y(\exists x X(x, y) \wedge \forall y X(x, y))$.

由定义可知,全称(存在)封闭式中是没有自由个体变元的.

定理 α 与 α 的全称封闭式在每个域中均同永真性,α 与 α 的存在封闭式在每个域中均同可满足性.

[证] 由同永真性和同可满足性的定义容易得证.

习 题

1. 试求下列公式在所给指派之下的值:

1.1 $\exists x(Xx \equiv Yx) \supset \exists x \exists y(Xx \supset Yy)$

I 为 $\{0, 1, 2, 3, 4\}$, $(Xe, Ye) = (e<2,\ e$ 为偶数$)$;

1.2 $\forall x \exists y(Xxy \wedge P \wedge Yxyz)$

I 为 $\{1, 3, 5\}$, $(Z, P, Xe_1e_2, Ye_1e_2e_3) = (1, T, e_1 \geqslant e_2, e_1 - e_2 = e_3)$.

2. 求 $\forall x \forall y Xxy$ 在 $\{1, 2\}$ 域上的成真指派和成假指派.

3. 试把下列公式化为前束范式:

3.1 $\forall x \alpha(x) \equiv \forall x \beta(x)$;

3.2 $\forall x \forall y \alpha(x, y) \supset (\forall x \exists y \beta(x, y) \wedge \exists x \exists y \alpha(x, y))$;

3.3 $\forall x (\forall y (\alpha(x) \supset \beta(y)) \supset \forall x (\alpha(x) \wedge \exists y \beta(y))$.

4. 下列二组公式是否同真假, 试证明之(其中 β 中不含自由变元 x):

4.1 $\forall x \alpha(x) \supset \beta$ 与 $\exists x (\alpha(x) \supset \beta)$;

4.2 $\forall x \alpha(x) \supset \beta$ 与 $\forall x (\alpha(x) \supset \beta)$.

5. 试证对于 α 而言, h 域上的任一指派与其 k 域上的导出指派必取相同的值.

6. 讨论 $\forall x Xx$, $\exists x Xx$ 的永真性及可满足性的情形, 即讨论在何域上永真, 在何域上可满足.

7. 证明

$$\exists x \forall y \alpha(x, y) \text{ 与 } \exists x ((\exists y (\alpha(x, y)) \wedge \overline{X(x, y)}) \vee \forall y X(x, y))$$

在每个域上都同永真性.

§2.5 狭义谓词演算永真公式的公理系统

狭义谓词演算的永真公式的公理系统如下:

甲. 组成规则.

命题变元: $p, q, r, p_1, q_1, r_1, \cdots$

个体变元: $x, y, z, x_1, y_1, z_1, \cdots$

谓词变元: 一元的 $X^1, Y^1, Z^1, X_1^1, X_2^1, \cdots$

二元的 $X^2, Y^2, Z^2, X_1^2, X_2^2, \cdots$

n 元的 $X^n, Y^n, Z^n, X_1^n, X_2^n, \cdots$

但在下文中, 将略去表示元数的肩码, 而随上下文确定.

联结词: $-$, $\vee$, $\wedge$, $\supset$, $\equiv$

量词: $\forall$(全称量词), $\exists$(存在量词)

括号：(,)是括号

公式：命题变元为公式；如果X为n元谓词，$x_1, \cdots, x_n$为n个个体变元，则$Xx_1x_2\cdots x_n$为公式(谓词填式)；如果α，β为公式，则$\bar{\alpha}$，$(\alpha \vee \beta)$，$(\alpha \wedge \beta)$，$(\alpha \supset \beta)$，$(\alpha \equiv \beta)$为公式；如果α为公式，x为个体变元，则$(\forall x\alpha)$，$(\exists x\alpha)$也为公式；所谓公式仅限于此.

此外还须定义自由出现、约束出现等概念.

自由出现与约束出现：如果$\forall x\beta$或$\exists x\beta$是α中的一个子公式，则x在该子公式中的一切出现均叫做x在α中的约束出现；如果x在α中的出现为非约束出现，则叫做x在α中的自由出现.

注意，变元x在α中可能既约束出现又自由出现，由β而作$\forall x\beta$，$\exists x\beta$时，并不要求x在β中自由出现.

约束关系：设$\forall x\beta(\exists x\beta)$为$\alpha$的子公式，$x$在$\beta$中的一切自由出现均叫做受($\beta$紧前)量词$\forall(\exists)$的约束.

注意，x在β中的约束出现，不受β紧前量词$\forall(\exists)$的约束.

自由变元与约束变元：如果一变元在一公式中有自由出现(约束出现)，则该变元叫该公式的自由变元(约束变元)．如一公式中不同自由变元的个数为n，则该公式称为n元公式.

注意，就一公式而言，同一变元可以既自由又约束.

封闭式：设α为n元公式，如果在α前，用全称量词(存在量词)把n个自由变元约束起来，所得的公式称为α的全称(存在)封闭式．α的全称封闭式记为$\Delta\alpha$．例如，$\forall xXxy \supset \exists uYuv$的全称封闭式是$\Delta(\forall xXxy \supset \exists uYuv)$，即为$\forall y\forall v(\forall xXxy \supset \exists uYuy)$.

乙．推理部分.

公理(其中α, β, γ等表示任意的谓词演算公式)：

10^{Δ}　$\Delta(\alpha \supset \alpha)$

11^{Δ}　$\Delta((\alpha \supset (\beta \supset \gamma)) \supset (\beta \supset (\alpha \supset \gamma)))$

12^{Δ}　$\Delta((\alpha \supset \beta) \supset ((\beta \supset \gamma) \supset (\alpha \supset \gamma)))$

13^{Δ} $\Delta((\alpha\supset(\alpha\supset\beta))\supset(\alpha\supset\beta))$

14^{Δ} $\Delta((\alpha\equiv\beta)\supset(\alpha\supset\beta))$

15^{Δ} $\Delta((\alpha\equiv\beta)\supset(\beta\supset\alpha))$

16^{Δ} $\Delta(\alpha\supset\beta)\supset((\beta\supset\alpha)\supset(\alpha\equiv\beta))$

17^{Δ} $\Delta((\alpha\wedge\beta)\supset\alpha)$

18^{Δ} $\Delta((\alpha\wedge\beta)\supset\beta)$

19^{Δ} $\Delta(\alpha\supset(\beta\supset(\alpha\wedge\beta)))$

20^{Δ} $\Delta(\alpha\supset(\alpha\vee\beta))$

21^{Δ} $\Delta(\beta\supset(\alpha\vee\beta))$

22^{Δ} $\Delta((\alpha\supset\gamma)\supset((\beta\supset\gamma)\supset((\alpha\vee\beta)\supset\gamma)))$

23^{Δ} $\Delta((\alpha\supset\bar{\beta})\supset(\beta\supset\bar{\alpha}))$

24^{Δ} $\Delta(\bar{\bar{\alpha}}\supset\alpha)$

50 $\Delta(\forall x\alpha(x)\supset\alpha(\xi))$

$\alpha(\xi)$指在$\alpha(x)$中把x合法地代入以任意个体变元ξ的结果.

51 $\Delta(\alpha(\xi)\supset\exists x\alpha(x))$ $\alpha(\xi)$同上

推理规则:

Δ分离规则(省记为分$^{\Delta}$): $\Delta(\alpha\supset\beta),\Delta\alpha\vdash\Delta\beta$

全称规则(省记为全): $\Delta(\gamma\supset\alpha(x))\vdash\Delta(\gamma\supset\forall x\alpha(x))$

其中γ中不含自由x.

存在规则(省记为存): $\Delta(\alpha(x)\supset\gamma)\vdash\Delta(\exists x\alpha(x)\supset\gamma)$

其中γ中不含自由x

关于这个公理系统,我们给出几点注意:

第一, $10^{\Delta}\sim24^{\Delta}$ 并非 10～24 的特例,而是 10～24 的推广. 例如, $\forall xXx\supset\forall xXx$ 是公理 10 之一,也是公理 10^{Δ} 之一, 但 $\forall x(Xx\supset Xx)$却不是公理 10,而是公理 10^{Δ} 之一,一般讲,仅当α,β,γ不含自由变元时 $10^{\Delta}\sim24^{\Delta}$ 才是 10～24 的特例,否则不是.

第二,"分$^{\Delta}$"也是"分"的推广,而非特例, 例如, 由 $\forall x\alpha(x)\supset$

$\forall y\beta(y)$, $\forall x\alpha(x)$得 $\forall y\beta(y)$，这既是实施分离规则而得，也是实施分$^{\Delta}$规则而得，但是由 $\forall y\forall x(\alpha(x)\supset\beta(y))$，$\forall x\alpha(x)$推得 $\forall y\beta(y)$只能实施分$^{\Delta}$规则而得，不能由实施分离规则而得。显然，如果实施的公式中没有自由变元，则实施"分"和实施"分$^{\Delta}$"完全一样，如果公式中有自由变元，则"分"和"分$^{\Delta}$"就不一样了．这时只能用"分$^{\Delta}$"．

易见，由"分$^{\Delta}$"及 10^{Δ}～24^{Δ} 命题演算中的永真公式即 100～159 均可推出，而且用完全相同的方法还可推出 100^{Δ}～159^{Δ}．

明白了这两点后，即使把"分$^{\Delta}$"写成"分"，把 10^{Δ}～24^{Δ} 写成10～24，把 100^{Δ}～159^{Δ} 写成 100～159 也不致误会了，并且今后经常把全称封闭式的"Δ"略去不写．

第三，如果把 "$\forall x\alpha(x)$" 想象为 "$\alpha(x_1)\wedge\alpha(x_2)\wedge\cdots\wedge\alpha(x_n)\wedge\cdots$"，把"$\exists x\alpha(x)$"想象为"$\alpha(x_1)\vee\alpha(x_2)\vee\cdots\vee\alpha(x_n)\vee\cdots$"，则

50 实际上是 17，18 的推广，即是$(\alpha\wedge\beta)\supset\alpha$，$(\alpha\wedge\beta)\supset\beta$ 的推广．

51 实际上是 20，21 的推广，即是 $\alpha\supset(\alpha\vee\beta)$，$\beta\supset(\alpha\vee\beta)$的推广．

全称规则实际上是 $\alpha\supset\beta$，$\alpha\supset\gamma\vdash\alpha\supset(\beta\wedge\gamma)$的推广．

存在规则实际上是 $\alpha\supset\gamma$，$\beta\supset\gamma\vdash(\alpha\vee\beta)\supset\gamma$ 的推广．

利用这个观点不但对 50，51，"全"，"存"有进一步理解，且对以后的推演有一定的指导作用．

第四，由这三个规则，可得出下列一些导出规则：

全0：　$\Delta\alpha(x)\vdash\Delta\forall x\alpha(x)$

全n：　$\Delta(\gamma_1\supset(\gamma_2\supset(\cdots\supset(\gamma_n\supset\alpha(x)))))$

$\vdash\Delta(\gamma_1\supset(\gamma_2\supset(\cdots\supset(\gamma_n\supset\forall x\alpha(x)))))$

其中诸 γ 中无自由 x 出现

存n：　$\Delta(\gamma_1\supset(\gamma_2\supset(\cdots\supset(\gamma_n\supset(\alpha(x)\supset\beta)))))$

$$\vdash\Delta(\gamma_1\supset(\gamma_2\supset(\cdots\supset(\gamma_n\supset(\exists x\alpha(x)\supset\beta)))))$$

其中 β 和诸 γ 中无自由 x 出现

今后可把"全 n"缩写为

$$\Delta((\gamma\supset)^n\alpha(x))\vdash\Delta((\gamma\supset)^n\forall x\alpha(x))$$

"存 n"缩写为

$$\Delta((\gamma\supset)^n(\alpha(x)\supset\beta))\vdash\Delta((\gamma\supset)^n(\exists x\alpha(x)\supset\beta))$$

(请自行导出这三个规则)

§2.6 推理定理

和命题演算一样,谓词演算的推理过程也有两种:永真推理过程和假设推理过程. 例如 $\forall x\alpha(x)\supset\forall y\alpha(y)$(这里 $\alpha(y)$ 是在 $\alpha(x)$ 中把 x 合法代以 y 的结果)的永真推理过程是:

50=(1) $\forall x\alpha(x)\supset\alpha(y)$

全(1)=(2) $\forall x\alpha(x)\supset\forall y\alpha(y)$

故原式得证.

假设推理过程是:

(1)* $\forall x\alpha(x)$

分50(1)=(2) $\alpha(y)$

全0(2)=(3) $\forall y\alpha(y)$

故原式得证.

由此例可见,在假设推理过程中除使用"分"外,还使用"全0",一般说,可使用"全$_n$","存$_n$"等规则. 显然要问:在谓词演算中,假设推理过程和永真推理过程的关系是否与在命题演算中两者的关系一样呢?这就是本节所要解决的问题.

推理定理 如果 $\gamma_1,\gamma_2,\cdots,\gamma_k\overset{(全,存)}{\vdash}\beta$,且在推理过程中对诸 γ 永不作代入,诸 γ 至少被使用一次,实施全、存规则时绝不对诸 γ 中的自由变元进行,则有

$$\gamma_1, \cdots, \gamma_k \overset{\text{(全,存)}}{\vdash\!\!\!-\!\!\!-} \gamma_{h+1} \supset (\cdots \supset (\gamma_k \supset \beta))$$
$$(0 \leqslant h < k)$$

[证] 设 $\alpha_1, \alpha_2, \cdots, \alpha_n (=\beta)$ 为由 $\gamma_1, \cdots, \gamma_k$ 证明 β 的假设推理过程. 今依下法另作一公式序列 $\beta_1, \beta_2, \cdots, \beta_n$.

如果 α_i 为假设 $\gamma_1, \cdots, \gamma_k$ 之一, 则 $\beta_i = \alpha_i$, 即仍为假设(把 β_i 看成由 α_i 加头而得, 所加之头的个数为 0).

如果 α_i 为永真公式, 则 $\beta_i = \alpha_i$ 仍为永真公式(把 β_i 看成由 α_i 加头而得, 所加之头的个数为 0).

如果 α_i 为假设 $\gamma_{h+1}, \cdots, \gamma_k$ 之一, 则 $\beta_i = \alpha_i \supset \alpha_i$ 即为公理 10 (把 β_i 看成由 α_i 加头而得, 所加之头是 $\gamma_{h+1}\supset, \cdots, \gamma_k\supset$ 之一).

如果 α_i 为由 $\alpha_s, \alpha_t (s,\ t$ 均$< i)$ 实施"分"而得, 则 β_i 为由 α_i 加头而得, 所加之头为 β_s 对 α_s 所加之头以及 β_t 对 α_t 所加之头的总和, 所用的规则显然为广义分离规则, 即"分分$\nabla^2$10".

如果 α_i 为由 $\alpha_s (s<i)$ 实施"全$_n$"或"存$_n$" 而得, 则 β_i 为由 α_i 加头而得, 所加之头为 β_s 对 α_s 所加之头, 所用的规则显然为"全 $m+n$"或"存 $m+n$"(m 为 β_s 对 α_s 所加之头的个数).

这样最后所得的 β_n 必将由 α_n(即 β)加头而得, 所加之头显然为若干 $\gamma_{k+1}\supset$, …, $\gamma_k\supset$的连接, 它们中的每一个可以出现多次, 但至少出现一次, 这是因为诸 γ 至少被使用一次. 利用"凝"和"调", 定理得证.

在日常推理过程中, 还经常使用存在推理法. 所谓存在推理法是指: 当假设中有 $\exists x \alpha x$ 或推理过程中推得 $\exists x \alpha x$ 时, 便可引入额外假设 $\alpha(e)$, e 为尚未使用过的变元, 意指"暂设 e 为使得 $\exists x \alpha x$ 成立的 x", e 叫做额外变元, 它依赖于额外假设中的自由变元. 推理在此新条件下继续进行下去. 且一旦推出一个不含自由 e 的公式 β 后, 便可立即消去额外假设 $\alpha(e)$, 而认为可以从原有的假设推出 β. 这种推理过程的合理性可以由下面的定理来保证.

存在推理定理 如果有

(1) $\gamma_1, \cdots, \gamma_k, \exists x\alpha x, \alpha(e) \vdash \beta$

其中β中不含自由 e，且在推理过程中决不对假设中的自由变元和额外变元 e 实施"全""存"，$\alpha(e)$至少被使用一次，则有

(2) $\gamma_1, \cdots, \gamma_k, \exists x\alpha x \vdash \beta$

[证] 根据推理定理由(1)得

$$\gamma_1, \cdots, \gamma_k, {}^{(3)}\exists x\alpha x \vdash {}^{(4)}\alpha(e) \supset \beta$$

$$\vdash {}^{(5)}\exists x d x \supset \beta \qquad (存(4))$$

$$\vdash \beta \qquad (分(5)(3))$$

定理得证.

利用推理定理和存在推理定理可使推理过程非常简单，使得除去推理过程的首尾部分外，其余部分基本上没有量词出现，从而推理过程基本上是命题演算的推理．其方法如下：

(1) 列出待证公式的各假设，假设中的全称量词可用 50 除去，存在量词可用额外假设法除去．

(2) 照命题演算的推理过程推导．

(3) 待证公式的后件如有全称量词，可设法使用"全 0"规则引入，如有存在量词可使用 51 引入．

在下列各例中我们仍以"*"及"**"分别表示假设及额外假设．

例 1: 试证 $\forall x(\alpha(x) \supset \beta(x)) \supset (\exists x\alpha(x) \supset \exists x\beta(x))$

[证] *(1) $\forall x(\alpha(x) \supset \beta(x))$

*(2) $\exists x\alpha(x)$

**(3) $\alpha(e)$

分 50(1)=(4) $\alpha(e) \supset \beta(e)$

分(4)(3)=(5) $\beta(e)$

分(51)(5)=(6) $\exists x\beta(x)$

根据存在推理定理，由(6)消去(3)得(7) $\exists x\beta(x)$ (注意以后

写为(6)消(3)=(7))

故由推理定理得证.

例 2: 试证　$\exists x\forall y\alpha(x,y)\supset\forall y\exists x\alpha(x,y)$

[证]　*(1)　$\exists x\forall y\alpha(x,y)$

**(2)　$\forall y\alpha(e,y)$

分 50(2)=(3)　　$\alpha(e,y)$

分 51(2)=(4)　$\exists x\alpha(x,y)$

(4)消(2)=(5)　$\exists x\alpha(x,y)$

全 0(5)=(6)　$\forall y\exists x\alpha(x,y)$

由推理定理得证.

本公式的逆命题是不成立的，事实上在本公理系统中是证明不出来的，现在试看证明逆命题时我们会遇到什么困难. 本公式的逆命题为: $\forall y\exists x\alpha(x,y)\supset\exists x\forall y\alpha(x,y)$

*(1)　$\forall y\exists x\alpha(x,y)$

分 50(1)=(2)　$\exists x\alpha(x,y)$

**(3)　$\alpha(e,y)$

因为(3)是假设，e 和 y 都是(3)中的自由变元，所以不能对(3)实施全称规则，于是推导无法进行下去. 要想把(3)中的 e 除去，只有利用 51，这样又退回到(2). 因此无论如何推不出待证命题的后件 $\exists x\forall y\alpha(x,y)$.

下面列出一些比较重要的定理.

关于量词间的颠倒次序以及量词的改名:

500　$\forall x\alpha(x)\supset\exists x\alpha(x)$

501　$\forall x\alpha(x)\equiv\forall y\alpha(y)$　这里$\alpha(x)$中无自由y, $\alpha(y)$是在$\alpha(x)$中把 x 合法代以 y 的结果.

502　$\exists x\alpha(x)\equiv\exists y\alpha(y)$　$\alpha(x)$, $\alpha(y)$间的关系同上.

503　$\forall x\forall y\alpha(x,y)\equiv\forall y\forall x\alpha(x,y)$

504 $\exists x \exists y \alpha(x, y) \equiv \exists y \exists x \alpha(x, y)$

505 $\exists x \forall y \alpha(x, y) \supset \forall y \exists x \alpha(x, y)$

关于量词和联结词的颠倒次序:

510 $\forall x \overline{\alpha}(x) \equiv \overline{\exists x \alpha(x)}$

511° $\exists x \overline{\alpha}(x) \equiv \overline{\forall x \alpha(x)}$

512 $\forall x(\gamma \supset \alpha(x)) \equiv (\gamma \supset \forall x \alpha(x))$ 这里和以下 γ 中无自由 x

513 $\forall x(\alpha(x) \supset \gamma) \equiv (\exists x \alpha(x) \supset \gamma)$

514° $\forall x(\alpha(x) \vee \gamma) \equiv (\forall x \alpha(x) \vee \gamma)$

515 $\forall x(\alpha(x) \wedge \gamma) \equiv (\forall x \alpha(x) \wedge \gamma)$

516 $\exists x(\alpha(x) \vee \gamma) \equiv (\exists x \alpha(x) \vee \gamma)$

517 $\exists x(\alpha(x) \wedge \gamma) \equiv (\exists x \alpha(x) \wedge \gamma)$

518° $\exists x(\gamma \supset \alpha(x)) \equiv (\gamma \supset \exists x \alpha(x))$

519° $\exists x(\alpha(x) \supset \gamma) \equiv (\forall x \alpha(x) \supset \gamma)$

520 $\forall x(\alpha(x) \wedge \beta(x)) \equiv (\forall x \alpha(x) \wedge \forall x \beta(x))$

521 $(\forall x \alpha(x) \vee \forall x \beta(x)) \supset (\forall x(\alpha(x) \vee \beta(x)))$

522 $\forall x(\alpha(x) \supset \beta(x)) \supset (\forall x \alpha(x) \supset \forall x \beta(x))$

523 $\forall x(\alpha(x) \supset \beta(x)) \supset (\exists x \alpha(x) \supset \exists x \beta(x))$

524 $\forall x(\alpha(x) \equiv \beta(x)) \supset (\forall x \alpha(x) \equiv \forall x \beta(x))$

525 $\forall x(\alpha(x) \equiv \beta(x)) \supset (\exists x \alpha(x) \equiv \exists x \beta(x))$

526 $\exists x(\alpha(x) \wedge \beta(x)) \supset (\exists x \alpha(x) \wedge \exists x \beta(x))$

527 $\exists x(\alpha(x) \vee \beta(x)) \equiv (\exists x \alpha(x) \vee \exists x \beta(x))$

528° $\exists x(\alpha(x) \supset \beta(x)) \equiv (\forall x \alpha(x) \supset \exists x \beta(x))$

和在命题演算中一样，在谓词演算中也有下列替换定理和替换规则.

替换定理 设 $\varphi(\alpha)$ 是任一含有 α 的公式，$\varphi(\beta)$ 是在 $\varphi(\alpha)$ 中把若干个 α 替换以 β 的结果，又设在 α 中自由但在 $\varphi(\alpha)$ 中约束的变元以及在 β 中自由在 $\varphi(\beta)$ 中约束的变元，均在 $x_1, \cdots, x_n$ 之中，

则有

$$\Delta(\forall x_1\forall x_2\cdots\forall x_n(\alpha\equiv\beta)\supset(\varphi(\alpha)\equiv\varphi(\beta)))$$

替换规则：设 $\varphi(\alpha)$, $\varphi(\beta)$ 关系同上，则有下列规则

$$\Delta(\alpha\equiv\beta)\vdash\Delta(\varphi(\alpha)\equiv\varphi(\beta))$$

$$\Delta(\alpha\equiv\beta),\Delta(\varphi(\alpha))\vdash\Delta(\varphi(\beta))$$

证明仍用数学归纳法.

习　题

指出下列"证明过程"的错误所在

1. $\exists x\alpha(x)\supset\forall x\alpha(x)$ 的"证明"如下：

*(1) $\exists x\alpha(x)$

**(2) $\alpha(e)$

全(2)=(3) $\forall x\alpha(x)$

(3)消(2)=(4) $\forall x\alpha(x)$

由推理定理得证.

2. $(\exists x\alpha(x)\wedge\exists x\beta(x))\supset\exists x(\alpha(x)\wedge\beta(x))$ 的"证明"如下：

*(1) $\exists x\alpha(x)\wedge\exists x\beta(x)$

分 17(1)=(2) $\exists x\alpha(x)$

分 18(1)=(3) $\exists x\beta(x)$

**(4) $\alpha(e)$

**(5) $\beta(e)$

分分 19(4)(5)=(6) $\alpha e\wedge\beta e$

分 51(6)=(7) $\exists x(\alpha(x)\wedge\beta(x))$

(7) 消(4)和(5)=(8) $\exists x(\alpha(x)\wedge\beta(x))$

由推理定理得证.

3. $\forall x(\alpha(x)\supset\beta(x))\supset(\exists x\alpha(x)\supset\forall x\beta(x))$ 的"证明"如下：

*(1) $\forall x(\alpha(x)\supset\beta(x))$

*(2) $\exists x\alpha(x)$

**(3) $\alpha(x)$

分 50(1)=(4) $\alpha(x)\supset\beta(x)$

分(4)(3)=(5) $\beta(x)$

全 0(5)=(6) $\forall x\beta(x)$

(6)消(3)=(7) $\forall x\beta(\bar{x})$

由推理定理得证.

§2.7 关于谓词演算公理系统的讨论

2.7.1 不矛盾性

定理 本系统推出的定理都是永真公式.

[**证**] 因为本系统中 17 条公理都是永真公式,又因为本系统中的分离规则、全称规则和存在规则都保持永真性,故本系统推出的一切定理都是永真公式.

由此易证下面两条定理.

定理 对于任何公式 α, α 和 $\bar{\alpha}$ 中至少有一个不是本系统的定理.

定理 不是所有的公式都是本系统的定理.

2.7.2 完备性

首先给出一种对由自然数组成的 h 元矢量 $(a_1, a_2\cdots, a_h)$ (h 为任一确定的数)进行排序的方法.

(i) 先按诸分量之和的大小来排序,其和小者排在前,其和大者排在后.

(ii) 分量之和相等时按"字典次序"排序,即从左到右按各个分量的大小来排序,先按 a_1 排序,小者在前大者在后,相等者再按 a_2 排序小者在前大者在后,相等者再按 a_3 排序,如此一直排下去。

(iii) 用从 0 开始的自然数对如此排序的 h 元矢量进行编号.

例如按此排序方法,自然数组成的 3 元矢量的次序如下:

第 0 个 3 元矢量是(0, 0, 0)

以上分量之和为 0

第 1 个 3 元矢量是(0, 0, 1)

第 2 个 3 元矢量是(0, 1, 0)

第 3 个 3 元矢量是(1, 0, 0)

以上分量之和为 1

第 4 个 3 元矢量是(0, 0, 2)

第 5 个 3 元矢量是(0, 1, 1)

第 6 个 3 元矢量是(0, 2, 0)

依次下去是(1, 0, 1), (1, 1, 0), (2, 0, 0)

以上分量之和为 2

再下去是(0, 0, 3), (0, 1, 2), (0, 2, 1), (0, 3, 0), …

显然按此方法排序后，每一个 h 元矢量都对应有唯一一个自然数(即其编号)，每一个自然数也都对应唯一一个 h 元矢量.

我们把 h 元矢量 $(a_1, a_2, \cdots, a_h)$ 的编号记为 $\rho(a_1, \cdots, a_h)$.

把第 n 个 h 元矢量记为 $(\langle n_1\rangle, \langle n_2\rangle, \cdots, \langle n_h\rangle)$，即编号为 n 的 h 元矢量的第 i 个分量记为 $\langle n_i\rangle (i=1, 2, \cdots, h)$.

显然有如下的简单性质:

$$\langle \rho(a_1, a_2, \cdots, a_h)_i\rangle = a_i \qquad (i=1, 2, \cdots, h)$$

$$\rho(\langle n_1\rangle, \langle n_2\rangle, \cdots, \langle n_h\rangle) = n$$

$$\rho(a_1, a_2, \cdots, a_h) \geqslant \sum_{i=1}^{h} a_i \geqslant a_i \qquad (i=1, 2, \cdots, h)$$

仍旧就 3 元矢量来说明这些记号的使用.

$$\rho(0, 0, 0) = 0,\ (\langle 0_1\rangle, \langle 0_2\rangle, \langle 0_3\rangle) = (0, 0, 0)$$

$$\rho(0, 0, 1) = 1,\ (\langle 1_1\rangle, \langle 1_2\rangle, \langle 1_3\rangle) = (0, 0, 1)$$

$$\rho(0, 1, 0) = 2,\ (\langle 2_1\rangle, \langle 2_2\rangle, \langle 2_3\rangle) = (0, 1, 0)$$

…

$$\rho(0,1,1)=5,\ (\langle 5_1\rangle,\langle 5_2\rangle,\langle 5_3\rangle)=(0,1,1)$$

$$\cdots$$

由此不难计算带有部分$\langle n_i\rangle$的矢量的序号，例如：

$$\rho(\langle 0_1\rangle,0,0)=\rho(0,0,0)=0$$

$$\rho(\langle 1_1\rangle,0,0)=\rho(0,0,0)=0$$

$$\rho(\langle 2_1\rangle,0,0)=\rho(0,0,0)=0$$

$$\rho(\langle 4_1\rangle,0,0)=\rho(0,0,0)=0$$

$$\rho(\langle 5_1\rangle,0,0)=\rho(0,0,0)=0$$

$$\rho(\langle 3_1\rangle,0,0)=\rho(\langle 7_1\rangle,0,0)=\rho(1,0,0)=3$$

$$\rho(\langle 2_0\rangle,\langle 2_1\rangle,0)=\rho(\langle 5_1\rangle,\langle 5_2\rangle,0)=\rho(0,1,0)=2$$

让我们再引进一些新的记号.

设α是任一谓词演算公式，

$\tilde{\alpha}$是α的前束范式.

又设$\tilde{\alpha}$中有h个存在量词，它们依次是$\exists x_1,\ \exists x_2,\cdots,\ \exists x_h$；有$k$个全称量词，它们依次是$\forall y_1,\ \forall y_2,\cdots,\ \forall y_k$.

又设第t个全称量词$\forall y_t$之前有s_t个存在量词$\exists x_1,\exists x_2,\cdots,\exists x_{s_t}(t=1,2,\cdots,k)$. 注意$s_t$可为0且不超过$h$，即$0\leqslant s_t\leqslant h$. 显然还有$0\leqslant s_1\leqslant s_2\leqslant\cdots\leqslant s_k\leqslant h$. 这样$\tilde{\alpha}$可表为

$$\tilde{\alpha}=\exists x_1\cdots\exists x_{s_1}\forall y_1\exists x_{s_1+1}\cdots\exists x_{s_2}\forall y_2\cdots\forall y_k\exists x_{s_k+1}\cdots\exists x_h$$
$$\beta(x_1,\cdots,x_{s_1},y_1,x_{s_1+1},\cdots,x_{s_2},y_2,\cdots,y_k,x_{s_k+1},\cdots,x_h)$$

其中β中不含量词，$0\leqslant s_1\leqslant s_2\leqslant\cdots\leqslant s_k\leqslant h$.

把β中的y_t代以$k\cdot\rho(x_1,\cdots,x_{s_t},0,\cdots,0)+t(t=1,2,\cdots,k)$，结果式记为$\beta\Delta(x_1,\cdots,x_h)$，即

$$\beta\Delta(x_1,\cdots,x_h)=\beta(x_1,\cdots,x_{s_1},k\cdot\rho(x_1,\cdots,x_{s_1},0,\cdots,0)+1,$$
$$x_{s_1+1},\cdots,x_{s_2},k\cdot\rho(x_1,\cdots,x_{s_2},0,\cdots,0)+2,\cdots,$$
$$k\cdot\rho(x_1,\cdots,x_{s_k},\ 0,\cdots,0)+k,x_{s_k+1},\cdots,x_h)$$

把$\beta\Delta$中的$(x_1,\cdots,x_h)$代以第n个h元矢量$(\langle n_1\rangle,\ \cdots,\ \langle n_h\rangle)$，

结果式记为 β_n, 即

$$\beta_0=\beta\Delta(\langle 0_1\rangle,\cdots,\langle 0_k\rangle)$$
$$\beta_1=\beta\Delta(\langle 1_1\rangle,\cdots,\langle 1_k\rangle)$$
$$\beta_2=\beta\Delta(\langle 2_1\rangle,\cdots,\langle 2_k\rangle)$$
$$\cdots$$
$$\beta_n=\beta\Delta(\langle n_1\rangle,\cdots,\langle n_k\rangle)$$
$$\cdots$$

又令 $\gamma_n=\beta_0\vee\beta_1\vee\cdots\vee\beta_n$

因此
$$\gamma_0=\beta_0$$
$$\gamma_1=\beta_0\vee\beta_1=\gamma_0\vee\beta_1$$
$$\gamma_2=\beta_0\vee\beta_1\vee\beta_2=\gamma_1\vee\beta_1$$
$$\cdots$$
$$\gamma_n=\gamma_{n-1}\vee\beta_n$$
$$\cdots$$

我们用一个简单的例子来说明这些记号. 设

$$\tilde{\alpha}=\forall y_1\exists x_1\exists x_2\forall y_2\exists x_3\forall y_3\forall y_4((X_1(x_1)\wedge p)\supset(X_1(x_2)\wedge X_2(y_1,x_1,x_2,y_2,x_3,y_3,y_4)))$$

这里 $h=3, k=4, s_1=0, s_2=2, s_3=s_4=3.$

$$\beta(y_1,x_1,x_2,y_2,x_3,y_3,y_4)=(X_1(x_1)\wedge p)\supset(X_1(x_2)\wedge X_2(y_1,x_1,x_2,y_2,x_3,y_3,y_4))$$

$$\begin{aligned}\beta\Delta(x_1,x_2,x_3)&=\beta(1,x_1,x_2,4\cdot\rho(x_1,x_2,0)+2,x_3,4\cdot\rho(x_1,x_2,x_3)+3,4\cdot\rho(x_1,x_2,x_3)+4)\\&=(X_1(x_1)\wedge p)\supset(X_1(x_2)\wedge X_2(1,x_1,x_2,4\cdot\rho(x_1,x_2,0)+2,x_3,4\cdot\rho(x_1,x_2,x_3)+3,4\cdot\rho(x_1,x_2,x_3)+4))\end{aligned}$$

$$\beta_0=\beta\Delta(0,0,0)=(X_1(0)\wedge p)\supset(X_1(0)\wedge X_2(1,0,0,2,0,3,4))$$

$$\beta_1=\beta\Delta(0,0,1)=(X_1(0)\wedge p)\supset(X_1(0)\wedge X_2(1,0,0,2,1,7,8))$$

$$\beta_2=\beta\Delta(0,1,0)$$
$$=(X_1(0)\wedge p)\supset(X_1(1)\wedge X_2(1,0,1,10,0,11,12))$$

$$\beta_3=\beta\Delta(1,0,0)$$
$$=(X_1(1)\wedge p)\supset(X_1(0)\wedge X_2(1,1,0,14,0,15,16))$$

$$\beta_4=\beta\Delta(0,0,2)$$
$$=(X_1(0)\wedge p)\supset(X_1(0)\wedge X_2(1,0,0,2,2,19,20))$$

…

让我们继续引进一些新的记号.

首先对诸 β_n 中出现的不同的基本公式(指的是命题变元和谓词填式,注意谓词填式中所填的都是自然数)按出现的先后次序如下地进行排序:

先排 β_0 中的基本公式,再排 β_1 中的且未在前面出现过的基本公式(即 β_0 中出现过的不要再排),再排 β_2 中的且未在前面出现过的基本公式,这样一直排下去,而各个 β_n 中的基本公式按书写时的次序从左到右进行排列.今把这些基本公式依次记为 e_0, $e_1, e_2, \cdots$.

其次把诸 β_n 中的基本公式换成为相应的 e,所得的式子记为 $\tilde{\beta}_n$.

令 $$\tilde{\gamma}_n=\tilde{\beta}_0\vee\tilde{\beta}_1\vee\cdots\vee\tilde{\beta}_n.$$

注意到所有的基本公式都可以看成是以真假为变域的变元,因此我们可以把诸 e 看成是命题变元,因而诸 $\tilde{\beta}_n$ 和诸 $\tilde{\gamma}_n$ 是命题演算公式,并且 β_n 与 $\tilde{\beta}_n$ 既同永真性又同可满足性,γ_n 与 $\tilde{\gamma}_n$ 也既同永真性又同可满足性.

我们仍用上例来说明这些记号.

因为 $\beta_0=(X_1(0)\wedge p)\supset(X_1(0)\wedge X_2(1,0,0,2,0,3,4,))$

所以 $\tilde{\beta}_0=(e_0\wedge e_1)\supset(e_0\wedge e_2)$

因为　$\beta_1=(X_1(0)\wedge p)\supset(X_1(0)\wedge X_2(1,0,0,2,1,7,8,))$

所以　$\tilde{\beta}_1=(\varepsilon_0\wedge\varepsilon_1)\supset(\varepsilon_0\wedge\varepsilon_3)$

因为　$\beta_2=(X_1(0)\wedge p)\supset(X_1(1)$

$\wedge X_2(1,0,1,10,0,11,12))$

所以　$\tilde{\beta}_2=(\varepsilon_0\wedge\varepsilon_1)\supset(\varepsilon_4\wedge\varepsilon_5)$

因为　$\beta_3=(X_1(1)\wedge p)\supset(X_1(0)\wedge X_2(1,1,0,14,0,15,16))$

所以　$\tilde{\beta}_3=(\varepsilon_4\wedge\varepsilon_1)\supset(\varepsilon_0\wedge\varepsilon_6)$

因为　$\beta_4=(X_1(0)\wedge p)\supset(X_1(0)\wedge X_2(1,0,0,2,2,19,20))$

所以　$\tilde{\beta}_4=(\varepsilon_0\wedge\varepsilon_1)\supset(\varepsilon_0\wedge\varepsilon_7)$

…

现在我们利用上面这些记号和性质来证明两条引理.

引理 1　如果 α 永真，则必有一自然数 n 使得 $\tilde{\gamma}_n$ 是一个命题演算的永真公式.

[证]　我们用反证法来证明.

设任何 $\tilde{\gamma}_n$ 均不是命题演算的永真公式。因此每个 $\tilde{\gamma}_n$ 均有成假指派.

先列出 $\tilde{\gamma}_0$ 的成假指派，再列出 $\tilde{\gamma}_1$ 的成假指派，再列出 $\tilde{\gamma}_2$ 的成假指派，….

虽然对于每一个 $\tilde{\gamma}_n$ 言成假指派只有有限多个，但是对全体 $\tilde{\gamma}$ 言，成假指派必有无穷多个. 把全体成假指派组成的集合记为 Π.

因为 $\tilde{\gamma}_n=\tilde{\beta}_0\vee\cdots\vee\tilde{\beta}_{n-1}\vee\tilde{\beta}_n=\tilde{\gamma}_{n-1}\vee\tilde{\beta}_n$

所以 $\tilde{\gamma}_n$ 的成假指派必是 $\tilde{\gamma}_{n-1}$ 的成假指派，从而是其前面任何一个 $\tilde{\gamma}_i(i=0,\cdots,n-1)$ 的成假指派. 而且 Π 中任何一个指派均不会使某个 $\tilde{\gamma}_i$ 成真.

因为如果某个 e_i 在某个 $\tilde{\gamma}_j$ 中出现，则该 e_i 也必在 $\tilde{\gamma}_{j+1}$, $\tilde{\gamma}_{j+2}$, $\tilde{\gamma}_{j+3}$, …中出现，所以对每个 e，必在 Π 中的无穷多个指派中作了指派，因此每个 e 都被指派了无穷多次，或者指派以无穷多次真，或

者指派以无穷多次假.

检查 e_0. 当真被指派了无穷多次时，就把 Π 中 e_0 为真的指派全部保留，其它的指派全部删除. 如真被指派了有限多次，则把 Π 中 e_0 为假的成假指派全部保留，Π 中其它的指派全部删除. 这样保留下的成假指派仍旧有无穷多个，把这些保留下来的无穷多个成假指派组成的集合记为 Π_1，这个集合中 e_0 指派了一个确定的值，记为 e°_0.

检查 e_1，在 Π_1 中 e_1 被指派了无穷多次，或者指派了无穷多次真，或者指派了无穷多次假. 仍旧依照上面的方法从 Π_1 中保留下无穷多个成假指派，这些指派中 e_1 指派了一个确定的值，记为 e°_1.

同法对各个 e_i 进行检查，并选取一个确定的值，记为 e°_i，每次保留下来的无穷多个指派都是成假指派.

这样就全体 e 而言我们得到了如下的指派：

$$(e_0, e_1, e_2, \cdots) \text{指派以} (e^\circ_0, e^\circ_1, e^\circ_2, \cdots)$$

这个指派暂称为“甲指派”.

显然甲指派是任何 $\tilde{\gamma}_n$ 的成假指派，也是任何 $\tilde{\beta}_n$ 的成假指派.

设 $\tilde{\alpha}$ 中所有不同的命题变元是 $p_1, \cdots, p_u$，所有不同的谓词变元是 $X_1, \cdots, X_v$.

今在自然数域中对 $\tilde{\alpha}$ 作如下的指派：

$$(p_1, \cdots, p_u; X_1, \cdots, X_v) \text{指派以} (e^\circ_{i_1}, \cdots, e^\circ_{i_u}; A_1, \cdots, A_v)$$

其中：e_{i_i} 是与 p_i 相对应的那个 $e(i=1, \cdots, u)$，

$e^\circ_{i_i}$ 是 e_{i_i} 在甲指派中所指派的值$(i=1, \cdots, u)$，

而 A_i 是如下定义的常谓词：

设 X_i 是 r 元谓词变元，对于自然数域上任意一个 r 元矢量 $(a_1, \cdots, a_r)$

$$A_i(a_1, \cdots, a_r) = \begin{cases} e^0_j, & \text{当 } X_i(a_1, \cdots, a_r) \text{是某个 } e_j \text{ 时} \\ T, & \text{当 } X_i(a_1, \cdots, a_r) \text{不是任何 } e \text{ 时} \end{cases}$$

这个指派暂称为“乙指派”.

显然，在甲指派之下，$\tilde{\beta}_n$ 所取之值与在乙指派下 β_n 所取之值完全一样，因此乙指派是一切 β_n 的成假指派，即在乙指派下，对于一切 n 均有

$$\beta_n=\beta\Delta(\langle n_1\rangle,\cdots,\langle n_k\rangle)=F.$$

据此我们将证明，在乙指派下，$\tilde{\alpha}=F$.

不失一般性，我们用前面的例子来证明这个结论，即就

$$\tilde{\alpha}=\forall y_1\exists x_1\exists x_2\forall y_2\exists x_3\forall y_3\forall y_4\beta(y_1,x_1,x_2,y_2,x_3,y_3,y_4)$$

来证明如下的结论:

在乙指派下，$\tilde{\alpha}=F$.

因为在乙指派下对于任何的 n 均有

$$\beta_n=\beta(1,\langle n_1\rangle,\langle n_2\rangle,4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,0)+2,\langle n_3\rangle,$$
$$4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,\langle n_3\rangle)+3,$$
$$4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,\langle n_3\rangle)+4)$$

所以在乙指派下，对于任何$(\langle n_1\rangle,\langle n_2\rangle,\langle n_3\rangle)$均有

$$y_3=4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,\langle n_3\rangle)+3$$
$$y_4=4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,\langle n_3\rangle)+4$$

使得

$$\beta(1,\langle n_1\rangle,\langle n_2\rangle,4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,0)+2,\langle n_3\rangle,y_3,y_4)=F$$

所以在乙指派下，对于任何$(\langle n_1\rangle,\langle n_2\rangle,\langle n_3\rangle)$均有

$$\forall y_3\forall y_4\beta(1,\langle n_1\rangle,\langle n_2\rangle,4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,0)+2,\langle n_3\rangle,y_3,y_4)=F$$

又因为对于任何$(\langle n_1\rangle,\langle n_2\rangle),\langle n_3\rangle$均穷尽一切自然数，

所以在乙指派下，对于任何$(\langle n_1\rangle,\langle n_2\rangle)$，$x_3$ 的所有值均使得

$$\forall y_3\forall y_4\beta(1,\langle n_1\rangle,\langle n_2\rangle,4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,0)+2,x_3,y_3,y_4)=F$$

所以，在乙指派下，对于一切$(\langle n_1\rangle,\langle n_2\rangle)$，均有

$$\exists x_3\forall y_3\forall y_4\beta(1,\langle n_1\rangle,\langle n_2\rangle,4\cdot\rho(\langle n_1\rangle,\langle n_2\rangle,0)+2,x_3,y_3,y_4)=F$$

由此可知，在乙指派下，对于任何$(\langle n_1\rangle,\langle n_2\rangle)$，均有$y_2=4\cdot\rho(\langle n_1\rangle,$

$\langle n_2\rangle, 0)+2$, 使得

$$\exists x_3 \forall y_3 \forall y_4 \beta(1, \langle n_1\rangle, \langle n_2\rangle, y_2, x_3, y_3, y_4) = F$$

所以在乙指派下, 对于任何$(\langle n_1\rangle, \langle n_2\rangle)$, 均有

$$\forall y_2 \exists x_3 \exists y_3 \forall y_4 \beta(1, \langle n_1\rangle, \langle n_2\rangle, y_2, x_3, y_3, y_4) = F$$

因为对于任何$\langle n_1\rangle, \langle n_2\rangle$均穷尽一切自然数, 所以在乙指派下, 对于任何$\langle n_1\rangle$, 所有的 x_2 均使得

$$\forall y_2 \exists x_2 \forall y_3 \forall y_4 \beta(1, \langle n_1\rangle, x_2, y_2, x_3, y_3, y_4) = F$$

所以在乙指派下, 对于任何$\langle n_1\rangle$, 均有

$$\exists x_2 \forall y_2 \exists x_3 \forall y_3 \forall y_4 \beta(1, \langle n_1\rangle, x_2, y_2, x_3, y_3, y_4) = F$$

同样在乙指派下, 有

$$\exists x_1 \exists x_2 \forall y_2 \exists x_3 \forall y_3 \forall y_4 \beta(1, x_1, x_2, y_2, x_3, y_3, y_4) = F$$

最后在乙指派下有

$$\forall y_1 \exists x_1 \exists x_2 \forall y_2 \exists x_3 \forall y_3 \forall y_4 \beta(y_1, x_1, x_2, y_2, x_3, y_3, y_4) = F$$

即在乙指派下 $\tilde{\alpha} = F$

因为 α 与 $\tilde{\alpha}$ 同真假, 所以在乙指派下 $\alpha = F$. 这与本引理的假设矛盾, 故本引理得证.

再引进几个记号.

对于任何 n, 在 β_n 中把每个自然数 i 均换为个体变元 v_i, 所得的公式记为 β_n^*.

因为 $\beta_n = \beta(\langle n_1\rangle, \cdots, \langle n_{s_1}\rangle, a_1(n), \langle n_{s_1+1}\rangle, \cdots, \langle n_{s_2}\rangle, a_2(n),$

$$\cdots, a_k(n), \langle n_{s_k+1}\rangle, \cdots, \langle n_h\rangle)$$

其中 $a_i(n)$表 $k\cdot\rho(\langle n_1\rangle, \cdots, \langle n_{s_i}\rangle, 0, \cdots, 0)+t \qquad (t=1, \cdots, k)$

所以

$$\beta_n^* = \beta(v_{\langle n_1\rangle}, \cdots, v_{\langle n_{s_1}\rangle}, v_{a_1(n)}, v_{\langle n_{s_1+1}\rangle}, \cdots, v_{\langle n_{s_2}\rangle}, v_{a_2(n)},$$

$$\cdots, v_{a_k(n)}, v_{\langle n_{s_k+1}\rangle}, \cdots, v_{\langle n_h\rangle})$$

由 ρ 的意义可以得到如下的性质:

性质 1. 对于任何 n，$a_1(n)$，$a_2(n)$，$\cdots$，$a_k(n)$均互不相同，且 $a_1(n)<a_2(n)<\cdots<a_k(n)$.

性质 2. 对于任何 n，在 β_n 中 $a_i(n)$比其左面的其它数都大，即

$$a_i(n)>\langle n_\mu\rangle \quad (\mu=1,\cdots,s_i)\ (i=1,\cdots,k)$$

且 $\quad a_i(n)>a_\mu(n)\ (\mu=1,\cdots,i-1)\ (i=1,\cdots,k)$

性质 3. 对于任何 m 和 n，对于任何 i 和 j，只要 $i\neq j$，必有 $a_i(m)\neq a_j(n)$.

性质 4. 对于任何 m 和 n，在 β_m 和 β_n 中，如果 $a_i(m)=a_i(n)$ 则$\langle m_1\rangle=\langle n_1\rangle$，$\langle m_2\rangle=\langle n_2\rangle$，$\cdots$，$\langle m_{s_i}\rangle=\langle n_{s_i}\rangle$，因而 β_m 中 $a_i(m)$ 左面的数与 β_n 中 $a_i(n)$左面的数均依次相同.

由这些性质又可得下面的性质：

性质 5. 对于任何 n，在 β_n^* 中，个体变元 $v_{a_1(n)}$，$v_{a_2(n)}$，$\cdots$，$v_{a_k(n)}$均互不相同，且 $v_{a_i(n)}$的足码比其左面其它个体变元的足码都大$(i=1,\cdots,k)$.

性质 6. 对于任何 m 和 n，任何 i 和 j，在 β_m^* 和 β_n^* 中，如果 $i\neq j$ 则 $v_{a_i(m)}$与 $v_{a_j(n)}$必互不相同.

性质 7. 对于任何 m 和 n，在 β_m^* 和 β_n^* 中，如果有某个 l 满足 $a_l(m)=a_l(n)$，则 $v_{a_l(m)}$左面各变元与 $v_{a_l(n)}$左面各变元均依次相同.

性质 8. 如果$a_l(m)=a_l(n)$且 β_m^* 中 $v_{a_l(m)}$右面各变元和 β_n^* 中 $v_{a_l(n)}$右面各变元都变得依次相同，那么 β_m^* 与 β_n^* 便完全相同.

令 $\quad \gamma_n^*=\beta_0^*\vee\beta_1^*\vee\cdots\vee\beta_n^*$

$\quad\ \delta_n=\Delta\gamma_n^*$

即 δ_n 是 γ_n^* 的全称封闭式.

又把形如 $\varphi_0\vee\varphi_1\vee\cdots\vee\varphi_n$ 的析取式暂记为 $\varphi_i[i=0\vee\cdots\vee i=n]$. 因此 $\gamma_n^*=\beta_i^*[i=0\vee\cdots\vee i=n]$.

利用这些性质和记号来证引理 2.

引理 2 对于任何公式 α, 任何自然数 n, 均有 $\delta_n \supset \alpha$ 为本谓词演算公理系统中的可证公式.

[证]

$$\delta_n = \Delta\gamma_n^*$$

$$= \Delta(\beta_i^*[i=0 \vee \cdots \vee i=n])$$

$$\supset \beta_i^*[i=0 \vee \cdots \vee i=n]$$

$$= \beta(v_{\langle i_1\rangle}, \cdots, v_{\langle i_{s_1}\rangle}, v_{a_1(i)}, \cdots, v_{a_k(i)}, v_{\langle i_{s_k+1}\rangle}, \cdots, v_{\langle i_k\rangle})$$

$$[i=0 \vee \cdots \vee i=n]$$

利用公理 51 和定理 137 可得

$$\supset \exists x_{s_k+1} \cdots \exists x_k \beta(v_{\langle i_1\rangle}, \cdots, v_{\langle i_{s_1}\rangle}, v_{a_1(i)}, \cdots, v_{a_k(i)}, x_{s_k+1}, \cdots, x_k)[i=0 \vee \cdots \vee i=n]$$

暂记为 $\exists x_{s_k+1} \cdots \exists x_k \beta_{a_k(i)}^*$

$$= \exists x_{s_k+1} \cdots \exists x_k \beta_{a_k(i)}^*[i=0 \vee \cdots \vee i=n]$$

检查 $a_k(0), \cdots, a_k(n)$ 中有无相等者. 若有, 则由性质 8 知, 相应的 β^* 便完全相同, 因此可利用 $(\varphi \vee \varphi) \equiv \varphi$ 把相同的 β^* 合并起来.

$$\equiv \exists x_{s_k+1} \cdots \exists x_k \beta_{a_k(i)}^*[i=0 \vee \cdots \vee i=n_1] \quad (n_1 \leqslant n)$$

这里 $a_k(0), \cdots, a_k(n_1)$ 均互不相等. 由性质 5.6.7 可知, 对于任何 i, $v_{a_k(i)}$ 与 $\beta_{a_k(i)}^*$ 中的其它变元以及其它 β^* 中的任何变元都不相同. 因此利用全称规则改名规则以及定理 514 可得

$$\supset \forall y_k \exists x_{s_k+1} \cdots \exists x_k \beta(v_{\langle i_1\rangle}, \cdots, v_{\langle i_{s_1}\rangle}, v_{a_1(i)} \cdots, v_{a_{k-1}(i)}, v_{\langle i_{s_{k-1}+1}\rangle}, \cdots, v_{\langle i_{s_k}\rangle}, y_k, x_{s_k+1}, \cdots, x_k)$$

$$[i=0 \vee \cdots \vee i=n_1]$$

重复上面的推导过程, 最后可得

$$\supset \exists x_1 \cdots \exists x_{s_1} \forall y_1 \exists x_{s_1+1} \cdots \exists x_{s_2} \forall y_2 \cdots \forall y_k \exists x_{s_k+1} \cdots \exists x_k$$

$$\beta(x_1, \cdots, x_{s_1}, y_1, x_{s_1+1}, \cdots, x_{s_2}, y_2 \cdots, y_k, x_{s_k+1}, \cdots, x_k)$$
$$=\tilde{\alpha}$$
$$\equiv\alpha$$

利用引理 1 和引理 2 便可证下面的哥德尔完备性定理了.

定理 如果 α 是谓词演算永真公式，则在本系统中 α 可证.

[**证**] 因为 α 永真，所以由引理 1 知必有 n 使得 $\tilde{\gamma}_n$ 为命题演算的永真公式，因而 γ_n^* 也为命题演算的永真公式(注意应把其中的谓词填式看作命题变元)，故 γ_n^* 在命题演算公理系统中可证，因而 γ_n^* 的全称封闭式 $\Delta\gamma_n^*$(即 δ_n) 在本谓词演算公理系统中可证. 由引理 2 知，α 可证.

定理 本公理系统不是绝对完备的.

[**证**] 我们知道

$$\exists x\alpha(x)\supset\forall x\alpha(x)$$

不是永真公理，但是 1 永真.

现把该公式作为公理加入本公理系统. 可以证明新的公理系统是不矛盾的. 这是因为原来的 17 条公理都是 1 永真的，三条规则又都保持 1 永真性，从而在新系统中所能推得的定理都是 1 永真公式，亦即非 1 永真的公式是推不出来的，故新系统是不矛盾的. 由绝对完备性的定义知，本系统不是绝对完备的.

§2.8 函数和摹状词

前面我们引出了个体、谓词、量词等概念. 利用这些概念基本上能把一切日常语句翻译成谓词演算公式，但仍有一些不能翻译. 为此我们引出函数和摹状词等概念.

2.8.1 函数

所谓(项值)函数是指以个体为变目，以个体为值的函数. 例如，“小华的父亲”，“x 的父亲”这两个语句中的“的父亲”是一元函

数;“今天南京的温度”,“x 时 y 地的温度”,“3 与 4 的乘积”,“x 与 y 的乘积”这些短语中的“的温度”、“的乘积”都是二元函数. 通常用 f, g, h 等符号表示函数,其变目在函数之后, 例如 f 表“的父亲”,则“小华的父亲”可表为 f(小华), “x 的父亲”可表为 $f(x)$. 在函数后的变目处填以个体或相当于个体的式子, 所得的结果式称为*函数填式*. 例如 f(小华), $f(x)$就是函数填式. 函数和函数填式是两个截然不同的概念. 函数填式是指个体. 例如 f(小华) 是指一特定的个体, $f(x)$也指个体,且随 x 变化. 但是函数就不是个体,而是一种映射,是个体到个体的映射. 为了明显地指出函数的元数. 我们有时在函数之后填以命名变元 e_1, e_2, e_3 等, 例如 $g(e_1, e_2)$表明 g 为二元函数. 函数后填以命名变元的式子叫做*函数命名式*. 函数命名式与函数是完全相同的概念, 区别仅仅在于一个指明了元数, 一个没有指明元数. 因此函数命名式与函数填式是两个截然不同的概念,不能混淆.

函数填式也有两种,一种不含变元的(例如 f(小华)), 一种是含有变元的 (例如 $f(x)$). 通常把含有变元的函数填式叫做*依变元*(或因变元),而填式中的变元叫做*自由变元*,而不含变元的函数填式叫做该函数在某个个体处的*值*. 因此依变元与函数是两个截然不同的概念,不能混为一谈,但是通常的数学书中常把这两者混淆在一起,同一式子有时表示函数,有时表示依变元,例如,“n^2+1 是递增函数”,这个句子提及的 n^2+1 指的是函数, 但是“n^2+1 永大于 0”这个句子中提及的 n^2+1 指的是依变元. 造成这种混乱的主要原因在于用相同的符号表示命名变元和自变元,上例中,第一句中的 n 是指命名变元,因此 n^2+1 是函数命名式, 第二句中的 n 是指自变元,因此 n^2+1 是函数填式, 也即是依变元. 用同一符号表示命名变元和自变元也有其方便之处, 在下一章中就带有这种情形. 这就要求大家能严格分清各种情况.

现在我们举例说明使用函数概念翻译日常语句的办法.

例 1: 我找不到你的小弟弟决不回家.

[**解**] 设Ae_1e_2表"e_1找到e_2", Be表"e回家", fe表"e的小弟弟", a表"我", b表"你", 则全句可译为

$$\bar{A}afb\supset Bb$$

例2: 杭州是浙江省的省会.

[**解**] 设fe表"e的省会", a表"杭州", b表"浙江省". 则全句可译为

$$a=f(b)$$

2.8.2 唯一性量词

日常的语句中常常出现"只有一个"、"恰巧有一个"之类的词句, 对此我们引进唯一性量词"$\exists_!$"来表示它们.

$\exists_! x\alpha(x)$读为恰巧有一个x使得$\alpha(x)$. $\exists_!$称为唯一性量词(又称恰有量词), x称为唯一性量词的指导变元, $\alpha(x)$称为该量词的作用域.

有了唯一性量词后, 有些语句可如下翻译:

例 3: 只有一个人没有去过上海.

[**解**] 设Ae表"e为人", Be_1e_2表"e_1去过e_2", a表"上海", 则全句可译为

$$\exists_! x(Ax\wedge\bar{B}xa)$$

例 4: 他是唯一没有去过上海的人.

[**解**] 设b表"他", 其余同上, 则全句可译为

$$\exists_! x(Ax\wedge\bar{B}xa\wedge x=b)$$

由唯一性量词的定义可知, 就某个体域I而言, $\exists_! x\alpha(x)$真当且仅当在I中有一个也只有一个个体使得$\alpha(x)$真.

另外还有下面的同真假式:

$\exists_! x\alpha(x)$与$\exists x(\alpha(x)\wedge\forall y(x\neq y\supset\bar{\alpha}(y)))$同真假

(注意,这里引入了常谓词"=")

$\exists_1 x\alpha(x)\vee\gamma$ 与 $\exists_1 x(\alpha(x)\vee\gamma)$同真假

(其中 γ 不含自由 x)

$\exists_1 x\alpha(x)\wedge\gamma$ 与 $\exists_1 x(\alpha(x)\wedge\gamma)$同真假

(其中 γ 不含自由x)

显然，利用这些同真假式可把唯一性量词化为全称性量词和存在量词.

2.8.3 摹状词

日常语句中还常见到"××东西的发明者"，"××书的作者"等这一类的短语,它们都是用来表示一个特定的个体,而确定的方法是利用个体的特征性质来指定，这个特征性质应为其它个体所不具有. 例如，"纸的发明者"是指使得"x 发明纸" 成立的那个个体x,即"蔡伦",又如"三国演义的作者"是指使得"x著作三国演义"成立的那个个体 x,即"罗贯中".尽管"蔡伦" 和纸的发明者指相同的个体,但两者的涵义却截然不同，不能交换乱用. 例如"蔡伦发明纸"告诉了我们一些历史知识,但是"纸的发明者发明纸"却是永真语句,与历史知识无关.

由此可见,上面这些短语与专名是不同的,其不同点在于，专名除对一个个体给以标记以外没有别的作用，但上面这类短语除确定一个个体外,还告诉我们有关该个体的特征性质,这种性质是为其它个体所没有的. 为了表示这种性质，特引人摹状词 ιx.

$\iota x\alpha(x)$读为: 使得 $\alpha(x)$ 成立的那个唯一的个体其中"ι"叫做*摹状词*,"x"叫做该摹状词的*指导变元*，"$\alpha(x)$"叫做该摹状词的*作用域*.

应该注意的是，摹状词的作用域与量词的作用域一样都是谓词演算公式,但摹状词的值是个体,而量词的值是真假. 还应该注意,使用摹状词时必须满足存在唯一性，否则便无所指. 例如"锄

头的发明者"就不满足唯一性，这个个体是指不出来的，又如"地球的创造者"就不满足存在性，这个个体也是指不出来的.

存在唯一性条件太严，使用不方便，因此我们引入下述概念.

$\iota_x^y\alpha(x)$读为：当 $\exists!\, x\alpha(x)$ 成立时指使得 $\alpha(x)$成立的那个个体，当 $\exists!\, x\alpha(x)$不成立时指 y

显然使用这个摹状词，可不必管是否满足存在唯一性，使用时比较方便.

下面我们举例说明使用摹状词翻译日常语句的方法.

例 5：并非读书最多的人最有知识.

[解] 设 Ae 表"e 为人"，Be_1e_2 表"e_1 比 e_2 读书多"Ce_1e_2 表"e_1 比 e_2 有知识"

则"读书最多的人"可译为

$$\iota_x^y(Ax\wedge\forall z((Az\wedge z\neq x)\supset Bxz))$$

简记为 a，故全句可译为

$$\bar{\forall}t((At\wedge t\neq a)\supset Cat)$$

显然，由摹状式的定义可知，下式是成立的.

$$\beta(\iota_x^y\alpha(x))=(\bar{\exists}!\, x\alpha(x)\wedge\exists t(\beta(t)\wedge\alpha(t)))\vee(\bar{\exists}!\, x\alpha(x)\wedge\beta(y))$$

由此可知，任何一个含有摹状词的公式均可化归为一个不含摹状词的公式.

总结上面所述可知，命题联结词是把命题变为命题，谓词是把个体变为命题，(项值)函数是把个体变为个体，量词的作用域是命题，指导变元是个体变元，结果为命题，而摹状词的作用域是命题，指导变元是个体变元，结果变为个体.

2.8.4 扩大的狭义谓词演算公式

有了函数、唯一性量词和摹状词，可以把前面的谓词演算公式的定义(见 § 2.5)修改如下：

命题变元: (同前)

个体变元: (同前)

谓词变元: (同前)

函数变元: 一元的 $F^1, G^1, H^1, F_1^1, F_2^1, \cdots$

二元的 $F^2, G^2, H^2, F_1^2, F_2^2, \cdots$

n 元的 $F^n, G^n, H^n, F_1^n, F_2^n, \cdots$

联结词: (同前)

量词: $\forall$ (全称量词), $\exists$ (存在量词)

$\exists!$ (唯一性量词)

摹状词: ι

公式及项: 命题变元为公式; 个体变元为项; 如果X为n元谓词, $\zeta_1, \cdots, \zeta_n$是n个项, 则 $X(\zeta_1, \cdots, \zeta_n)$为公式(谓词填式); 如果$F$为$n$元函数, $\zeta_1, \cdots, \zeta_n$是项, 则 $F(\zeta_1, \cdots, \zeta_n)$为项(函数填式); 如果$\alpha, \beta$为公式, 则 $\bar{\alpha}$, $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \equiv \beta)$, $(\alpha \supset \beta)$均为公式; 如果α是公式, x、y为个体变元, 则 $\forall x(\alpha)$、$\exists x(\alpha)$、$\exists! x(\alpha)$是公式, 而 $\iota \frac{y}{x}(\alpha)$是项.

习　题

把下列语句翻译成谓词演算公式:

1. 北京是中华人民共和国的首都.
2. 任何数的平方必$\geqslant 0$, 除非它为虚数.
3. 反对这个提案的人只有一个.
4. 地球是唯一有人的星球; 地球并非是唯一有人的星球.
5. 昨天访问你的人留下一封信.
6. 最后一个离开教室的人关门窗关电灯.

把下列公式中的唯一性量词和摹状词消去:

7. $\exists! x(Xx \wedge \exists! y Yxy)$
8. $Xx \vee \iota y(Yxy \wedge \bar{Z}xy)$

§2.9 约束谓词演算和应用谓词演算

2.9.1 狭义谓词演算与约束谓词演算

在前面的讨论中，我们引进了量词和摹状词，它们的指导变元都是个体变元．我们没有讨论指导变元为命题变元、函数变元和谓词变元的量词和摹状词．现在我们来简单的讨论一下。

先看指导变元为命题变元的情形．

因为命题变元是以真假为变域的变元，所以 $\forall p\alpha(p)$ 和 $\exists p\alpha(p)$ 分别等价于 $\alpha(T)\wedge\alpha(F)$ 和 $\alpha(T)\vee\alpha(F)$．又因为 T 和 F 分别等价于 $p\vee\bar{p}$ 和 $p\wedge\bar{p}$，所以 $\forall p\alpha(p)$ 和 $\exists p\alpha(p)$ 分别等价于 $\alpha(p\vee\bar{p})\wedge\alpha(p\wedge\bar{p})$ 和 $\alpha(p\vee\bar{p})\vee\alpha(p\wedge\bar{p})$．由此可知，命题变元的量词是可以消去的．同样其它的命题变元的量词（如唯一性量词等），命题变元的摹状词也都是可以消去不用的．因此一般不讨论命题变元的量词摹状词．

再看指导变元为谓词变元和函数变元的情形．

$\forall X\alpha(X)$，$\exists X\alpha(X)$，$\forall G\alpha(G)$ 和 $\exists G\alpha(G)$ 这四个公式的含义分别为“任何谓词 X 均使得 $\alpha(X)$”，“有谓词 X 使得 $\alpha(X)$”，“任何函数 G 均使得 $\alpha(G)$” 和 “有函数 G 使得 $\alpha(G)$”．有无必要引进这些约束谓词变元和约束函数变元呢？从数学和日常语言看这是必要的．因为在数学和日常生活中常常要使用关于谓词变元和函数变元的量词．例如我们知道下面两命题是成立的：

1．对任何谓词 X 均有 $\exists x\forall yXxy\supset\forall y\exists xXxy$ 真．

2．并非对任何谓词 X 均有 $\forall y\exists xXxy\supset\exists x\forall yXxy$ 真．

若要把这两个命题用逻辑符号表达出来就必须使用关于谓词变元的量词，它们可分别表为：

1．$\forall X(\exists x\forall yXxy\supset\forall y\exists xXxy)$

2．$\overline{\forall}X(\forall y\exists xXxy\supset\exists x\forall yXxy)$

我们把只讨论关于个体变元的量词、摹状词以及其它约束词的谓词演算称为*狭义谓词演算*，把讨论具有谓词变元和函数变元的量词、摹状词和其它约束词的谓词演算称为*约束谓词演算*.

2.9.2 纯逻辑演算与应用逻辑演算

在前面讨论的命题演算系统和谓词演算系统中，除了真值联结词、量词、摹状词外，都只限于使用个体变元、命题变元、谓词变元、函数变元，而不允许使用常个体(如"张三"，"15")、常命题(如"雪是白的"，"张三是学生")、常谓词(如"≥"，"是兄弟")、常函数(如"+"、"−")．但是在讨论实际问题时却经常遇到常个体、常命题、常谓词、常函数.

凡是只使用变元和真值联结词、量词、摹状词而不再使用别的常量的演算都称为*纯逻辑演算*.

变量和常量均使用的逻辑演算称为*应用逻辑演算*.

由定义可知，前面我们讨论的命题演算和谓词演算都是纯逻辑演算．算术、几何、群论等各门数学则属于应用逻辑演算．例如算术是讨论自然数(即常个体)和+、−、×、÷四则运算(即常函数)的应用逻辑演算．几何是讨论常谓词"在上"、"介于"、"合同于"等应用逻辑演算.

任何一个应用逻辑演算系统都需要在前面给出的纯谓词演算或纯命题演算系统基础上添加一些新的公理或规则．作为应用逻辑演算系统的一个例子，下面我们简单介绍一下相等性演算.

2.9.3 相等性演算

只含常谓词"="而不再含别的常谓词常函数的应用逻辑演算叫做*相等性演算*.

下面是不含量词和摹状词的相等性演算公理系统.

组成部分

个体变元：$x, y, z, x_1, y_1, z_1, x_2, y_2, z_2, \cdots$

命题变元: $p, q, r, p_1, q_1, r_1, p_2, q_2, r_2, \cdots$

谓词变元: $X, Y, Z, X_1, Y_1, Z_1, X_2, Y_2, Z_2, \cdots$

函数变元: $F, G, H, F_1, G_1, H_1, F_2, G_2, H_2, \cdots$

常谓词: $=$

真值联结词: $\neg, \wedge, \vee, \supset, \equiv$

括号: $(,)$

项: 个体变元是项; 如果 F 是 n 元函数变元, $\xi_1, \cdots, \xi_n$ 是项, 则 $F(\xi_1, \cdots, \xi_n)$ 是项

公式: 命题变元是公式; 如果 ξ_1, ξ_2 是项, 则 $(\xi_1 = \xi_2)$ 是公式; 如果 X 是 n 元谓词变元, $\xi_1, \cdots, \xi_n$ 是项, 则 $X(\xi_1, \cdots, \xi_n)$ 是公式; 如果 α, β 是公式, 则 $\bar{\alpha}, (\alpha \wedge \beta), (\alpha \vee \beta), (\alpha \supset \beta), (\alpha \equiv \beta)$ 是公式. 所谓项和公式仅限于此.

推理部分

公理: 10～24°(同前)

60 $\xi = \xi$ 其中 ξ 为任意的项

61 $\xi = \eta \supset (\alpha(\xi) \supset \alpha(\eta))$ 其中 ξ, η 是任意的项, $\alpha(\xi)$ 是任何的公式, $\alpha(\eta)$ 是在 $\alpha(\xi)$ 中用 η 替换以 ξ 的结果.

规则: 分离规则 (同前)

定理: (同前)

由这个公理系统容易得到下面的导出规则

分分 61 规则: $\xi = \eta, \alpha(\xi) \vdash \alpha(\eta)$

这个规则称为相等替换规则, 简记为"等替".

下面两条定理也是容易得到的:

601 $\xi = \eta \supset \eta = \xi$ (分分 11, 61, 60)

602 $\xi = \eta \supset (\eta = \zeta \supset \xi = \zeta)$ (分分 12, 601, 61)

§2.10 应用——程序的部分正确性证明

前面我们讲述了命题演算及谓词演算中的基本概念和重要性质，建立了它们的公理系统，讨论了系统的不矛盾性、完备性等等．表面上看这些内容都十分抽象，似乎不会有什么实用价值．其实不然，上面讲的形式化公理化的方法在科学研究中有着十分重要的作用，对计算机科学尤其如此．数理逻辑的理论和方法可以使我们更准确地理解程序，也可使我们较为方便容易地构造程序．现在人们正在计算机科学中进一步发展这些方法，并用来证明程序的正确，检测程序中的错误，改善程序运行的效率，扩充或修改现存的错误，甚至构造满足给定规范要求的程序．这是一个非常活跃的研究领域．可以预料，这些研究将会使计算机程序的产生方式发生巨大的变化．

本节我们仅简单介绍逻辑方法在程序正确性证明上的一些应用．我们之所以叙述程序正确性证明问题，首先是因为它是每个软件人员都会遇到的问题．任何人都希望自己编写的程序正确无误，能得到预期的结果．可是实际上却不容易做到，不仅初学者做不到这一点，就是经验丰富的程序员也不能确保他自己的程序个个正确．其次是因为程序中的错误有时会造成严重的后果．例如1962年6月美国曾发射一颗飞向金星的“水星1号”宇宙飞船，由于飞船里的一个导航程序中的一个错误，致使飞船偏离航线而使整个计划遭到破坏．该程序中有一语句，语法上是正确的，语义上却与程序员所期望的完全不同．虽然后果如此严重的错误并不多，但是程序中的错误是常有的，而且是有影响的．第三是因为传统的查找程序中错误的方法——调试方法不能肯定程序一定正确，调试法只能肯定程序存在错误，不能肯定程序中没有错误．第四是因为这个问题是目前研究得较多理解得最好的问题，而且已经

有一些实验性的程序验证系统，因而有内容可以介绍．最后是因为这个课题目前仍然存在不少问题有待人们研究解决．

所谓程序正确性证明是指证明一给定的程序对于任何合法的输入而执行该程序时最终一定停止，而且产生预期的结果，也即需要证明两件事：一是证明程序最终一定停止，二是证明产生预期的效果．前者称为程序的*终止性问题*，后者称为程序的*部分正确性问题*．终止性加上部分正确性就是程序的*完全正确性问题*．本节只讨论程序的部分正确性问题．

我们知道，任何一个程序其目的都是对一组输入数据（或称初始数据）进行加工计算，最后得到所要求的结果（称为输出数据或结果数据）．设 P 是一个程序，$x_1, \cdots, x_n$（记为 $\vec{x}$）是它的输入数据，$y_1, \cdots, y_n$（记为 $\vec{y}$）是它的输出数据．又设 BODY P 是该程序的程序体．一般 P 呈下面的形式：

```
program P;
read(x⃗);
BODY P;
write(y⃗)
end P
```

通常输入数据是要满足一定条件的．输入数据必须满足的条件称为程序的*输入断言*．同样输出数据也必需满足一定的条件，即必须是所要求的结果．输出数据必须满足的条件称为程序的*输出断言*．设 INAP($\vec{x}$)是程序 P 的输入断言，OUTAP($\vec{x}, \vec{y}, \vec{z}$)是程序 P 的输出断言，其中 $\vec{z}$ 是程序的中间数据．P 可以改写如下：

```
program P;
read(x⃗);
{INAP(x⃗)}
BODY P;
```

{OUTAP($\vec{x}$, $\vec{y}$, $\vec{z}$)}

write($\vec{y}$)

end P

应该注意的是{INAP($\vec{x}$)}和{OUTAP($\vec{x}$, $\vec{y}$, $\vec{z}$)}是程序中的非执行部分. 它们的作用是指明输入数据和输出数据应满足的条件,以便让人们检验输入数据和输出数据是否正确. 也就是说,它们仅被用来验证程序的正确性. 所以我们特地用花括号括起来以示区别. 这样程序的部分正确性证明问题可以表述如下:

对于任何 $\vec{x}$, 任何 $\vec{y}$ 和任何 $\vec{z}$, 如果执行 BODY P 前 INAP($\vec{x}$)真, 且 BODYP 执行一定终止, 则执行 BODYP 后 OUTAP($\vec{x}$, $\vec{y}$, $\vec{z}$)真.

这是一种非形式的描述. 我们约定它的形式描述是

(1) {INAP($\vec{x}$)}BODY P{OUTA P($\vec{x}$, $\vec{y}$, $\vec{z}$)}

即上面程序中第三行到第五行的部分. (1) 称为上面程序的验证公式.

一般地说,设 P 是一个程序或一个程序段或一个程序语句, A 和 B 是两个断言语句 (或为输入断言或为输出断言或为其它断言), 则形式公式

{A} P {B}

称为 P 的验证公式, 其含义是"如果执行 P 前 A 真, 且 P 的执行一定终止则执行 P 后 B 真".

现在我们用一个实际例子来作进一步的阐明. 下面是计算两个非负且不同时为零的整数 x_1 和 x_2 的最大公约数 y 的程序.

program GCD;

read(x_1, x_2);

(z_1, z_2) := (x_1, x_2);

while $z_1 \neq 0$ **do**

if $z_2 \geqslant z_1$ **then** $z_2 := z_2 - z_1$ **else** $(z_1, z_2) := (z_2, z_1)$

od;

$y := z_2$;

write(y)

end GCD

其中“$(z_1, z_2) := (x_1, x_2)$”意思是“变量 z_1 和 z_2 分别且同时赋以 x_1 和 x_2 的值”. 于是“$(z_1, z_2) := (z_2, z_1)$”有交换 z_1 和 z_2 的值的作用.

该程序的输入断言应是“x_1 和 x_2 是非负的且不同时为零的整数”, 即为

“$x_1 \geqslant 0 \wedge x_2 \geqslant 0 \wedge (x_1 \neq 0 \vee x_2 \neq 0)$”

注意, 这里省略了“x_1 和 x_2 为整数”, 这可以理解为在本节中始终假定变量取整数值. 该程序的输出断言应是“y 是 x_1 和 x_2 的最大公约数”, 即

“$y = \mathrm{MAXu}(u \div x_1 \wedge u \div x_2)$”

其中“$e_1 \div e_2$”是二元谓词, 表示“e_1 除尽 e_2”; “MAXu $\alpha(u)$”表示“使得 $\alpha(u)$ 成立的一切 u 中的最大者”. 这样带有输入输出断言的求最大公约数的程序是

program GCD;

read (x_1, x_2);

$\{x_1 \geqslant 0 \wedge x_2 \geqslant 0 \wedge (x_1 \neq 0 \vee x_2 \neq 0)\}$

$(z_1, z_2) := (x_1, x_2)$;

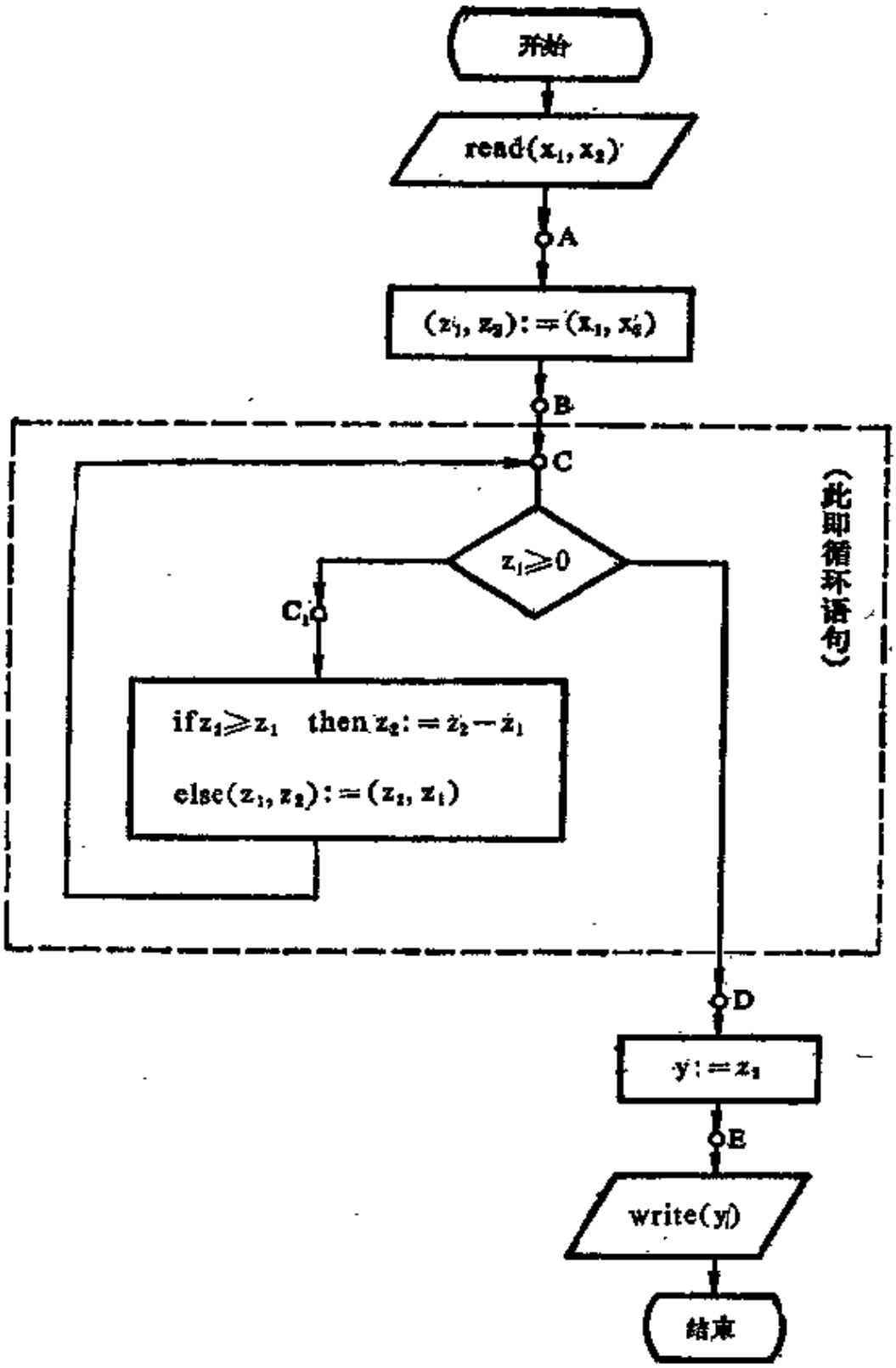
while $z_1 \neq 0$ **do**

if $z_2 \geqslant z_1$ **then** $z_2 := z_2 - z_1$ **else** $(z_1, z_2) := (z_2, z_1)$

od;

$y := z_2$;

$\{y = \mathrm{MAXu}(u \div x_1 \wedge u \div x_2)\}$


开始
read(x1, x2)
A
(z1, z2) := (x1, x2)
B
C
z1≥0
C1
if z1≥z1 then z2 := z2−z1
else(z1, z2) := (z2, z1)
(此即循环语句)
D
y := z2
E
write(y)
结束

求最大公约数的流程图

write (y)

end GCD

该程序的验证公式是

(2) $\{x_1 \geqslant 0 \wedge x_2 \geqslant 0 \wedge (x_1 \neq 0 \vee x_2 \neq 0)\}$

$(z_1, z_2) := (x_1, x_2)$;

while $z_1 \neq 0$ **do**

if $z_2 \geqslant z_1$ **then** $z_2 := z_2 - z_1$ **else** $(z_1, z_2) := (z_2, z_1)$

od;

$y := z_2$

$\{y = MAXu(u \div x_1 \wedge u \div x_2)\}$

现在的问题是如何来证明(2)是永真的. 为使问题逐步深入，我们先给出一个非形式的数学论证，然后再给出形式的推理系统.

(2)是否永真的问题并非一目了然. 重要的是必须对程序有比较深入的理解. 为了理解该程序，我们把该程序改成流程图的形式(见127页).

在这个流程图中，当控制达到 A 点时，输入断言应成立. 当控制达到 E 点时，输出断言应成立. 当控制达到 B 点、C 点和 D 点时情形怎样呢？先看 B 点. 当控制由 A 点达到 B 点时，z_1, z_2 的最大公约数应该就是 x_1, x_2 的最大公约数，且 z_1, z_2 也应该是不同时为 0 的非负整数. 这就是说，在 B 点应有下面的断言成立:

$$z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge MAXu(u \div z_1 \wedge u \div z_2) = MAXu(u \div x_1 \wedge u \div x_2)$$

再看 C 点. 控制既可由 B 点达到 C 点，也可以由 C 点经过 C_1 点回到 C 点，而且可以循环许多次. 在循环过程中，z_1 和 z_2 的值是在不断改变的. 尽管如此，控制达到 C 点时，z_1 和 z_2 的最大公约数却始终不变，而且恰好就是 x_1 和 x_2 的最大公约数，此外 z_1 和 z_2 仍旧是不同时为 0 的非负整数. 这就是说，在 C 点仍然有和

B 点处的断言完全相同的断言成立，即

$$z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge \mathrm{MAXu}(u\div z_1\wedge u\div z_2)$$
$$=\mathrm{MAXu}(u\div x_1\wedge u\div x_2)$$

最后看 D 点．当控制达到 D 点时，循环已结束．这时 z_2 的值应为 x_1 和 x_2 的最大公约数，即在 D 点应有下面的断言成立

$$z_2=\mathrm{MAXu}(u\div x_1\wedge u\div x_2)$$

因为 B、C、D 点在程序中间，所以把这些点处的断言称为程序的*中间断言*．又因为 C 点在循环语句内部，而且 C 点处的断言在循环过程中始终保持不变，所以通常把这种断言称为循环语句的*不变断言*，又称为循环语句的*不变关系式*，简称为*不变式*．现在(2)可进一步改为

(3)
$\{x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0)\}$
$(z_1, z_2):=(x_1, x_2);$
$\{z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge$
$\mathrm{MAXu}(u\div z_1\wedge u\div z_2)=\mathrm{MAXu}(u\div x_1\wedge u\div x_2)\}$
while $\{z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge$
$\mathrm{MAXu}(u\div z_1\wedge u\div z_2)=\mathrm{MAXu}(u\div x_1\wedge u\div x_2)\}$
$z_1\neq 0$ **do**
if $z_2\geqslant z_1$ **then** $z_2:=z_2-z_1$ **else** $(z_1, z_2):=(z_2, z_1)$
od;
$\{z_2=\mathrm{MAXu}(u\div x_1\wedge u\div x_2)\}$
$y:=z_2;$
$\{y=\mathrm{MAXu}(u\div x_1\wedge u\div x_2)\}$

由上面流程图可看出，我们可以把该流程图分解成三段：(A, B)段，(B, D)段和(D, E)段．同样由(3)也可看出，可以把对(3)的证明分解成下面(4)、(5)、(6)三个式子来证明．

(4) $\{x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0)\}$

$(z_1, z_2) := (x_1, x_2)$;

$\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$

$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2) = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

这就是流程图中(A, B)段的情形.

(5) $\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$

$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2) = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

while $\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$

$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2) = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

$z_1 \neq 0$ **do**

if $z_2 \geqslant z_1$ **then** $z_2 := z_2 - z_1$ **else** $(z_1, z_2) := (z_2, z_1)$

od;

$\{z_2 = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

这就是流程图中(B, D)段的情形.

(6) $(z_2 = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

$y := z_2$

$\{y = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

这就是流程图中(D, E)段的情形.

验证公式(4)的含义是:"如果 A 点处的输入断言真,而且 z_1、z_2 是执行 $(z_1, z_2) := (x_1, x_2)$ 的结果,则 B 点处的断言真。"因为赋值语句把 z_1 置为 x_1, z_2 置为 x_2, 所以(4)可写成

(7) $(x_1 \geqslant 0 \wedge x_2 \geqslant 0 \wedge (x_1 \neq 0 \vee x_2 \neq 0))$

$\supset (x_1 \geqslant 0 \wedge x_2 \geqslant 0 \wedge (x_1 \neq 0 \vee x_2 \neq 0) \wedge$

$\mathbf{MAXu}(u \div x_1 \wedge u \div x_2) = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2))$

(7)中的蕴涵后件是由 B 点处的中间断言中 z_1 换成 x_1 和 z_2 换成 x_2 形成的. 这个逻辑公式当然永真.

(6)的含义是:

$(z_2 = \mathbf{MAXu}(u \div x_1 \wedge u \div x_2) \wedge y$ 是执行 $y := z_2$ 的结果)

$$\supset(y=MAXu(u\div x_1 \wedge u\div x_2))$$

因为执行 $y:=z_2$ 后 y 的值就是 z_2，所以这个式子可写成

(8) $$(z_2=MAXu(u\div x_1 \wedge u\div x_2))$$
$$\supset MAXu(u\div x_1 \wedge u\div x_2)$$

其中的蕴涵后件是把 E 点处的输出断言中的 y 换为 z_2 形成的. 这个逻辑公式也当然永真.

(5)的含义是:“如果 B 点处的断言真，则执行循环语句后 D 点处的断言也真.”由流程图可以看出，可把(B, D)段分解成(B, C)段、(C, C_1, C)段和(C, D)段，相应的(5)也可分解成下面三个式子.

(9) $$(z_1\geqslant 0 \wedge z_2\geqslant 0 \wedge (z_1\neq 0 \vee z_2\neq 0) \wedge$$
$$MAXu(u\div z_1 \wedge u\div z_2)=MAXu(u\div x_1 \wedge u\div x_2))$$
$$\supset(z_1\geqslant 0 \wedge z_2\geqslant 0 \wedge (z_1\neq 0 \vee z_2\neq 0) \wedge$$
$$MAXu(u\div z_1 \wedge u\div z_2)=MAXu(u\div x_1 \wedge u\div x_2))$$

(10) $$(z_1\geqslant 0 \wedge z_2\geqslant 0 \wedge (z_1\neq 0 \vee z_2\neq 0) \wedge$$
$$MAXu(u\div z_1 \wedge u\div z_2)=MAXu(u\div x_1 \wedge u\div x_2)$$
$$\wedge z_1\neq 0)\supset(\text{执行条件语句 } \mathbf{if}\ z_2\geqslant z_1\ \mathbf{then}\ z_2:=z_2-z_1$$
$$\mathbf{else}(z_1, z_2):=(z_2, z_1)\text{后 C 点处的断言 } z_1\geqslant 0 \wedge z_2\geqslant 0$$
$$\wedge (z_1\neq 0 \vee z_2\neq 0) \wedge MAXu(u\div z_1 \wedge u\div z_2)=MAXu(u$$
$$\div x_1 \wedge u\div x_2))$$

(11) $$(z_1\geqslant 0 \wedge z_2\geqslant 0 \wedge (z_1\neq 0 \vee z_2\neq 0) \wedge$$
$$MAXu(u\div z_1 \wedge u\div z_2)=MAXu(u\div x_1 \wedge u\div x_2)$$
$$\wedge z_1=0)\supset(z_2=MAXu(u\div x_1 \wedge u\div x_2))$$

逻辑公式(9)显然永真. 为了证明(10)，我们把条件语句中的条件区分为两种情形: $z_2\geqslant z_1$ 和 $z_1>z_2$. 对于 $z_2\geqslant z_1$ 的情形，因执行赋值语句 $z_2:=z_2-z_1$，故执行该语句后 z_2 的值为 z_2-z_1. 这时(10)为

(10′) $(z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge \mathbf{MAXu}(u \fallingdotseq z_1 \wedge u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2) \wedge z_1 \neq 0 \wedge z_2 \geqslant z_1) \supset (z_1 \geqslant 0 \wedge z_2 - z_1 \geqslant 0 \wedge (z_1 \neq 0 \vee (z_2 - z_1) \neq 0) \wedge \mathbf{MAXu}(u \fallingdotseq z_1 \wedge u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2))$

因为 $z_1 \neq 0 \supset (z_1 \neq 0 \vee (z_2 - z_1) \neq 0)$永真，

$z_2 \geqslant z_1 \supset z_2 - z_1 \geqslant 0$ 永真，

又因为 $(u \fallingdotseq z_1 \wedge u \fallingdotseq z_2) \equiv (u \fallingdotseq z_1 \wedge u \fallingdotseq (z_2 - z_1))$ 永真(即 z_1 和 $z_2 - z_1$ 的公约数与 z_1 和 z_2 的公约数相同)，从而 $\mathbf{MAXu}(u \fallingdotseq z_1 \wedge u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq z_1 \wedge u \fallingdotseq (z_2 - z_1))$永真，所以(10′)永真.

对于 $z_2 < z_1$ 的情形，该条件语句应执行赋值语句 $(z_1, z_2) := (z_2, z_1)$，即 z_1 与 z_2 的值互换，这时(10)为

(10″) $(z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge \mathbf{MAXu}(u \fallingdotseq z_1 \wedge u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2) \wedge z_1 \neq 0 \wedge z_2 < z_1) \supset (z_2 \geqslant 0 \wedge z_1 \geqslant 0 \wedge (z_2 \neq 0 \vee z_1 \neq 0) \wedge \mathbf{MAXu}(u \fallingdotseq z_2 \wedge u \fallingdotseq z_1) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2))$

这显然是永真的. 既然(10′)和(10″)均永真，(10)也就永真.

现在证明(11). 因为 $u \fallingdotseq 0$ 永真(即任何整数都除尽 0)，又因为 $y > 0 \supset y = \mathbf{MAXu}(u \fallingdotseq y)$永真 (即任何正整数是它自己的最大约数)，所以

$$z_1 = 0, \mathbf{MAXu}(u \fallingdotseq z_1 \wedge u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2), z_2 > 0$$
$$\vdash \mathbf{MAXu}(u \fallingdotseq 0 \wedge u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2)$$
$$\vdash \mathbf{MAXu}(u \fallingdotseq z_2) = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2)$$
$$\vdash z_2 = \mathbf{MAXu}(u \fallingdotseq x_1 \wedge u \fallingdotseq x_2)$$

因而(11)永真.

这就结束了对程序 GCD 的部分正确性证明. 证明中涉及到六个无循环的路径论证：从 A 点到 B 点的一个路径，从 B 点到 C

点的一个路径，从C点经C_1点再回到C点的两个路径，从C点到D点的一个路径，从D点到E点的一个路径．如果我们不引进中间断言，我们就不得不对包含在输入输出断言之间的循环路径进行论证，由于循环次数是不确定的，因而论证是困难的．由此可知，为了使上述的证明方法成功，中间断言是不可少的．由于给出了适当的断言，所以如果程序是部分正确，则所有的验证公式就是真．如果程序不是部分正确，则至少有一个验证公式是假．因而我们已把程序部分正确性证明问题转变成几个逻辑公式的真假证明问题．

综上所述，程序的部分正确性证明由三个阶段组成，首先给出输入输出断言和中间断言，尤其是循环的不变式，第二是产生相应的断言语句，第三是证明断言语句真．第二阶段是一个单纯的机械工作，但第一阶段要求对程序有深刻的理解，第三阶段需要严格的逻辑推理，还需要程序所属领域的知识（例如，公约数的性质）．

上面的讨论是非形式的．下面我们给出一个公理系统，使这些论证形式化．由上面知道，我们要论证的对象是验证公式．在验证公式中涉及到各种常谓词常函数和常个体，因此我们所要建立的系统是一个应用逻辑演算系统．这个系统是与所讨论的程序设计语言密切相关的，是与所论证的程序的所属领域的知识有关的，现在我们假设所讨论的程序设计语言包括有赋值语句、条件语句、循环语句以及语句串．又假设所论证的程序是求最大公约数的程序．

一个验证公式演算系统

公理　10～24　（同前）

　　60, 61　（同前）

规则　分离规则　（同前）

　　赋值规则　$\alpha(u_1, u_2, \cdots, u_n) \supset \beta(v_1, v_2, \cdots, v_n)$

$$\vdash \{\alpha(u_1, \cdots, u_n)\}(u_1, \cdots, u_n) := (v_1, \cdots, v_n);$$
$$\{\beta(u_1, \cdots, u_n)\}$$

其中 α 和 β 是任意的逻辑公式，$\beta(v_1, \cdots, v_n)$是在 $\beta(u_1, \cdots, u_n)$ 中用 v_i 合法替换以 u_i 的结果. 这个规则是相应于赋值语句$(u_1, \cdots, u_n) := (v_1, \cdots, v_n)$的规则. 它告诉我们，如果 $\alpha(\vec{u}) \supset \beta(\vec{v})$真，则必有

$\alpha(\vec{u}) \supset$执行$(\vec{u}) := (\vec{v})$后的 $\beta(\vec{u})$真.

条件规则　$\{\alpha \wedge B\}S_1\{\beta\}, \{\alpha \wedge \overline{B}\}S_2\{\beta\}$
$$\vdash \{\alpha\}\textbf{if } B \textbf{ then } S_1 \textbf{ else } S_2; \ \{\beta\}$$

其中 α, β 和 B 是任意的逻辑公式，S_1 和 S_2 是任意的语句. 这个规则是相应于条件语句

if B then S_1 else S_2 的规则.

循环规则　$\{\alpha \wedge B\}S\{\alpha\}, \alpha \wedge \overline{B} \supset \beta$
$$\vdash \{\alpha\}\textbf{while B do S; od}\{\beta\}$$

其中 α, β 和 B 同上，S 是任意的语句. 这个规则是相应于循环语句 **while B do S od** 的规则.

结合规则　$\{\alpha\}S_1\{\beta\}, \{\beta\}S_2\{\gamma\}$
$$\vdash \{\alpha\}S_1; \ S_2; \ \{\gamma\}$$

其中 α、β 和 γ 是任意的逻辑公式，S_1, S_2 是任意的语句. 这个规则是相应于语句串 S_1; S_2 的规则.

现在我们利用这个系统来证明:

(甲)
$$\{x_1 \geqslant 0 \wedge x_2 \geqslant 0 \wedge (x_1 \neq 0 \vee x_2 \neq 0)\}$$
$$(z_1, z_2) := (x_1, x_2);$$
$$\textbf{while } z_1 = 0 \textbf{ do}$$
$$\textbf{if } z_2 \geqslant z_1 \textbf{ then } z_2 := z_2 - z_1 \textbf{ else}(z_1, z_2) := (z_2, z_1)$$
$$\textbf{od;}$$
$$y := z_2;$$

$\{y=\mathbf{MAXu}(u\div x_1\wedge u\div x_2)\}$

证明: 公理 10=(1) $(x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0))$
$\supset(x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0))$

公理 60=(2) $\mathbf{MAXu}(u\div x_1\wedge u\div x_2)=\mathbf{MAXu}(u\div x_1\wedge u\div x_2)$

分分 120(1)(2)=(3) $(x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0))$
$\supset(x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0)$
$\wedge\mathbf{MAXu}(u\div x_1\wedge u\div x_2)$
$=\mathbf{MAXu}(u\div x_1\wedge u\div x_2))$

赋值(3)=(4) $\{x_1\geqslant 0\wedge x_2\geqslant 0\wedge(x_1\neq 0\vee x_2\neq 0)\}(z_1, z_2):$
$=(x_1, x_2);$
$\{z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge$
$\mathbf{MAXu}(u\div z_1\wedge u\div z_2)=\mathbf{MAXu}(u\div x_1\wedge u\div x_2)\}$

整数性质 A_1=(5) $(z_2\geqslant z_1)\supset(z_2-z_1\geqslant 0)$

公理 20=(6) $(z_1\neq 0)\supset(z_1\neq 0\vee z_2-z_1\neq 0)$

整数性质 A_2=(7) $\mathbf{MAXu}(u\div z_1\wedge u\div(z_2-z_1))$
$=\mathbf{MAXu}(u\div z_1\wedge u\div z_2)$

分 602(7)=(8) $(\mathbf{MAXu}(u\div z_1\wedge u\div z_2)$
$=\mathbf{MAXu}(u\div x_1\wedge u\div x_2))$
$\supset(\mathbf{MAXu}(u\div z_1\wedge u\div(z_2-z_1))$
$=\mathbf{MAXu}(u\div x_1\wedge u\div x_2))$

由(5)(6)(8)可得(9) $(z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge$
$\mathbf{MAXu}(u\div z_1\wedge u\div z_2)$
$=\mathbf{MAXu}(u\div x_1\wedge u\div x_2)\wedge$
$z_1\neq 0\wedge z_2\geqslant z_1)$
$\supset(z_1\geqslant 0\wedge(z_2-z_1)$

$$\geqslant 0 \wedge (z_1 \neq 0 \vee (z_2 - z_1) \neq 0) \wedge$$
$$\mathbf{MAXu}(u \div z_1 \wedge u \div (z_2 - z_1))$$
$$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2))$$

赋值(9)=(10) $\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$
$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2)$
$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2) \wedge$
$z_1 \neq 0 \wedge z_2 \geqslant z_1\} z_2 := z_2 - z_1;$
$\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$
$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2)$
$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

易证(11) $\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$
$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2)$
$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2) \wedge$
$z_1 \neq 0 \wedge z_2 < z_1\} (z_1, z_2) := (z_2, z_1);$
$\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$
$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2)$
$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

条件(10)(11)=(12) $\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$
$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2)$
$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)$
$\wedge z_1 \neq 0\}$ **if** $z_2 \geqslant z_1$ **then** $z_2 :$
$= z_2 - z_1$ **else** $(z_1, z_2) := (z_2, z_1)$
$\{z_1 \geqslant 0 \wedge z_2 \geqslant 0 \wedge (z_1 \neq 0 \vee z_2 \neq 0) \wedge$
$\mathbf{MAXu}(u \div z_1 \wedge u \div z_2)$
$= \mathbf{MAXu}(u \div x_1 \wedge u \div x_2)\}$

整数性质 A_3=(13) $z_2 = \mathbf{MAXu}(u \div z_2)$

整数性质 A_4=(14) $\mathbf{MAXu}(u \div z_2)$

$$=MAXu(u\div 0\wedge u\div z_2)$$

**假设(15)　$z_1=0$

等替(15)=(16)　$MAXu(u\div 0\wedge u\div z_2)$

$$=MAXu(u\div z_1\wedge u\div z_2)$$

**假设(17)　$MAXu(u\div z_1\wedge u\div z_2)=MAXu\ (u\div x_1\wedge u\div x_2)$

(分 602)^3(13)(14)(16)(17)=(18)

$$z_2=MAXu(u\div x_1\wedge u\div x_2)$$

消去(15)(16)得(19)　$(z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge$

$MAXu(u\div z_1\wedge u\div z_2)$

$=MAXu(u\div x_1\wedge u\div x_2)\wedge z_1=0)$

$\supset(z_2=MAXu(u\div x_1\wedge u\div x_2))$

循环(12)(19)=(20)　$\{z_1\geqslant 0\wedge z_2\geqslant 0\wedge(z_1\neq 0\vee z_2\neq 0)\wedge$

$MAXu(u\div z_1\wedge u\div z_2)$

$=MAXu(u\div x_1\wedge u\div x_2)\}$

while $z_1\neq 0$ **do**

if $z_2\geqslant z_1$ **then** $z_2:=z_2-z_1$

else$(z_1, z_2):=(z_2, z_1)$

od;

$\{z_2=MAXu(u\div x_1\wedge u\div x_2)\}$

易证(21)　$\{z_2=MAXu(u\div x_1\wedge u\div x_2)\}\ y:=z_2;$

$\{y=MAXu(u\div x_1\wedge u\div x_2)\}$

(结合)^2(4)(20)(21)=(甲)

证毕.

上面我们对部分正确性的处理虽然只考虑了一个很简单的程序设计语言和一个很简单的程序，但由此可见一斑. 当然，实际上在验证各类程序时会遇到很多复杂的问题；例如程序设计语言可能会有更复杂的特点，计算机上的运算会出现误差，计算过程中可

能产生上溢下溢. 这里我们就略而不论, 留待读者进一步探讨.

习　题

证明下列程序是部分正确的(程序中说明部分略去未写)

1. **program** p1;
   read a, b;
   q:=0; r:=a;
   while r≥b **do**
   q:=q+1; r:=r−b
   od;
   write q, r
   end p1

2. **program** p2;
   read x;
   s:=0;
   for i:=1 **step** 1 **until** n **do**
   s:=s+x[i]
   od;
   write s
   end p2

注意, 应首先给出关于 **for** 语句的推理规则.

第三章　递归函数

§3.0　数学归纳法

在递归函数论中讨论的基本对象是以自然数集为定义域和值域的数论函数，因此与自然数集有密切关系的一种论证方法——数学归纳法将起着重要的作用．为此本章首先介绍一下数学归纳法．

设$P(n)$是一个含有自然数变元n的待证命题．我们可用如下的*数学归纳法*来证明它对一切自然数n均成立：

第一步证明当$n=0$时该命题成立，即证明$P(0)$成立．这一步称为*奠基*．

第二步证明对于任何自然数k，如果$n=k$时该命题成立，那么$n=k+1$时该命题也成立，即证明$P(k)\supset P(k+1)$成立．这一步称为*归纳*．

由此可知，用数学归纳法来证明一个命题成立时，对该命题有两点要求．第一，要求该命题中含有自然数变元n．第二，要求该命题对一切自然数n均成立．

另外，使用数学归纳法时还需注意在归纳步骤中不是无条件地证明“$n=k+1$时待证命题成立”，而是在“$n=k$时待证命题成立”的假设前提之下证明“$n=k+1$时待证命题成立”．这个假设前提称为*归纳假设*．

容易看出，如果奠基和归纳这两步均得证的话，则必有待证命题对任何自然数n均成立．这是因为，由第一步得知$n=0$时待证命题$P(n)$成立，即$P(0)$真．既然$P(0)$真，由第二步便得知$P(1)$真．

既然 $P(1)$真，由第二步又得知$P(2)$真．这样一直推下去，便得知对一切自然数n，待证命题 $P(n)$均真．因此当这两步得证后，我们便说，根据数学归纳法，该命题 $P(n)$对于一切自然数n均成立．变元n称为归纳变元．

现举例说明数学归纳法的使用方法．

例1：求证对于一切n，$2^n \geqslant n+1$

[证]　奠基　　$n=0$时

左端$=2^0=1 \geqslant 0+1=$右端

归纳　　$n=k+1$时

左端$=2^{k+1}=2\cdot 2^k \geqslant 2\cdot(k+1)$　　[归纳假设]

$=(k+1)+(k+1) \geqslant (k+1)+1=$右端

故依数学归纳法，原式得证．

由这个式子容易推得下面的结论．

推论1　$2^{n-1} \geqslant n$.

推论2　$2^n \geqslant 2n$.

归纳假设有各种形式，常用的有三种．

第一种就是上面介绍的形式，即假设$n=k$时待证命题成立．这种形式的归纳假设称为简单归纳假设．使用简单归纳假设的归纳法称为简单归纳法．

第二种形式是假设$n \leqslant k$时待证命题成立．这种形式的归纳假设称为强归纳假设．相应的证明方法称为强归纳法，或叫做串值归纳法．

第三种归纳假设是对含有参数的命题而言的，设有待证命题 $P(m, n)$，其中m为参数，n为归纳变元．如欲证明对于一切m，n，$P(m, n)$成立，则奠基中应证明 $P(m, 0)$对一切m均成立，归纳步骤中应证明对于一切k，如果$P(m, k)$对一切m均成立，那么$P(m, k+1)$也对一切m均成立．这里的归纳假设“$P(m, k)$对一切m成立”

叫做参变归纳假设. 相应的证明方法叫做参变归纳法.

下面举例说明强归纳法和参变归纳法的使用方法.

设P_k表示第k个质数. 因此有

$$P_0=2,\quad P_1=3,\quad P_2=5,$$
$$P_3=7,\quad P_4=11,\quad \cdots$$

例2: 求证 $P_n\leqslant 2^{2^n}$

[**证**] 奠基　当$n=0$时

$$\text{左端}=P_0=2\leqslant 2^{2^0}=\text{右端}$$

归纳　因为$P_0\cdot P_1\cdot\cdots\cdot P_k$能被$P_i(i=0,1,\cdots,k)$整除, 所以$(P_0\cdot P_1\cdot\cdots\cdot P_k+1)\div P_i$的余数为1$(i=0,1,\cdots,k)$. 即$P_0, P_1,\cdots, P_k$均不能整除$(P_0\cdot P_1\cdot\cdots\cdot P_k+1)$. 因而$P_0, P_1,\cdots, P_k$均不是$(P_0\cdot P_1\cdot\cdots\cdot P_k+1)$的质因子. 故必有$(P_0\cdot P_1\cdot\cdots\cdot P_k+1)$的质因子$\geqslant P_{k+1}$. 因此

$$\begin{aligned}P_{k+1}&\leqslant P_0\cdot P_1\cdot\cdots\cdot P_k+1\\&\leqslant 2^{2^0}\cdot 2^{2^1}\cdot\cdots\cdot 2^{2^k}+1 &&\text{[强归纳假设]}\\&=2^{(2^0+2^1+\cdots+2^k)}+1\\&=2^{(2^{k+1}-1)}+1\\&=\frac{2^{2^{k+1}}+2}{2}\\&\leqslant 2^{2^{k+1}}\end{aligned}$$

故依数学归纳法原式得证.

必须注意, 在使用归纳假设时须防止无形中引入不相干的假设, 否则会引起错误. 例如"任意$n+1$条直线均重合成一条直线"这个命题当然是错的. 但是我们可以用下面的错误证明来证得这个结论.

奠基　当$n=0$时, 该命题显然成立.

归纳　利用强归纳法可以有如下的强归纳假设: 任意1条, 2

条，3条，…，$k+1$ 条直线均重合成一条直线．求证：任意 $k+2$ 条直线重合成一条直线．

设 $k+2$ 条直线为 $l_1, l_2, \cdots, l_k, l_{k+1}, l_{k+2}$ 由强归纳假设得 $l_1, l_2, \cdots, l_{k+1}$ 这 $k+1$ 条直线重合为一条直线，记为 l．又由强归纳假设得 l 和 l_{k+2} 这两条直线重合为一条直线．于是任意 $k+2$ 条直线便重合为一条直线了．

依强归纳法该命题得证．

不细心分析的读者也许不易发现这个证明中的错误．实际上这里的错误是由于错误地使用了强归纳假设造成的．具体地说就是在"l 和 l_{k+2} 这两条直线重合为一条直线"这一点把强归纳假设使用错了．强归纳假设并没有包含这一条件．这是因为在采用强归纳假设的归纳步骤中要证明的结论是：

如果任意1条，2条，3条，…，$k+1$ 条直线重合成一条直线，那么，任意 $k+2$ 条直线也重合成一条直线．特别应该注意的是这个结论要求对于一切大于等于 0 的 k 均成立，而不是对于一切大于等于1的 k 成立，这是因为我们奠基时，奠的基是 $n=0$ 而不是 $n=1$．上面的证明中所假设的 l 和 l_{k+2} 重合为一条直线实际上是要求 $k \geqslant 1$．这就是上面证明的错误所在．

例3：设 $f(n,m)=\begin{cases}0, & \text{当 } n=0 \text{ 时} \\ f(n-1,m^2)\cdot f(n-1,2mn), & \text{当 } n\neq 0 \text{ 时}\end{cases}$

求证 $f(n,m)=0$．

［证］ 把 m 看作参数，对 n 施行归纳．

奠基 当 $n=0$ 时，　左端 $=f(0,m)=0=$ 右端

归纳 当 $n=k+1$ 时

$$\begin{aligned}\text{左端} &= f(k+1,m)=f(k,m^2)\cdot f(k,2m(k+1)) \\ &= 0\cdot 0 \qquad \text{［参变归纳假设］} \\ &= 0=\text{右端}\end{aligned}$$

故依数学归纳法本题得证.

上面我们已指出，使用数学归纳法时，对待证命题有两点要求．其中第二点要求是：待证命题对一切自然数n均成立．但是我们有时碰到要求证明命题$P(n)$对于一切大于等于某个常数c的n成立．对于这种命题能否用数学归纳法来证明呢？显然是可以的．这是因为有下面的等价关系：

"对于一切$n\geqslant c$的n，$P(n)$成立"

等价于

"对于一切$n\geqslant 0$的n，$P(n+c)$成立"

由此可知，归纳法可作如下的修改：

第一步证明当$n=c$时$P(n)$成立，

第二步证明对于任何$k\geqslant c$的k，如果$n=k$时$P(n)$成立，则$n=k+1$时$P(n)$也成立．

应用归纳法证明命题时往往需要施展一定的技巧．下面再举一例来说明某种技巧的使用．

例4：求证　$n\geqslant 3$时$n^{(n+1)}\geqslant(n+1)^n$

[**证**]　容易验证，当$n=3$时，该不等式是成立的．但是无论采用哪种归纳假设，归纳步骤都难以得证．然而如果按下面的方式把该式中的部分n改写成u，部分n保持不变：

(1)　　$nu^n\geqslant(u+1)^n$

则可用归纳法证明当$u\geqslant n\geqslant 3$时，(1)成立．其证明如下(把u看作参数)：

奠基　当$n=3$时，$u\geqslant 3$

(1)的左端$=3u^3=u^3+u\cdot u^2+u^2\cdot u$

$\geqslant u^3+3u^2+9u$

$>u^3+3u^2+3u+1=(u+1)^3=$(1)的右端

归纳　当$n=k+1$时

(1)的左端$=(k+1)u^{(k+1)}=u(k+1)u^k$

$=(uk+u)u^k\geqslant(uk+k)u^k$

$=k(u+1)u^k$

$\geqslant(u+1)(u+1)^k$ [归纳假设]

$=(u+1)^{k+1}=$(1)的右端

故依数学归纳法(1)得证.

令 $u=n$, (1)便化为$n^{n+1}\geqslant(n+1)^n$

即为原不等式. 故原式得证.

例 4 的证明方法称为拆裂法. 一个待证命题可以拆裂成各种形式, 例如例 4 的不等式除可拆裂成(1)外, 还可拆裂成下面几种形式:

(2) $u^{n+1}\geqslant(u+1)^n$

(3) $n^{u+1}\geqslant(u+1)^n$

(4) $u^{n+1}\geqslant(n+1)^u$

当$u\geqslant n\geqslant3$时, (2)是成立的, 但难以使用归纳法证明; (3)也是成立的, 可以用归纳法证明, 但比较麻烦; (4)根本不成立.

由此可知, 拆裂法是一种灵活多变的方法, 必须善于处理才行.

上面已指出使用数学归纳法时要求待证命题含有自然数变元 n, 并对n施行归纳证明. 应该注意的是有些命题表面上看不含有自然数变元, 但仔细一分析就看出实际上含有自然数变元. 例如, 在§1.2中曾证明下面的定理:

"对于任何公式α, 均有$\bar{\alpha}=\alpha^{*-}$"

表面上看, 该定理中不含有自然数变元, 但是因为公式是由命题变元利用真值联结词构成的, 公式中所含的命题联结词的个数n是自然数, 所以该定理隐含地含有自然数变元, 我们可以据此施行数学归纳法来证明该定理. 在§1.2中, 我们的确是用数学归纳法来证

它的.

其实数学归纳法不仅可用于含有自然数变元n的命题,而且可用于含有某些其它集合上的变元的命题. 我们称它为归纳集.

设有一个集合A. 如果它满足下面三个性质:

1. $a_1, a_2, \cdots, a_n$是A中元素$(n\geqslant 1)$;

2. 如果x是A中元素, 则$f_{11}(x), f_{12}(x), \cdots, f_{1n_1}(x)$也是$A$中元素$(n_1\geqslant 0)$;

如果x, y是A中元素, 则$f_{21}(x, y), f_{22}(x, y), \cdots, f_{2n_2}(x, y)$也是$A$中元素$(n_2\geqslant 0)$;$\cdots$

如果$x_1, \cdots, x_m$是A中元素, 则$f_{m1}(x_1, \cdots, x_m), f_{m2}(x_1, \cdots, x_m), \cdots, f_{mn_m}(x_1, \cdots, x_m)$也是$A$中元素$(m\geqslant 1, n_m\geqslant 0)$.

其中诸f_{ij}均是A上的函数.

3. A中元素也仅限于此.

我们把上面这样的集合A称为*归纳集*, $a_1, a_2, \cdots, a_n$称为该集的*开始元素*, 诸f_{ij}称为该集的*生成函数*.

显然自然数集是归纳集, 其开始元素为0, 其生成函数为$f(x)=x+1$.

由a, b, c, d四个元素利用$+$, $-$运算所构成的一切算术表达式组成的集合是归纳集, 它的开始元素是a, b, c, d, 它的生成函数是$f_1(x, y)=x+y, f_2(x, y)=x-y$.

命题演算公式集和谓词演算公式集也都是归纳集.

对于一个含有某个归纳集A上的变元x的待证命题$P(x)$, 可用如下的*广义数学归纳法*来证明:

奠基 证明对于A中的所有开始元素待证命题均成立, 即证明$P(a_1), P(a_2), \cdots, P(a_n)$成立.

归纳 证明对于$1\leqslant i\leqslant m$和$1\leqslant j\leqslant n_i$的所有$i, j$, 对于$A$中任何元素$x_1, x_2, \cdots, x_i$, 如果$P(x_1), P(x_2), \cdots, P(x_i)$成立, 则$P(f_{ij}$

$(x_1, x_2, \cdots, x_i)$)也成立.

下面我们举例说明广义归纳法的使用.

例 5: 求证在由a, b, c, d四个标识符利用+，−两个运算符构成的一切算术表达式所组成的集合中的任何一个表达式里标识符的个数等于该表达式里运算符的个数加1.

[**证**] 因为该集合是归纳集，所以我们可用广义归纳法来证明该命题.

奠基 对于该集合的四个开始元素 a, b, c, d，因为它们的标识符个数均为1，运算符个数均为0，所以该命题显然成立.

归纳 对于由该集合中的元素 x 和y用+构成的新元素$x+y$，我们设

$x+y$ 中标识符个数为m，运算符个数为 n;

x 中标识符个数为 m_1，运算符个数为n_1;

y 中标识符个数为 m_2，运算符个数为 n_2.

容易得到

$$m = m_1 + m_2 = (n_1 + 1) + (n_2 + 1) \qquad \text{[归纳假设]}$$
$$= (n_1 + n_2 + 1) + 1 = n + 1$$

故该命题成立.

同理，对于$x-y$，该命题也成立.

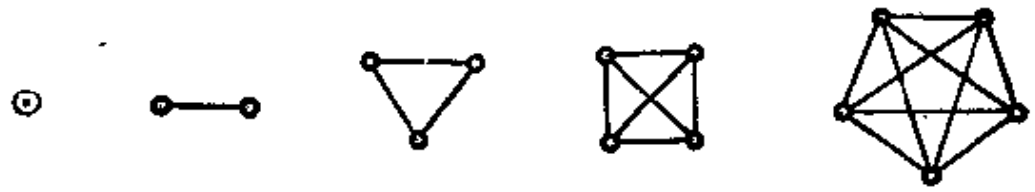
故依广义数学归纳法本命题成立.

习　题

1. 证明对于任何自然数 n, $(3n+1)7^n - 1$ 均为 9 的倍数.
2. 证明对于任何自然数 n, $2^n \geqslant 2n$.
3. 证明对于任何自然数 n, $2^{n+1} > n(n+1)$.
4. 证明对于任何自然数 n, $\sum_{i=0}^{n} i^3 = \left(\sum_{i=1}^{n} i\right)^2$.

5. 证明对于任何自然数 n, $\sum_{i=1}^{n} 2^i = 2^{n+1} - 1$.

6



上图是含有 1, 2, 3, 4 和 5 个节点的完全图的例子. 类似这些完全图, n 个节点的完全图由 n 个节点以及联结任何两个节点的边构成. 试给出 n 个节点的完全图中边的总数的公式, 并证明之.

7. 证明对于任何自然数 n, 如果 $f(n+1) > f(n)$ 且 $f(0) > 0$, 则 $f(n) > n$, 其中 f 的值域为自然数域.

8. 证明对于任何自然数 n, 当 $n \neq 3$ 时 $2^n \geqslant n^2$.

9. 证明对于任何正整数 n 以及任何正数 $a_1, a_2, \cdots, a_n$, $2^{a_1+\cdots+a_n+n} \geqslant 2^{a_1+1} + \cdots + 2^{a_n+1}$.

10. 证明对于任何自然数 n, $n! \leqslant n^n \leqslant 2^{n^2}$. 这里约定 $0! = 1$, $0^0 = 1$.

11. 证明对于任何大于 1 的正整数 n 以及任何大于 -1 且不等于 0 的实数 x, $(1+x)^n > 1+nx$.

由此可以证明对于任何实数 u 以及任何自然数 n, 当 $u \geqslant n$ 时, $\left(1+\frac{1}{u}\right)^u < \left(1+\frac{1}{u+1}\right)^{u+1}$; 当 $1 \leqslant u \leqslant n$ 时, $\left(1+\frac{1}{u}\right)^{u+1} > \left(1+\frac{1}{u+1}\right)^{u+2}$.

12. 斐波纳契(Fibonacci)数列的定义如下:

$$\begin{cases} F(0) = 0 \\ F(1) = 1 \\ F(n+2) = F(n+1) + F(n) \end{cases} \quad (n \text{ 为任意自然数})$$

证明对于任何自然数 n, 成立

i) $\sum_{i=0}^{n} F(i) = F(n+2) - 1$;

ii) $\sum_{i=0}^{n} F(2i) = F(2n+1) - 1$;

iii) $\sum_{i=0}^{n} F^2(i) = F(n) \cdot F(n+1)$;

设 $a=(1+\sqrt{5})/2$

iv) $F(n) \leqslant a^{(n-1)}$;

v) 当 $n \geqslant 1$ 时, $a^{(n-2)} \leqslant F(n)$.

13. 修改强归纳法的形式, 使之适合于证明对于$[n_0, m_0]$范围内的一切n, $P(n)$成立.

14. 指出下面证明过程中的错误.

错误的待证命题是:

对于任何大于等于 1 的自然数 n

$$\frac{1}{1\cdot 2}+\frac{1}{2\cdot 3}+\frac{1}{3\cdot 4}+\cdots+\frac{1}{(n-1)(n)}=\frac{3n-2}{2n}$$

错误的归纳证明是:

奠基 $n=1$ 时, 左端$=\frac{1}{1\cdot 2}=\frac{3-2}{2}=$右端

归纳 $n=k+1$ 时,

$$\text{左端}=\left(\frac{1}{1\cdot 2}+\frac{1}{2\cdot 3}+\cdots+\frac{1}{(k-1)\cdot k}\right)+\frac{1}{k(k+1)}$$

$$=\frac{3k-2}{2k}+\frac{1}{k(k+1)} \qquad \text{[归纳假设]}$$

$$=\frac{(3k-2)(k+1)+2}{2k(k+1)}=\frac{3(k+1)-2}{2(k+1)}=\text{右端}$$

故由数学归纳法本命题成立.

15. 指出下面证明过程中的错误.

错误的待证命题是:

对于任何自然数 n 以及任何非 0 实数 a, $a^n=1$;

错误的证明过程是:

奠基 $n=0$ 时本命题显然成立.

归纳 $n=k+1$ 时, 左端$=a^{k+1}=a^k\cdot a=a^k\cdot\frac{a^k}{a^{k-1}}$

$$=1\cdot\frac{1}{1} \qquad \text{[强归纳假设]}$$

$$=1=\text{右端}$$

故依数学归纳法本命题成立.

§3.1 数论函数与数论谓词

上面提到，在递归函数论中，自始至终都是以自然数(即正整数及零)为其研究对象. 为什么要作这样严格的限制？这是因为在递归函数论中所使用的主要方法(即幂状式和递归式)是从自然数集的研究中产生的，而且从本质上说也只能应用到自然数集上去.

因此，本章中所谓"数"专指自然数，所谓"变元"专指以自然数集为变域的变元，所谓"函数"专指以自然数集为定义域及值域的函数. 这种函数称为*数论函数*.

下列函数都是简单的常用的数论函数(其中x, y均为自然数集中的变元).

$x+y$　指x与y的和.

$x\cdot y$　指x与y的积.

$x\dot{-}y$　指x与y的算术差，即当$x\geqslant y$时其值为x减y之差，当$x<y$时其值永为0. 显然这与代数中减法的意义不相同，代数中引入了负数的概念，而我们这里是不讨论负数的.

$[x/y]$　指y除x的算术商，并约定$[x/0]=0$. 即当$y\neq 0$时其值为y除x的代数商的整数部分，当$y=0$时其值为0. 例如，$[4/7]=[0.57\cdots]=0$, $[10/3]=[3.3\cdots]=3$, $[11/3]=[3.6\cdots]=3$, $[15/5]=[3]=3$. 显然这与代数中除法的意义不相同. 代数中引入了小数的概念，我们这里却始终限于自然数.

$[\sqrt{x}]$　指x的平方根的整数部分. 例如，$[\sqrt{2}]=[1.414\cdots]=1$, $[\sqrt{9}]=[3]=3$. $[\sqrt{15}]=[3.8\cdots]=3$.

$x\dot{-}\dot{-}y$　指x与y的绝对差，即大数减小数之差.

$rs(x, y)$　指y除x的剩余. 例如，$rs(4, 7)=4$, $rs(15, 5)=0$, 并约定$y=0$时，$rs(x, y)=x$.

x^y 指 x 的 y 次方幂，并约定 $x^0=1$.

Ex 指 $x\doteq[\sqrt{x}]^2$，叫做平方剩余.

$dv(x,y)$ 指 x 与 y 的最大公约数，并约定 $x\cdot y=0$ 时其值为 $x+y$.

$lm(x,y)$ 指 x 与 y 的最小公倍数，并约定 $x\cdot y=0$ 时，其值为 0.

P_x 指第 x 个(从第 0 个起算)质数. 例如，$P_0=2$, $P_1=3$, $P_2=5$, $P_3=7$, $P_4=11$.

ep_ax 指 x 的质因子分解式中第 a 个质数的幂指数. 例如，$ep_0 12=ep_0(2^2\cdot 3)=2$, $ep_1 12=1$, $ep_2 12=0$, $ep_3 12=0$, 当 $i\geqslant 2$ 时，$ep_i 12=0$.

上面这些函数都是利用算术或代数中的“和”，“差”，“积”，“商”等知识来定义的，下面我们再给出若干个更简单的可以直接定义的函数. 所谓可以直接定义的函数是指可以用直接给出运算法则的方法来定义的函数.

$I_{(x)}=x$，即函数的值与自变元的值永相同. 这个函数叫做么函数.

$I_{mn}(x_1,x_2,\cdots,x_m)=x_n$，即函数的值与第 n 个自变元的值永相同. 这个函数叫做广义么函数.

$O(x)=0$，即函数之值永为0，这个函数叫做零函数.

$C_a(x)=a$，即函数之值永为 a，这个函数叫做常值函数.

$$xNy=\begin{cases}x, & \text{当}y=0\text{时}\\ 0, & \text{当}y\neq 0\text{时}\end{cases}$$

$$xN^2y=\begin{cases}0, & \text{当}y=0\text{时}\\ x, & \text{当}y\neq 0\text{时}\end{cases}$$

特例，当 $x=1$ 时有

$$Ny=\begin{cases}1, & \text{当}y=0\text{时}\\ 0, & \text{当}y\neq 0\text{时}\end{cases}\qquad N^2y=\begin{cases}0, & \text{当}y=0\text{时}\\ 1, & \text{当}y\neq 0\text{时}\end{cases}$$

$$eq(x,y)=\begin{cases}0, & \text{当}x=y\text{时}\\ 1, & \text{当}x\neq y\text{时}\end{cases}$$

如果已知自然数的大小关系，还可直接定义下列函数：

$$max(x,y)=\begin{cases}x, & \text{当}x\geqslant y\text{时}\\ y, & \text{当}x<y\text{时}\end{cases}$$

$$min(x,y)=\begin{cases}x, & \text{当}x\leqslant y\text{时}\\ y, & \text{当}x>y\text{时}\end{cases}$$

如果已知自然数的先后次序，还可直接定义下列函数：

$Sx=$紧接在 x 之后的数，通常表为$Sx=x+1$，这个函数叫做*后继函数*.

$Dx=$紧列于 x 之前的数，并约定$D0=0$，通常表为

$$Dx=\begin{cases}x-1, & \text{当}x\neq 0\text{时}\\ 0, & \text{当}x=0\text{时}\end{cases}$$

这个函数叫做*前驱函数*.

上面这些函数虽很简单，却是我们今后讨论的基础，必须熟悉之. 尤其是广义么函数、零函数和后继函数更为重要，为此特把这三者称为*本原函数*.

在递归函数论中，经常要论及到谓词以及含变元的语句.

我们已经知道，谓词是以个体或命题为定义域，以真、假为值的函数，而含变元的语句是指含有个体变元的谓词填式或者由它们利用真值联结词和量词组成的式子，也就是谓词演算公式，因为递归函数论中只讨论自然数，所以个体只能是自然数，凡是以自然数集为定义域，以真、假值为值域的谓词均称为*数论谓词*；任何数论谓词填式或者由它们利用真值联结词和量词组成的任一公式均

称为数论语句. 例如,“大于”,“为质数”,“为平方数”都是数论谓词,“9>7且9为平方数”,“$(x\geqslant 3y+2)\supset(y<x)$”,都是数论语句(一般讲语句不含有变元,这里我们广义地来理解).

在递归函数论中所讨论的谓词仅限于数论谓词,所讨论的语句仅限于数论语句,因此,在本章中所谓“谓词”专指数论谓词,所谓“含变元的语句”专指含自然数变元的数论语句.

定义: 设$A(x_1,\cdots,x_n)$是一个含有n个变元的语句,$f(x_1,\cdots,x_n)$是一个n元数论函数,且满足下列关系,对于任何变元组$(x_1,\cdots,x_n)$均有

$$\begin{aligned} &A(x_1,\cdots,x_n)\text{真时},\ f(x_1,\cdots,x_n)=0\\ &A(x_1,\cdots,x_n)\text{假时},\ f(x_1,\cdots,x_n)=1 \end{aligned}\tag{1}$$

则说$f(x_1,\cdots,x_n)$是语句$A(x_1,\cdots,x_n)$的特征函数,记为

$$ctA(x_1,\cdots,x_n)$$

由此定义易得下面定理.

定理1 任何一个语句A均有一个也只有一个相应的特征函数.

[**证**] 先证存在性,因为对于任何一个语句A,我们恒可以直接用(1)来定义函数f. 这样定义出来的函数当然是A的特征函数,故存在性得证. 再证唯一性,设f和g均为A的特征函数,由定义知,当$A(x_1,\cdots,x_n)$真时,$f(x_1,\cdots,x_n)=g(x_1,\cdots,x_n)=0$,当$A(x_1,\cdots,x_n)$假时,$f(x_1,\cdots,x_n)=g(x_1,\cdots,x_n)=1$,故$f(x_1,\cdots,x_n)=g(x_1,\cdots,x_n)$,即$A$只有一个特征函数. 唯一性得证.

定理2 如果有一函数$f(x_1,\cdots,x_n)$满足下列条件:

$$A(x_1,\cdots,x_n)\text{真当且仅当}\ f(x_1,\cdots,x_n)=0\tag{2}$$

则$N^2f(x_1,\cdots,x_n)$为语句$A(x_1,\cdots,x_n)$的特征函数.

请自行证明.

满足条件(2)的函数$f(x_1,\cdots,x_n)$称为$A(x_1,\cdots,x_n)$的准特征

函数.

下面是一些含变元的语句及其相应的特征函数:

语　句	特征函数
x为0	N^2x
x异于0	Nx
x为偶数	$rs(x, 2)$
x为3的倍数	$N^2rs(x, 3)$
x为平方数	$N^2(x\dot{-}[\sqrt{x}]^2)$, 即 N^2Ex
x等于y	$N^2(x\dot{-}y)$
x小于y	$N^2((x+1)\dot{-}y)$或 $N^2[(x+1)/(y+1)]$
x小于等于y	$N^2(x\dot{-}y)$或$N^2[x/(y+1)]$
x为y的倍数	$N^2rs(x, y)$

现在我们讨论复合语句的特征函数.

由简单语句构造复合语句的方法有两种: 一是利用命题联结词; 一是利用量词. 试问如果已知成份语句的特征函数, 能否求出复合语句的特征函数? 回答是肯定的. 首先看看利用真值联结词构造新语句时, 新旧语句的特征函数间的关系.

定理3 设A, B为任意两个语句, 则有

$$ct\bar{A}=1\dot{-}ctA=NctA$$

$$ct(A\vee B)=ctA\cdot ctB=min(ctA, ctB)$$

$$ct(A\wedge B)=N^2(ctA+ctB)=max(ctA, ctB)$$

$$ct(A\supset B)=ctB\cdot NctA=ctBNctA$$

$$ct(A\equiv B)=ctA\dot{-}ctB$$

由此定理可知, 对于利用命题联结词而作成的新语句, 其特征函数可以由其成份语句(即上面的A, B)的特征函数以及若干个常函数(加, 乘, 绝对减, N等)复合而得.

我们再来看当用量词构造新语句时，新旧语句之间的关系.

设$A(x)$为任意一个含变元x的语句. $\forall xAx$和$\exists xAx$是由Ax利用量词造出的新语句，分别表示“所有自然数x均使得Ax成立”，“有一个自然数使得Ax成立”，$\forall xAx$和$\exists xAx$的特征函数可以由Ax的特征函数表示出来，但是所用的方法不属递归函数论讨论的范围，所以不予讨论. 但是下面两个受限量词却是值得讨论的.

$\underset{x\to n}{\forall} Ax$表示：对于0到$n$间的一切$x$均使得$Ax$成立.

$\underset{x\to n}{\exists} Ax$表示：在0到$n$间至少有一个$x$使得$Ax$成立.

我们称$\underset{x\to n}{\forall} Ax$是由$Ax$利用受限全称量词而得到的新语句，$\underset{x\to n}{\exists} Ax$是由$Ax$利用受限存在量词而得到的新语句. 显然这二个新语句都是含有变元n的语句. 它们的特征函数与Ax的特征函数有下列关系.

定理4 设Ax为任意一个含有x的语句，则有

$$\begin{aligned} ct(\underset{x\to n}{\forall} Ax) &= max(ctA(0), ctA(1), \cdots, ctA(n)) \\ &= N^2(ctA(0)+ctA(1)+\cdots+ctA(n)) \end{aligned} \tag{3}$$

$$\begin{aligned} ct(\underset{x\to n}{\exists} Ax) &= min(ctA(0), ctA(1), \cdots, ctA(n)) \\ &= ctA(0)\cdot ctA(1)\cdot \cdots \cdot ctA(n) \end{aligned} \tag{4}$$

其中：

$max(x_0, \cdots, x_n)$表示诸x_i中最大者，

$min(x_0, \cdots, x_n)$表示诸x_i中最小者.

[证] 现对(3)作证明，(4)的证明仿此.

当$ct(\underset{x\to n}{\forall} Ax)=0$时由特征函数的定义知，这时$\underset{x\to n}{\forall} Ax$为真即$A(0)$，$A(1)$，$\cdots$，$A(n)$均真，故$ctA(0)$，$ctA(1)$，$\cdots$，$ctA(n)$均为0，故$max(ctA(0), ctA(1), \cdots, ctA(n))=0$，而且有$N^2(ctA(0)+ctA(1)+\cdots+ctA(n)=N^20=0$，所以当$ct(\underset{x\to n}{\forall} Ax)=0$，(3)成立.

当$ct(\underset{x\to n}{\forall} Ax)=1$时，由定义知这时$\underset{x\to n}{\forall} Ax$为假，即至少有一个$x$($\leqslant n$)使得$Ax$假，设$A(i)$为假，则$ctA(i)=1$，故$max(ctA(0), ctA$

$(1), \cdots, ctA(n))=1$, 又因为 $ctA(0)+\cdots+ctA(i)+\cdots+ctA(n)$ $\geqslant ctA(i) \fallingdotseq 0$, 故有, $N^2(ctA(0)+\cdots+ctA(n))=1$, 所以 $\underset{x\to n}{\forall} Ax$ 为假时(3)也成立.

由此可知(3)成立.

这样, 对于利用受限量词而作成的新语句, 其特征函数可以由其成分语句的特征函数以及 $max(x_0, \cdots, x_n)$ 和 $min(x_0, \cdots, x_n)$ 而得.

最后, 特别应该指出, 为了节省括号起见, 在本章中, 除使用通常的先乘方后乘除, 先乘除后加减等约定外, 还使用左结合的约定. 例如, $a \dot{-} b+b \dot{-} a$ 指 $((a \dot{-} b)+b) \dot{-} a$.

习　题

1. 已知如何把一数表示成二进制数, 试直接定义下列函数:

1.1 Sx;

1.2 Dx;

1.3 $max(x, y)$, (x 和 y 中的大者);

1.4 $min(x, y)$, (x 和 y 中的小者);

1.5 $x+y$.

2. 直接定义下列函数:

2.1 $1 \dot{-} x$;

2.2 $rs(2, x)$;

2.3 $[2/x]$.

3. 给出下列语句的特征函数:

3.1 只要 $x>0$, 便有 $x^3>0$;

3.2 $3^n \geqslant n^3$ 当且仅当 $n \geqslant 3$;

3.3 a 为 b, c 的公倍数;

3.4 x 非负数;

3.5 a 为 b, c 的公约数;

3.6 a, b 互质；

3.7 y 为 x 以下(包括 x)的最大平方数；

3.8 在 a, b 间一切 x 均使 $A(x)$ 成立；

3.9 在 a, b 间至少有一 x 使 $A(x)$ 成立；

3.10 如果存在这样的 c，使得 $c>0$，$c\leqslant n$ 且 $a\leqslant b+c$，那么 $a\leqslant b$；

3.11 如果 $A(x, y)$ 真，则 $B(y, z)$ 真，除非 $D(x, y, z)$ 成立；

3.12 当 $A(x)$ 成立时，$B(y, z)$ 成立的充要条件是 $D(y, z)$ 成立.

§3.2 迭置与算子

上一节我们直接定义了么函数、广义么函数等十个很简单的函数. 显然这种直接定义函数的方法只能对极简单的函数使用，而不能大规模地使用. 要源源不断地造出新函数只有利用旧函数来构造，由旧函数造新函数的方法称为派生法，派生法可分两大类，迭置法和算子法.

设新函数在某一变元组处的值与诸旧函数的n个值有关，如果n不随新函数的变元组的变化而变化，则说该新函数是由诸旧函数利用迭置而得；如果n随着新函数的变元组而变化，则说该新函数是由诸旧函数利用算子法而得.

例如，设有后继函数$S(x)$，又设a为常数，y为变元，现作下列函数：$S(x)$，$S(S(x))$(缩写为 $S^2(x)$)，$S(S^2(x))$(缩写为$S^3(x)$)，…，$S(S^{a-1}(x))$(缩写为$S^a(x)$)…

$S(S^{y-1}(x))$(缩写为$S^y(x)$)

先看 $S^a(x)$. 因为a为常数，所以 $S^a(x)$仍为一元函数. $S^a(x)$在x_0处的值与$S(x)$在$S^{a-1}(x_0)$处的值有关，而$S^{a-1}(x_0)$的值与 $S(x)$在$S^{a-2}(x_0)$处的值有关，这样一直下去，可知与$S(x_0)$的值有关，而$S(x_0)$的值与$S(x)$在x_0处的值有关. 因此$S^a(x_0)$在 x_0处的值与x_0，$S(x_0)$，$S^2(x_0)$，…，$S^{a-1}(x_0)$等a个值有关. 因为a为常数，不随x_0的变化而变化，所以$S^a(x)$可以由$S(x)$利用迭置而得.

再看$S^y(x)$. 因为y为变元, 所以$S^y(x)$是一个依赖于x, y的二元函数. 该二元函数在(x_0, y_0)处的值与$S(x)$在x_0, $S(x_0)$, $S^2(x_0)$, …, $S^{y_0-1}(x_0)$等y_0个值有关. 因为y_0是随(x_0, y_0)的变化而变化, 所以$S^y(x)$是由$S(x)$利用算子法而得, 而不是利用迭置而得.

不难看出 $$S^a(x)=x+a \tag{1}$$

$$S^y(x)=x+y \tag{2}$$

在(1)中把x代以零函数$O(x)$得

$$S^a O(x)=O(x)+a=0+a=a=C_a(x)$$

即 $$S^a O(x)=C_a(x) \tag{3}$$

由此可知, 常值函数可由后继函数和零函数利用迭置而得, 加法可由后继函数利用算子法而得.

下面我们分别讨论迭置与算子. 现在先讨论迭置法.

迭置法的形式较多, 其标准形式是(m, n)迭置. 所谓(m, n)迭置是指下列形式的迭置.

设有一个m元函数 $f(x_1, x_2, \cdots, x_m)$, m个 n 元函数 $g_1(x_1, \cdots, x_n)$, …, $g_m(x_1, \cdots, x_n)$. 由f与诸g作下列函数

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

此种方法叫做(m, n)迭置, 又叫做多多迭置; 并称新函数 h 是由诸 g 对 f 作(m, n)迭置而得, 表为

$$h=f(g_1, \cdots, g_m)$$

从而有 $$h(x_1, \cdots, x_n)=f(g_1, \cdots, g_m)(x_1, \cdots, x_n)$$

这种表示方法既简便又明了, 今后将经常采用.

应该注意, 在(m, n)迭置中, 有三点要求:

1. 诸 g 的个数应与 f 的元数相等(等于m);
2. 诸 g 的元数均相等(都是n元的函数);
3. 诸 g 的自变元都相同(都是$x_1, \cdots, x_n$).

显然通常的迭置很少满足这三个要求, 因而不是(m, n)迭置,

而呈各种不同的形式，但是可以证明，利用本原函数，各种迭置均可化为(m, n)迭置．因为迭置的形式极多，证明就比较麻烦．然而化归的方法却很简单，从下面的例子中，不难看出这一论断的正确性．

例：试利用本原函数把下面的迭置化为(m, n)迭置．

$h(x_1, x_2, x_3, x_4)=f(5, g_1(x_4, x_3, 0), g_2(x_1, x_2, x_2))$

[解]

作 $h_1(x_1, x_2, x_3, x_4)=C_5I_{41}(x_1, x_2, x_3, x_4)$（它是$(1, 4)$迭置）

$$=S^5OI_{41}(x_1, x_2, x_3, x_4)$$

（C_5是由S和O作五次$(1, 1)$迭置而得）

则有 $h_1(x_1, x_2, x_3, x_4)=5$

作 $h_2=OI_{41}$ （它是$(1, 4)$迭置）

则有 $h_2(x_1, x_2, x_3, x_4)=Ox_1=0$

作 $h_3=g_1(I_{44}, I_{43}, h_2)$ （它是$(3, 4)$迭置）

则有 $h_3(x_1, x_2, x_3, x_4)=g_1(I_{44}, I_{43}, h_2)(x_1, x_2, x_3, x_4)$

$$=g_1(x_4, x_3, 0)$$

作 $h_4=g_2(I_{41}, I_{42}, I_{42})$ （它是$(3, 4)$迭置）

则有 $h_4(x_1, x_2, x_3, x_4)=g_2(I_{41}, I_{42}, I_{42})(x_1, x_2, x_3, x_4)$

$$=g_2(x_1, x_2, x_2)$$

故 $h=f(h_1, h_3, h_4)$

此为$(3, 4)$迭置，即为所求之结果．

由此可知，如果原迭置中有用常数代入的情形（称为变元特化，上例中f的第一变目，g_1的第三变目均属此情形），化归成(m, n)迭置时可用常值函数代替，而常值函数又可用本原函数作出．如果原迭置中有用函数代入的情形，且所代入的函数的变元与新函数的变元不一致（上例中f的第二、三变目均属此情形），则化归成(m, n)迭置时可用广义么函数使它们一致起来．

数学中经常使用的显式定义方法，本质上说是迭置法．例如：

$$\mathrm{tg}x=\sin x/\cos x$$

是关于$\mathrm{tg}x$的显式定义，它由两步组成：第一步是由y/x，$\sin x$，$\cos x$这三个函数作迭置而得$\sin x/\cos x$；第二步是把由迭置而得的新函数$\sin x/\cos x$，记为$\mathrm{tg}x$．显然，显式定义的重点是第一步，即迭置．因此，今后我们把显式定义法看作是与迭置法相同的方法，统称为迭置法（也称为复合法，代入法）．

数学中经常使用的另一种定义函数的方法是凑合定义法（又称分别情形定义法），表面上它与迭置不同，但实际上却可归为迭置．

所谓凑合定义法是指下列形式的构造新函数的方法：

$$h(x_1,\cdots,x_n)=\begin{cases}f_1(x_1,\cdots,x_n), & \text{当}A_1(x_1,\cdots,x_n)\text{成立时}\\ f_2(x_1,\cdots,x_n), & \text{当}A_2(x_1,\cdots,x_n)\text{成立时}\\ \cdots & \cdots\\ f_k(x_1,\cdots,x_n), & \text{当}A_k(x_1,\cdots,x_n)\text{成立时}\end{cases}$$

新函数 h 称为由旧函数 $f_1,\cdots,f_k$ 以及 $A_1,\cdots,A_k$ 利用凑合定义而得．其中诸条件 $A_1,\cdots,A_k$ 当然是数论语句，并要诸 $A_1,\cdots,A_k$ 之间相互穷尽且互不可兼，即对变元组任一组给定的值$(x_1,\cdots,x_n)$，必有一条件且只有一条件 A_i 成立．

欲把凑合定义化归为迭置，可利用上一节给出的$max(x,y)$及xNy两函数，并把 $max(max(x,y),z)$记为$max(x,y,z)$，$max(max(x,y,z),u)$记为$max(x,y,z,u)$等等，这样便得

$$h(x_1,\cdots,x_n)=max(f_1(x_1,\cdots,x_n)NctA_1(x_1,\cdots,x_n),\cdots,$$
$$f_k(x_1,\cdots,x_n)NctA_k(x_1,\cdots,x_n))$$

这就是说，有了 max及xNy后，上面的凑合定义便可化归为迭置．

其次讨论算子．

算子的类型也很多. 这里我们介绍几种重要的常用的算子.

3.2.1 原始复迭式

设有一元函数$f(x)$, 现作下列函数:

$f(x), f^2(x)$(即$f(f(x))$), $f^3(x)$(即$f(f^2(x))$), …, $f^n(x)$(即$f(f^{n-1}(x))$), 并约定$f^0(x)=x$, $f^n(x)$可以看作依赖于x及n的二元函数, 记为$g(x,n)$,

即
$$g(x,n)=f^n(x) \tag{3}$$

也即

$$\begin{cases} g(x,0)=x \\ g(x,n+1)=f^{n+1}(x)=f(f^n(x)) \end{cases}$$

则说$g(x,n)$是$f(x)$利用原始复迭式而得, 又可记为

$$g(x,n)=\underset{t\to(x,n)}{itr}\, f(t)$$

也即

$$\begin{cases} g(x,0)=\underset{t\to(x,0)}{itr}\, f(t)=x \\ g(x,n+1)=\underset{t\to(x,n+1)}{itr}\, f(t)=f(\underset{t\to(x,n)}{itr}\, f(t)) \end{cases}$$

其中t叫做作用变元, x,n叫做新添变元, x又名初值变元, $f(t)$叫做作用域, 作用域中除作用变元以外的其它变元(如果有的话)均叫做参数.

原始复迭式的应用很广泛, 上面举的由$S(x)$造$x+y$的例子, 就是利用原始复迭式而得, 即

$$x+y=S^y(x)=\underset{t\to(x,y)}{itr}\, S(t) \tag{5}$$

由$x+y$利用原始复迭式还可作出$x\cdot y$, 作法如下:

$$x\cdot y=\underset{t\to(0,y)}{itr}\,(x+t)\text{(其中}x\text{为参数)} \tag{6}$$

证明 令$f_x(t)=x+t$, 则$\underset{t\to(0,y)}{itr}\,(x+t)=f_x^y(0)$

奠基　　$y=0$时

(6)的左端$=x\cdot 0=0$

(6)的右端$=\underset{t\to(0,0)}{itr}\ (x+t)=f_x^0(0)=0$

故两端相等.

归纳　$y=n+1$ 时

(6)的右端$=\underset{t\to(0,n+1)}{itr}\ (x+t)=f_x^{n+1}(0)=f_x(f^n(x))$

$=f_x(x\cdot n)$　　(归纳假设)

$=x+(x\cdot n)=x\cdot(n+1)=$(6)的左端

故(6)成立, 证毕.

由 $x\cdot y$ 利用原始复迭式还可作出 x^y, 作法如下:

$x^y=\underset{t\to(1,y)}{itr}x\cdot t$(其中 x 为参数)　　(7)

证明　令 $f_x(t)=x\cdot t$ 则$\underset{t\to(1,y)}{itr}\ x\cdot t=f_x^y(1)$

奠基　$y=0$ 时

(7)的右端$=\underset{t\to(1,0)}{itr}\ x\cdot t=f_x^0(1)$

$=1=x^0=$(7)的左端

归纳　$y=n+1$ 时

(7)的右端$=\underset{t\to(1,n+1)}{itr}(x\cdot t)=f_x^{n+1}(1)$

$=f_x(f_x^n(1))$

$=f_x(x^n)$　　(归纳假设)

$=x\cdot(x^n)=x^{n+1}=$(7)的左端

故(7)成立, 证毕.

由(5)、(6)、(7)可知, $x+y$, $x\cdot y$, x^y 都可以由 $S(x)$利用原始复迭式而得.

3.2.2　迭函算子

设有一个二元函数 $A(x,y)$, 一个一元函数 $f(x)$, 利用它们作下列函数:

$$g(0)=f(0)$$
$$g(1)=Ag(0)f(1)=Af(0)f(1)$$
$$g(2)=Ag(1)f(2)=A^2f(0)f(1)f(2)$$
……
$$g(n+1)=Ag(n)f(n+1)=A^{n+1}f(0)f(1)\cdots f(n+1)$$

显然，$g(n)$依赖于函数 $A(x,y)$和 $f(x)$. 若把 A 固定，而把 $f(x)$看作被改造的函数，即把新函数 $g(n)$看作由旧函数 $f(x)$(和某一常函数 A)改造而得，则说新函数 $g(n)$是由旧函数 $f(x)$利用迭函算子(亦可称为迭 A 算子)而得，记为

$$g(n)=\underset{x\to n}{A}f(x)$$

其中 x 叫做作用变元，n 则叫做新添变元.

当 A 给定为各种常函数时便可得到各种迭 A 算子，其中最常用的有下列四种:

1. 迭加算子. 取 A 为加法，并把"$\underset{t\to n}{+}$"写为"$\sum\limits_{t\to n}$"时得

$$\sum_{t\to n}f(t)=f(0)+f(1)+\cdots+f(n)$$

2. 迭乘算子. 取 A 为乘法，并把"$\underset{t\to n}{\times}$"写为"$\prod\limits_{t\to n}$"时得

$$\prod_{t\to n}f(t)=f(0)\cdot f(1)\cdot\cdots\cdot f(n)$$

3. 迭大算子. 取 A 为 max 时得

$$\underset{t\to n}{max}f(t)=max(f(0),f(1),\cdots,f(n))$$

4. 迭小算子. 取 A 为 min 时得

$$\underset{t\to n}{min}f(t)=min(f(0),f(1),\cdots,f(n))$$

3.2.3 原始递归式

我们知道，由 $A(x,y)$和 $f(x)$可作

$$\begin{cases} g(0)=f(0) \\ g(n+1)=A(g(n),\ f(n+1)) \end{cases}$$

在上面，我们是固定 A，而把 f 看作被改造的函数，从而得迭函算子，反之，现把 f 固定，取为 $D(x)$，而把 A 看作被改造的函数，即把新函数 $g(n)$ 看作是由旧函数 $A(x,y)$ 经如下改造而得

$$g(0)=0$$

$$g(n+1)=A(g(n),\ D(n+1))=A(g(n),n)$$

则说 $g(n)$ 是由旧函数 $A(x,y)$ 利用*原始递归式*而得.

原始递归式的一般形式如下:

$$\begin{cases} g(0)=a \\ g(n+1)=B(n,g(n)) \end{cases} \tag{8}$$

其中 a 为常数，$B(x,y)$ 为已知函数. 此式称为无参数的原始递归式的标准形式.

$$\begin{cases} g(u_1,\cdots,u_r,0)=A(u_1,\cdots,u_r) \\ g(u_1,\cdots,u_r,S(n))=B(u_1,\cdots,u_r,n,g(u_1,\cdots,u_r,n)) \end{cases} \tag{9}$$

其中 $A(u_1,\cdots,u_r)$，$B(u_1,\cdots u_r,x,y)$ 是已知函数. 此式称为含参数的原始递归式的标准形式.

显然新函数 g 是由旧函数 A 和 B 改造而得. 这个定义式的特点是在第二个定义式的右端出现被定义的函数 g，但左端是 $g(u_1,\cdots,u_r,S(x))$，右端是 $g(u_1,\cdots,u_r,x)$ 是两个不同的填式. 由定义式易知，只要 B，A 处处有定义，则新函数 g 也处处有定义，先由第一个定义式得到 g 在 $(u_1,\cdots,u_r,0)$ 处的值，再由第二个定义式得到 g 在 $(u_1,\cdots,u_r,1)$ 处的值，在 $(u_1,\cdots,u_r,2)$ 处的值，在 $(u_1,\cdots,u_r,3)$、$(u_1,\cdots,u_r,4)$ 等任意处的值(严格的证明，应该用数学归纳法).

$$x+y,\ x\cdot y,\ x^y,\ \max_{x\to n}f(x),\ \min_{x\to n}f(x),$$

$$\sum_{x\to n}f(x),\ \prod_{x\to n}f(x)$$

均可用原始递归式定义:

1) $x+y$

$$\begin{cases} x+0=x \\ x+Sy=S(x+y) \end{cases}$$

这里A为I(么函数), B为SI_{22}(后继函数与广义么函数的迭置).

2) $x\cdot y$

$$\begin{cases} x\cdot 0=0 \\ x\cdot Sy=x\cdot y+x \end{cases}$$

这里A为O(零函数), B为加法.

3) x^y

$$\begin{cases} x^0=1 \\ x^{Sy}=x^y\cdot x \end{cases}$$

这里A为C_1(常值函数), B为乘法.

4) $\max\limits_{x\to n} f(x)$

$$\begin{cases} \max\limits_{x\to 0} f(x)=f(0) \\ \max\limits_{x\to Sn} f(x)=max(\max\limits_{x\to n} f(x), f(Sn)) \end{cases}$$

这里A为常数$f(0)$; $B(x,y)$为$max(y, f(Sx))$.

5) $\min\limits_{x\to n} f(x)$

$$\begin{cases} \min\limits_{x\to 0} f(x)=f(0) \\ \min\limits_{x\to Sn} f(x)=min(\min\limits_{x\to n} f(x), f(Sn)) \end{cases}$$

这里A为常数$f(0)$, $B(x,y)$为$min(y, f(Sx))$.

6) $\sum\limits_{x\to n} f(x)$

$$\begin{cases} \sum\limits_{x\to 0} f(x)=f(0) \\ \sum\limits_{x\to Sn} f(x)=\sum\limits_{x\to n} f(x)+f(Sn) \end{cases}$$

这里 A 为常数 $f(0)$, $B(x,y)$ 为 $y+f(Sx)$.

7) $\prod\limits_{x\to n} f(x)$

$$\begin{cases}\prod\limits_{x\to 0} f(x)=f(0)\\ \prod\limits_{x\to Sn} f(x)=\prod\limits_{x\to n} f(x)\cdot f(Sn)\end{cases}$$

这里 A 为常数 $f(0)$, $B(x,y)$ 为 $y\cdot f(Sx)$.

3.2.4 一般递归式

设有二元函数 $B(x,y)$, 一元函数 $g(x)$, a 为常数, 现在作下列函数:

$$\begin{cases}f(0)=a\\ f(Sx)=B(x,f(g(Sx)))\end{cases}\tag{10}$$

则说新函数 $f(x)$ 是由函数 $B(x,y)$, $g(x)$ 利用*半递归式*(又称*部分递归式*)而得. 如果其中的 $g(x)$ 具有下列性质: 对于任何 x, 恒有一数 m, 使得

$$g^m(x)=0\ (\text{即}\ \underset{t\le\psi(x,m)}{itr}\ g(t)=0)\tag{11}$$

则说新函数 $f(x)$ 是由旧函数 $B(x,y)$, $g(x)$ 利用*一般递归式*(又称*有序递归式*)而得, 而有性质 (11) 的函数 $g(x)$ 称为在 x 处归宿于 0 的函数, 处处都归宿于 0 的函数简称为*归宿函数*.

设 $g(x)$ 是归宿函数, 因此对于任何 x, 至少有一数 m 使得

$$g^m(x)=0$$

显然 m 与 x 有关, 即 m 随 x 而变化, 也即 m 是 x 的函数, 故可记为 $m(x)$. 而且具有性质 (11) 的 $m(x)$ 可能不止一个, 甚至可能有无穷多个, 但至少有一个, 既然如此, 易见在这样的数中必有一个最小者(这便是有名的最小数原理). 设 $d(x)$ 是 $g(x)$ 在 x 处具有性质(11)的数中最小者, 则称该 $d(x)$ 是 $g(x)$ 在 x 处的*归宿步骤*,

记为

$$d(x)=\underset{t\to x}{stp}\ g(t) \tag{12}$$

这个式子称为由 $g(x)$ 造新函数 $d(x)$ 的归宿步骤式.

(10) 是无参数一般递归式的标准形式, 下式是含有参数的一般递归式的标准形式:

$$\begin{cases} f(u_1,\cdots,u_r,0)=A(u_1,\cdots,u_r) \\ f(u_1,\cdots,u_r,Sx)=B(u_1,\cdots,u_r,xf(u_1,\cdots,u_r,g(u_1, \\ \cdots,u_r,Sx))) \end{cases} \tag{13}$$

其中 A, B, g 为已知函数, 而且 g 是归宿于 0 的函数.

任给一定元组 $(u_1,\cdots,u_r,x)$, f 在该变元组处的值可如下计算: 如果 $x=0$, 则由第一式立即可求得在该变元组处的值, 如果 $x\neq 0$, 则由第二式向前追溯 $d(x)$ 次, 便可化归为计算 $f(u_1,\cdots,u_r,g^{dx}(x))$ 的值, 因为 $g^{dx}(x)=0$ 所以

$$f(u_1,\cdots,u_r,g^{dx}(x))=f(u_1,\cdots,u_r,0)=A(u_1,\cdots,u_r)$$

从而可以计算出 f 在所给变元组处的值, 由此可知, 只要 A、B 处处有定义; g 是归宿于 0 的函数, 那么由它们利用一般递归式(13)所定义的函数 f 必是处处有定义的函数(严格证明应该用归纳法).

现考察前驱函数 Dx. 对于任何 x, 恒有 $D^x(x)=0$, 故 $D(x)$ 是归宿于 0 的函数. 今取 $g(x)=D(x)$, 则(10)可化归成

$$\begin{cases} f(0)=a \\ f(Sx)=B(x,f(g(Sx)))=B(x,f(D(Sx)))=B(x,f(x)) \end{cases}$$

此即无参数原始递归式的标准形式. 同样, (13)也可化归成含有参数的原始递归式的标准形式, 因此, 原始递归式是一般递归式的特例.

还可以证明, 归宿步骤式(12)也是一般递归式的特例. 因为 $g^0(x)=x$, 故 g 在 $x=0$ 处的归宿步骤永为 0, 即 $d(0)=0$. 再讨论 g 在 $x+1$ 处的归宿步骤 $d(x+1)$. 因为 $g^{d(x+1)}(x+1)=0$, 所

以 $g^{d(x+1)-1}g(x+1)=0$，由归宿步骤定义知，应有 $d(x+1)\dot{-}1\geqslant d(g(x+1))$现证 $d(x+1)\dot{-}1=d(g(x+1))$. 这可用反证法证明之，如果 $d(x+1)\dot{-}1\neq d(g(x+1))$，则应有 $d(x+1)\dot{-}1>d(g(x+1))$，所以 $d(x+1)>d(g(x+1))+1$. 因为 $g^{d(g(x+1))}(g(x+1))=0$，所以 $g^{d(g(x+1))+1}(x+1)=0$，这样我们便找到一个比 $d(x+1)$更小的数即 $d(g(x+1))+1$，它使得 $g^{d(g(x+1))+1}(x+1)=0$ 成立，因而 $d(x+1)$不是 g 在 $x+1$ 处的归宿步骤. 显然这与 $d(x)$的定义矛盾. 故由反证法得证. 由上可知:

$$\begin{cases} d(0)=0 \\ d(x+1)=d(g(Sx))+1 \end{cases}$$

这是一般递归式，即可由一般递归式来定义归宿步骤 $d(x)$，因此，归宿步骤式是一般递归式的特例.

3.2.5 摹状式

在数学中经常用隐式定义法来定义新函数，所谓隐式定义是指先由若干个已知函数作成一个方程(代数方程、微分方程等等)，再由该方程定义出一个新函数的方法. 例如，由减法和乘法可以作出下列方程:

$$t^3-x^5=0$$

由数学知识可知，在实数范围内，任给一个 x，有一个也只有一个 t 满足该方程，而且这个 t 就是$\sqrt[3]{x^5}$(通常称为方程的根). 这是一个新函数，是由该方程而唯一定义的，这个新函数还可表述为"在实数范围内，使得 $t^3-x^5=0$成立的那个 t"，用上一章摹状词一节中的记号，此语句可表为

$$\iota t(t^3-x^5=0)$$

这个式子称为摹状式，新函数可以用这个摹状式来定义.

当然，隐式定义并没有限定非要用方程式来定义不可. 事实上，我们可以用任一个条件来定义新函数. 例如，关于极限的定义

就是用下面的条件来定义的，$f(x)$在$x=a$点的极限，$\lim\limits_{x\to a}f(x)$，是指满足下列条件的l：

$\forall e\exists\delta\forall n((0<|h|<\delta)\supset(|f(a+h)-l|<e))$，一般地说，任给一条件$A(u_1,\cdots,u_r,t)$，只要对于任一变元组$(u_1,\cdots,u_r)$均有一个也只有一个$t$满足该条件，那么便可以用"满足条件$A$的$t$"来定义这个$t$．因为任给一个变元组$(u_1,\cdots,u_r)$都有相应的一个$t$，所以$t$是$u_1,\cdots,u_r$的函数，记为$f(u_1,\cdots,u_r)$，用数理逻辑的记号，可记为

$$\iota tA(u_1,\cdots,u_r,t)$$

即　　$f(u_1,\cdots,u_r)=\iota tA(u_1,\cdots,u_r,t)$

我们说，新函数是由条件A利用摹状式而得，f又称为A的根．

应该注意的是，一般说任给$(u_1,\cdots,u_r)$不一定存在唯一的t使得$A(u_1,\cdots,u_r,t)$成立．因此由A利用摹状式构造的新函数f不一定处处有定义，即f是部分函数．如果要求对于任何$(u_1,\cdots,u_r)$均存在唯一的t使得$A(u_1,\cdots,u_r,t)$成立，则说新函数f是由A利用正常摹状式而得．

现在的问题是，当A不满足存在唯一性条件时应该怎么办？不满足存在唯一性条件的情形有两种，一是有多个t（至少有两个t）使得A成立，即满足存在性不满足唯一性，二是没有t使得A成立，即存在性唯一性均不满足．

对于第一种情形，一般都是添加一些新条件，使得同时满足原条件和新条件的t有一个也只有一个，这样便得到正常摹状式了．例如，在实数域中，满足$t^2-2=0$的t有二个，但附加条件"$t>0$"后，满足"$t^2-2=0\wedge t>0$"的t便只有一个，因此$\iota t(t^2-2=0\wedge t>0)$是正常摹状式．

在数学中．添加新条件使得唯一性条件成立，并不是一件容易的事，往往是十分困难的．但是在递归函数论中却很简单，因为

递归函数论讨论的范围始终限于自然数，所以只要有 t 使得 $A(u_1, \cdots, u_r, t)$成立那么在这样的 t 中必有一最小者（通常称此为最小数原理），因此，只要有 t 使得 A 成立，那么使得 $A(u_1, \cdots, u_r, t)$ 成立且为最小者的 t 必有一个也只有一个，即有一个也只有一个 t 使得下列条件成立：

$$A(u_1, \cdots, u_r, t,) \wedge \forall t^*(t^* < t \supset \bar{A}(u_1, \cdots, u_r, t^*))$$

依照数理逻辑的习惯记法，“满足条件 A 的最小的 t”记为

$$\mu t A(u_1, \cdots, u_r, t)$$

这个式子仍称为摹状式．虽然这种摹状式与 A 的唯一性条件无关，但与 A 的存在性条件有关．凡要求 A 满足存在性条件的摹状式 $\mu t A(u_1, \cdots, u_r, t)$也叫正常摹状式．

对于第二种情形，处理方法很多，本章中我们规定，当在$(u_1^0, \cdots, u_r^0)$处没有 t 使得 $A(u_1^0, \cdots, u_r^0, t)$成立时，上面这两种摹状式仍允许使用，但说新函数 $\iota t A(u_1, \cdots, u_r, t)$和 $\mu t A(u_1, \cdots, u_r, t)$在$(u_1^0, \cdots, u_r^0)$处无定义．这是数学中经常采用的一种方法，例如，$1/x$在 $x=0$ 处无定义．

由于 $A(u_1, \cdots, u_r, t) \vee (t=x)$始终满足存在性条件（因为当 t 为 x 时，该条件成立，所以至少有一个 t 满足该条件），所以$\mu t((A(u_1, \cdots, u_r, t) \vee (t=x))$为正常摹状式，通常记为

$$\mu_{t \to x} A(u_1, \cdots, u_r, t)$$

这个式子称为受限摹状式，又称为受限于 x 的摹状式．显然使用受限摹状式时无须先作存在性证明，更不必作唯一性证明．这是因为当 $A(u_1, \cdots, u_r, t)$在 x 以下（包括 x，今后语句“x 以下”均指包括 x 在内而言）有 t 根时，$\mu_{t \to x} A(u_1, \cdots, u_r, t)$恰巧是 $A(u_1, \cdots, u_r, t)$的最小 t 根；当 $A(u_1, \cdots, u_r, t)$在 x 以下没有 t 根时$\mu_{t \to x} A(u_1, \cdots,$

$u_r, t)$就是 x, 即

$$\underset{t\to x}{\mu} A(u_1, \cdots, u_r, t) = \begin{cases} \mu t A(u_1 \cdots, u_r, t), & \text{当在 } x \text{ 以下有 } t \text{ 满足 } A \text{ 时} \\ x, & \text{当在 } x \text{ 以下无 } t \text{ 满足 } A \text{ 时} \end{cases}$$

由此可知恒有$\underset{t\to x}{\mu} A(u_1, \cdots, u_r, t) \leqslant x$.

必须注意，这里 $A(u_1, \cdots, u_r, t)$是"条件"，即是含 $u_1, \cdots, u_r, t$ 的语句，而不是函数，因此，上面三种摹状式不是由旧函数构造新函数的算子，而是由语句构造新函数的"准算子"。对此，应给以改进，使之真正成为由旧函数而造新函数的算子，这是很容易办到的，只要利用特征函数便可.

设语句 $A(u_1, \cdots, u_r, t)$的准特征函数为 $a(u_1, \cdots, u_r, t)$，显然有

$$\iota t A(u_1, \cdots, u_r, t) = \iota t (a(u_1, \cdots, u_r, t) = 0)$$

$$\mu t A(u_1, \cdots, u_r, t) = \mu t (a(u_1, \cdots, u_r, t) = 0)$$

$$\underset{t\to x}{\mu} A(u_1, \cdots, u_r, t) = \underset{t\to x}{\mu} (a(u_1, \cdots, u_r, t) \cdot (t \doteq x) = 0)$$

现引入下列记号:

把$\iota t (a(u_1, \cdots, u_r, t) = 0)$记为

$$\underset{t}{r\iota u}\, a(u_1, \cdots, u_r, t)$$

把$\mu t (a(u_1, \cdots, u_r, t) = 0)$记为

$$\underset{t}{r\iota i}\, a(u_1, \cdots, u_r, t)$$

把$\underset{t\to x}{\mu} (a(u_1, \cdots, u_r, t) \cdot (t \doteq x) = 0)$记为

$$\underset{t\to x}{r\iota i}\, a(u_1, \cdots, u_r, t).$$

因为 $a(u_1, \cdots, u_r, t)$是数论函数，所以 $\underset{t}{r\iota u}$, $\underset{t}{r\iota i}$, $\underset{t\to x}{r\iota i}$ 均是由旧函数构造新函数的算子. 前两者称为摹状算子. 后者称为受限摹状算子. 如果要求对于任何$(u_1, \cdots, u_r)$, $a(u_1, \cdots, u_r, t) = 0$ 都存在唯一的 t 根(存在 t 根)，则$\underset{t}{r\iota u}$($\underset{t}{r\iota i}$)称为正常摹状算子.

由于通常把一个方程式 $f(x) = 0$ 的根叫做函数 $f(x)$的零点，

所以 $\underset{t}{rtu}a(u_1, \cdots, u_r, t)$ 可读作 "$a(u_1, \cdots, u_r, t)$ 的唯一 t 零点"，$\underset{t}{rti}\, a(u_1, \cdots, u_r, t)$ 可读作"$a(u_1, \cdots, u_r, t)$ 的最小 t 零点"，$\underset{t\to x}{rti}\, a(u_1, \cdots, u_r, t)$ 可读作"$a(u_1, \cdots, u_r, t)$ 的 x 以下的最小 t 零点".

此外，我们再引进三个受限幕状算子.

$$\underset{t\to x}{rtu}\, a(u_1, \cdots, u_r, t) = \begin{cases} \underset{t}{rtu}\, a(u_1, \cdots, u_r, t), & (\text{当在}(0, x)\text{中 } a(u_1, \cdots, u_r, t) \text{ 有唯一 } t \text{ 零点时}) \\ x, & (\text{此外}) \end{cases}$$

$$\underset{t\to x}{rta}\, a(u_1, \cdots, u_r, t) = \begin{cases} (0, x)\text{中} a(u_1, \cdots, u_r, t)\text{的最大 } t \text{ 零点}, \\ \quad (\text{当在}(0, x)\text{中} a(u_1 \cdots, u_r, t) \text{有 } t \text{ 零点时}) \\ x, (\text{此外}) \end{cases}$$

$\underset{t\to x}{rtn}\, a(u_1, \cdots, u_r, t) =$ 在 x 以下 $a(u_1, \cdots, u_r, t)$ 的 t 零点的个数.

在今后的讨论中，(正常)幕状式以 $\underset{t}{rti}$(即 μt)为代表，受限幕状式以 $\underset{t\to x}{rti}$ (即 $\underset{t\to x}{\mu}$)为代表.

上面我们介绍了五种算子. 最后再介绍算子的一般形式.

设有一算子 α，它把 k 个 m 元函数 $f_1(x_1, \cdots, x_m)$，$\cdots$，$f_k(x_1, \cdots, x_m)$ 改造成为一个 n 元函数 $g(y_1, \cdots, y_n)$，则可记为

$$g(y_1, \cdots, y_n) = \underset{\langle x\rangle\to\langle y\rangle}{\alpha}(f_1(x_1, \cdots, x_m), \cdots, f_k(x_1, \cdots, x_m))$$

其中 $\langle x\rangle$ 和 $\langle y\rangle$ 分别是矢列 $(x_1, \cdots, x_m)$ 和 $(y_1, \cdots, y_n)$ 的缩写. 这里诸 $x_1, \cdots, x_m$ 叫做作用变元，诸 $y_1, \cdots, y_n$ 叫做新添变元，$(f_1(x_1, \cdots, x_m), \cdots, f_k(x_1, \cdots, x_m))$ 叫做 α 的作用域，作用域中除作用变元外的其它变元(如果有的话)叫做参变元.

如果对于诸 $f_i(x_1, \cdots, x_m)$ 及任意一组值 $(y_1, \cdots, y_n)$，只要 $\underset{\langle x\rangle\to\langle y\rangle}{\alpha}(f_1(x_1, \cdots, x_m), \cdots, f_k(x_1, \cdots, x_m))$ 有定义，便可在有限步骤内求得其值，则说 α 是半能行算子. 如果能在有限步内判知它是否有定义，则进一步称为能行算子. 如果尽管 $\underset{\langle x\rangle\to\langle y\rangle}{\alpha}(f_1(x), \cdots, f_k(x))$ 有定义，但未必能在有限步骤内算出其值，则说 α 是非能行算子.

由此定义可知，原始复迭算子，迭函算子，原始递归算子，一般递归算子，正常幂状算子，受限幂状算子均是能行算子；幂状算子 $\underset{x}{rti}$，$\underset{x}{rtu}$ 是半能行算子，部分递归算子也是半能行算子；$\underset{x}{rta}$ 是非能行算子。

习　题

1. 试把下列迭置化归为 (m, n) 迭置：

1.1　$h(x_1, x_2, x_3)=f(x_2, g_1(x_1, x_3), 4, g_2(x_1))$；

1.2　$h(x_1, x_2, x_3)=f(x_3, g_1(x_1, x_3), x_2, x_3)$。

2. 验证下列各式：

2.1　$N^2max(x, y)=max(N^2x, N^2y)$；

2.2　$N^2min(x, y)=min(N^2x, N^2y)$；

2.3　$xNyNz=xNzNy=xN(y+z)$；

2.4　$xNy=x\cdot(1\dot{-}y)=[x/(x\cdot y+1)]=[x/(1\dot{-}y)]$。

3. 求分别用 $\underset{x>(10,4)}{itr}$，$\sum_{x\to 10}$，$\prod_{x\to 10}$，$\underset{x\to 10}{max}$，$\underset{x\to 10}{min}$ 等五个算子对下列函数改造的结果：

3.1　$f(x)=x^2+x\dot{-}3$；

3.2　$f(x)=(3x^2+2x\dot{-}4)N(2x\dot{-}4)$。

4. 用原始递归式定义下列函数：

4.1　$D(x)$；

4.2　$x\dot{-}y$；

4.3　$x!$；

4.4　$\sum_{i\to n}(2i+1)$。

5. 试证原始复迭式和迭函算子是原始递归式的特例。

6. 用显式表示下列函数：

6.1　$\begin{cases}g(0)=0,\\ g(n+1)=g(n)+(n+1)^2;\end{cases}$

6.2　$\begin{cases}g(x, 0)=f(0),\\ g(x, Sn)=g(x, n)\cdot x+f(Sn).\end{cases}$

7. 求证下列各函数均为归宿函数，并求出其归宿步骤：

7.1 $O(x)$；

7.2 $N(x)$；

7.3 $D(x)$；

7.4 $E(x)$.

7.5 $$g(x)=\begin{cases}x\dot{-}1, & \text{当 } x\leqslant 3 \text{ 时}\\ x+2, & \text{当 } x>3 \text{ 且 } x \text{ 非平方数时}\\ \sqrt{x}, & \text{当 } x>3 \text{ 且 } x \text{ 为平方数时}\end{cases}$$

8. 设

$$\begin{cases}f(0)=0\\ f(Sx)=B_1(x,f(x))\end{cases}$$

$$\begin{cases}g(0)=0\\ g(Sx)=B_2(x,g(x))\end{cases}$$

求证如果 $B_1(x,y)\leqslant B_2(x,y)$ 且 $B_2(x,y)$ 对 y 递增，则 $f(x)\leqslant g(x)$.

9. 用显式表示下列函数：

9.1 $\underset{t\to n}{rtu}(t\dot{-}x\dot{-}x)$；

9.2 $\underset{t}{\mu}(x\dot{-}t\cdot y=0)$；

9.3 $\underset{t}{rti}(x\dot{-}2t)$；

9.4 $\underset{t\to n}{\mu}(x+t\dot{-}y=0)$；

9.5 $\underset{t\to a+b}{rti}(a\dot{-}bt\dot{-}t^2)$；

9.6 $\underset{t\to n}{rta}(t\dot{-}x+x\dot{-}t)$；

9.7 $\underset{t\to n}{rtn}(x\dot{-}t+t\dot{-}x)$.

10. 求证当 n 以下 $a(u_1,\cdots,u_r,t)$ 有零点时 $\underset{t\to n}{rta}\,a(u_1,\cdots,u_r,t)=n\dot{-}\underset{t\to n}{rti}\,a(u_1,\cdots,u_r,n\dot{-}t)$

§3.3 函数的定义过程和各种函数集

上面已经介绍了定义函数的两种方法，一种是直接给出函数的运算法则，另一种是由旧函数利用迭置法或算子法来定义新函数．后者称为派生法．派生出来的新函数又可用来派生其它新函数，这样逐步派生下去，便可得到各种各样的函数．

现在我们考虑另一个问题，任给一个函数，试问这个函数是怎样从旧函数一步一步派生出来的？这就是求函数的定义过程问题．为此必须首先对定义过程下一个明确的定义．

定义 设有一函数序列 $g_1, g_2, \cdots, g_n$，如果它们满足下列条件：

i) 诸 g_i 或者是本原函数之一；

ii) 或者是函数 $A_1, \cdots, A_m$ 之一；

iii) 或者是由前面若干个 g 利用算子 $\alpha_1, \cdots, \alpha_k$ 之一而得；

iv) 或者是由前面若干个 g 利用迭置而得．

则说该函数序列 $g_1, \cdots, g_n$ 是函数 g_n 的定义过程，或称组成过程，而 $A_1, \cdots, A_m$ 叫做开始函数．更详细地说，函数序列 $g_1, \cdots, g_n$ 是由开始函数 $A_1, \cdots, A_m$ 出发，利用迭置及算子 $\alpha_1, \cdots, \alpha_k$ 而作函数 g_n 的定义过程(或组成过程)．

例 1：令 $A_1(x, y)=x+y$, $A_2(x, y)=x\cdot y$，求由 A_1, A_2 出发，利用迭置而作 $f(x, y)=((x+y)^2+y)^2$ 的定义过程．

[解] $g_1(x, y)=A_1(x, y)=x+y$ (开始函数)

$g_2(x, y)=A_2(x, y)=x\cdot y$ (开始函数)

$g_3(x, y)=I_{22}(x, y)=y$ (本原函数)

$g_4(x, y)=g_2(g_1, g_1)(x, y)=(x+y)^2$ (2.2 迭置)

$g_5(x, y)=g_1(g_4, g_3)(x, y)=(x+y)^2+y$ (2.2 迭置)

$g_6(x, y)=g_2(g_5, g_5)(x, y)=((x+y)^2+y)^2$ (2.2迭置)

因为 $g_6=f$，故 g_1—g_6 是所求的 f 的定义过程．

例 2：设 $A(x)=Sx$，求由 $A(x)$出发利用迭置与原始复迭式作 $((x+y)^2+y)^2$ 的定义过程．

[解] $g_0(x)=Sx$ (开始函数)

$g_1(x, y)=\underset{t\to(x, y)}{itr}\, g_0(t)=x+y$ (复迭式)

$$g_2(x,y)=\underset{t\to(0,x)}{itr}\,g_1(t,y)=x\cdot y \qquad \text{(复迭式)}$$

其余各步(g_3-g_8)同前，于是得所求定义过程.

例 3: 设$A_1(x,y)=x+y$, $A_2(x,y)=xNy$, $A_3(x)=rs(x,2)$ 求由这三个函数出发，利用迭置作 $dv(x,2)$的组成过程.

[解] 因为

$$dv(x,2)=\begin{cases}1, & \text{当 } x \text{ 为奇数时，即 } rs(x,2)=1 \text{ 时}\\ 2, & \text{当 } x \text{ 为偶数时，即 } rs(x,2)=0 \text{ 时}\end{cases}$$

所以

$$dv(x,2)=1+Nrs(x,2)$$

作下列函数:

$$g_1(x,y)=A_1(x,y)=x+y \qquad \text{(开始函数)}$$
$$g_2(x,y)=A_2(x,y)=xNy \qquad \text{(开始函数)}$$
$$g_3(x)=A_3(x)=rs(x,2) \qquad \text{(开始函数)}$$
$$g_4=0 \qquad \text{(本原函数)}$$
$$g_5=S \qquad \text{(本原函数)}$$
$$g_6(x)=SO(x)=1 \qquad \text{(1.1 迭置)}$$
$$g_7(x)=g_2(g_6,g_3)(x)=Nrs(x,2) \qquad \text{(2.1 迭置)}$$
$$g_8(x)=g_1(g_6,g_7)(x)=1+Nrs(x,2)=dv(x,2) \qquad \text{(2.1 迭置)}$$

故 g_1-g_8 为所求的定义过程.

由上可知，求一函数的定义过程并非一件容易的事，必须对所讨论的函数有透彻的认识才行.

下面定义几个重要的函数集.

定义 由本原函数及 $x+y$, $x\dot{-}y$, $x\cdot y$, $[y/x]$, $[\sqrt{x}]$这五个函数出发，利用迭置而得的函数称为五则函数. 所有五则函数组成的集合称为五则函数集.

由本原函数以及 $x \dot{-} y$ 出发，利用迭置，迭加$\left(\sum_{x\to n}\right)$、迭乘$\left(\prod_{x\to n}\right)$而得的函数称为初等函数．所有初等函数组成的集合称为初等函数集．

由本原函数出发，利用迭置和原始递归式而得的函数称为原始递归函数．所有原始递归函数组成的集合称为原始递归函数集．

由本原函数出发，利用迭置和一般递归式而得的函数称为一般递归函数．所有一般递归函数组成的集合称为一般递归函数集．

由本原函数出发，利用迭置和部分递归式而得的函数称为部分递归函数．所有部分递归函数组成的集合称为部分递归函数集．

由本原函数以及 $x+y, x\cdot y, eq(x,y)$ 出发，利用迭置和(正常)幂状算子 rti 而得的函数称为(正常)幂状函数．所有(正常)幂状函数组成的集合称为(正常)幂状函数集．

凡其特征函数为五则函数的谓词均称五则谓词．凡其特征函数为初等函数的谓词均称为初等谓词．原始递归谓词、一般递归谓词等仿此定义．

定义 设有一函数集 Δ．如果只要函数 $f_1,\cdots,f_n$ 在 Δ 中，则由 $f_1,\cdots f_n$ 出发，利用迭置而作成的函数也在 Δ 中，那么称 Δ 对迭置是封闭的．如果只要函数 $f_1,\cdots,f_n$ 在 Δ 中，则由 $f_1,\cdots,f_n$ 出发，利用算子 α 而作成的函数也在 Δ 中，那么称 Δ 对算子 α 是封闭的．

由定义易知，上面五个函数集均对迭置封闭．初等函数集对迭加迭乘是封闭的．原始递归函数集对原始递归式是封闭的．一般递归函数集对一般递归式是封闭的．幂状函数集对幂状式是封闭的．

定义 设有一函数集 Δ，如果只要谓词 A、B 的特征函数在Δ中，那么$\bar{A}(A\vee B, A\wedge B, A\supset B, A\equiv B)$的特征函数也在$\Delta$中，则说该函数集$\Delta$对否定词(析取词，合取词，蕴涵词，等价词)是封闭的.

如果只要谓词 Ax 的特征函数在Δ中，那么$\underset{x\to n}{\forall} Ax(\underset{x\to n}{\exists} Ax)$的特征函数也在$\Delta$中，则说$\Delta$对受限全称(存在)量词是封闭的.

如果一函数集对否定词等五个命题联结词均是封闭的，则说该函数集对命题联结词封闭. 如果一函数集对受限全称量词和受限存在量词都是封闭的，则说该函数集对受限量词是封闭的.

定理 对于一个函数集Δ，如果它对迭置封闭，并且含有 Nx 和$x+y$，那么Δ对命题联结词是封闭的；又如果Δ对 $\underset{x\to n}{max}$，$\underset{x\to n}{min}$ 封闭，那么Δ对受限量词是封闭的.

[证] 因为$ct\bar{A}=NctA$，即$ct\bar{A}$是由 Nx 和 ctA 迭置而得，由于 Nx 和ctA 均属于Δ，且Δ对迭置封闭，故 $ct\bar{A}$ 也在Δ中. 因此Δ对否定词封闭.

因为 $ct(A\wedge B)=N^2(ctA+ctB)$，所以 $ct(A\wedge B)$可由 Nx,$x+y$, ctA, ctB 迭置而得. 因为这四个函数均在Δ中，且Δ对迭置封闭，故 $ct(A\wedge B)$也在Δ中，因此该Δ对合取词封闭.

因为 $A\vee B=\overline{\bar{A}\wedge\bar{B}}$, $A\supset B=\bar{A}\vee B=\overline{A\wedge\bar{B}}$, $A\equiv B=(A\wedge B)\vee(\bar{A}\wedge\bar{B})=\overline{\overline{A\wedge B}\wedge\overline{\bar{A}\wedge\bar{B}}}$, 又因为$\Delta$对否定词和合取词均封闭，所以$\Delta$对析取词蕴涵词等价词均封闭. 因此$\Delta$对命题联结词封闭.

因为$ct(\underset{x\to n}{\forall} Ax)=max(ctA(0),\cdots,ctA(n))$

$$=\underset{x\to n}{max}ctA(x)$$

又因为Δ对 $\underset{x\to n}{max}$ 封闭，$ctA(x)$ 在Δ中，而$ct(\underset{x\to n}{\forall} Ax)$是由 $ctA(x)$ 利用 $\underset{x\to n}{max}$ 而得，故 $ct(\underset{x\to n}{\forall} Ax)$在$\Delta$中. 因此$\Delta$对受限全称量词封闭.

同法可证 $ct(\underset{x\to n}{\exists} Ax)$也在$\Delta$中. 因此$\Delta$对受限存在量词封闭.

因此 Δ 对受限量词封闭.

本定理得证.

习　题

1. 求由 $x+y, yNx, rs(x,2)$ 出发, 利用迭置作 $lm(x,2)$ 的定义过程.

2. 求由 $x+y, yNx, rs(x,3)$ 出发, 利用迭置作 $dv(x,3), lm(x,3)$ 的定义过程.

3. 设 $A_1(x,y)=x+y$, $A_2(x,y)=xNy$, $A_3(x)=rs(x,2)$,

$$A_4(x)=\begin{cases}x-2, & \text{当 } x\geqslant 2 \text{ 且 } x \text{ 为偶数时}\\ g(x),\ (g \text{ 为任意一函数}), & \text{此外}\end{cases}$$

求由 A_1, A_2, A_3, A_4 出发, 利用迭置作 Dx 的定义过程.

4. 试由 $eq(x,y)$ 出发利用迭置作出函数 Nx.

5. 求证如果一函数集 Δ 对迭置封闭, 且函数 $x\dot{-}y$ 在 Δ 中, 则 Δ 对命题联结词封闭, 又如果 Δ 对 $\sum_{x\to n}$、$\prod_{x\to n}$ 封闭, 则 Δ 对受限量词封闭.

§3.4　五则函数

上面介绍了五种函数集, 现在首先讨论五则函数集.

我们已经知道, 五则函数是指由本原函数以及 $x+y$, $x\dot{-}y$, $x\cdot y$, $[y/x]$, $[\sqrt{x}]$ 出发, 利用迭置而得的函数. 因为

$$rs(x,y)=x\dot{-}y\cdot[x/y]$$

$$x\dot{-}y=(x\dot{-}y)+(y\dot{-}x)$$

$$x^a=x\cdot x\cdot\cdots\cdot x \qquad (\text{即 } a \text{ 个 } x \text{ 的乘积, 这里 } a \text{ 为定数})$$

$$Ex=x\dot{-}[\sqrt{x}]^2$$

$$Dx=x\dot{-}1$$

$$Nx=1\dot{-}x$$

$$xNy=x\cdot(1\dot{-}y)$$

$$max(x,y)=x+(y\dot{-}x)$$

$$min(x,y)=x\dot{-}(x\dot{-}y)$$

所以上面九个函数均是五则函数. 事实上除了 x' 以及第 n 个质数 P_n 等外, 前面两节中引入的函数几乎都是五则函数.

孙子定理(又称中国剩余定理) 如果各 b_i 两两互质, $c_i<b_i$ $(i=1, 2, \cdots, n)$, 则下列联立方程组

$$\begin{cases} rs(x, b_1)=c_1 \\ \cdots \\ rs(x, b_n)=c_n \end{cases}$$

在区间 $(0, b_1\cdot b_2\cdots\cdot b_n-1)$ 上必有一个也只有一个根, 其余的根则由这个根加 $b_1\cdot b_2\cdot\cdots b_n$ 的倍数而得.

[**证**] 本定理共有三个结论. 第一, 在 $(0, b_1\cdot b_2\cdot\cdots\cdot b_n-1)$ 上至少有一个根; 第二, 在 $(0, b_1\cdot b_2\cdot\cdots\cdot b_n-1)$ 上最多只有一个根; 第三, 其余的根与 $(0, b_1\cdot b_2\cdot\cdots\cdot b_n-1)$ 上的根的差恰为 $b_1\cdot b_2\cdot\cdots\cdot b_n$ 的倍数.

暂把上面的方程组写成下面的形式:

$$(1)\qquad rs(x, (b_1, \cdots, b_n))=(c_1, \cdots, c_n)$$

其中矢量 $(c_1, \cdots, c_n)$ 叫做 x 的(关于 $(b_1, \cdots, b_n)$ 的)剩余矢量.

设 u、v 是该方程组的任意两个根. 因此

$$rs(u, b_i)=c_i \text{ 且 } rs(v, b_i)=c_i,$$

故有

$$u=b_i\cdot r_i+c_i, \qquad v=b_i\cdot s_i+c_i$$

其中 r_i 和 s_i 分别为 u 和 v 除以 b_i 的商.

故有

$$u-v=b_i(r_i-s_i)$$

即 $u-v$ 为 b_i 的倍数 $(i=1, 2, \cdots, n)$. 因为各 b_i 两两互素, 故 $u-v$ 必为 $b_1\cdot b_2\cdot\cdots\cdot b_n$ 的倍数. 因此方程组(1)的任意两个根(如果存在的话)之差为 $b_1\cdot b_2\cdot\cdots\cdot b_n$ 的倍数. 当然, 如果方程组(1)在 $(0, b_1\cdot b_2\cdot\cdots\cdot b_n-1)$ 上有根的话, 那么该方程组的任何其余的根(如果有

的话)与 $(0, b_1, \cdot b_2 \cdot \cdots \cdot b_n - 1)$ 上的根之差必为 $b_1 \cdot b_2 \cdots b_n$ 的倍数，结论三得证.

设 u、v 是方程组(1)的任意两个不同的根. 由上知 $u-v$ 为 $b_1 \cdot b_2 \cdot \cdots \cdot b_n$ 的倍数. 故

$$u \doteq v = k \cdot b_1 \cdot b_2 \cdot \cdots \cdot b_n \geqslant b_1 \cdot b_2 \cdot \cdots \cdot b_n \ (k \geqslant 1)$$

但是在$(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$上的任意两个不同的数 x_1、x_2 必有 $x_1 \doteq x_2 < b_1 \cdot b_2 \cdot \cdots \cdot b_n$，故 x_1 和 x_2 不可能同时为方程组(1)的根. 因此在 $(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$上，方程组(1)最多只有一个根. 结论二得证.

因为在$(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$ 上的任意两个不同的 x_1 和 x_2 决不可能为同一方程组的根，因此在$(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$上不同的 x 必产生不同的剩余矢量. 由于这区间中共有 $b_1 \cdot b_2 \cdot \cdots \cdot b_n$ 个不同的 x，故这区间中的 x 共产生 $b_1 \cdot b_2 \cdot \cdots \cdot b_n$ 个不同的剩余矢量. 又由于 c_i 必小于 b_i，故不同的剩余矢量最多只有 $b_1 \cdot b_2 \cdot \cdots \cdot b_n$ 个. 由此可知，在$(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$上的 x 全体已给出了一切可能的剩余矢量，即在$(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$上，必有一个 x 使得方程组(1)成立，也即原方程组在$(0, b_1 \cdot b_2 \cdot \cdots \cdot b_n - 1)$上必有一个根. 结论一得证. 于是定理得证.

下面我们用例子给出一个求具体方程组之根的方法.

例：设有一数，依 8 数余 3，依 5 数余 4，依 11 数余 6. 求该数.

[解] 本题实为求下面方程组的根：

$$\begin{cases} rs(x, 8) = 3 \\ rs(x, 5) = 4 \\ rs(x, 11) = 6 \end{cases}$$

先将 5×11(即 $b_2 \times b_3$)的倍数依次用 8(即 b_1)除，直到余数为 3(即 c_1)时停止.

$$rs(55, 8) = 7$$

$$rs(2\times 55, 8)=6$$
$$rs(3\times 55, 8)=5$$
$$rs(4\times 55, 8)=4$$
$$rs(5\times 55, 8)=3$$

故 55 的倍数中被 8 除余 3 的数是 $5\times 55=275$.

再将 8×11(即 $b_1\times b_2$)的倍数依次用 5(即b_2)除，直到余数为 4(即c_2)时停止.

$$\begin{cases} rs(88, 5)=3 \\ rs(2\times 88, 5)=1 \\ rs(3\times 88, 5)=4 \end{cases}$$

故所找之数为 $3\times 88=264$.

再找 8×5(即 $b_1\times b_2$)的倍数中被 11(即 b_3)除余 6 (即 c_3) 的那个数.

$$rs(40, 11)=7$$
$$rs(2\times 40, 11)=3$$
$$rs(3\times 40, 11)=10$$
$$rs(4\times 40, 11)=6$$

故所找之数为 $4\times 40=160$.

所求的根必为该三数之和，再加减 $8\times 5\times 11$(即 $b_1\cdot b_2\cdot b_3$)的倍数，即

$$\begin{aligned} x &= 275+264+160\pm 8\times 5\times 11\times k_1 \\ &= 259+440k \end{aligned}$$

请读者自行证明本方法的正确性.

现在让我们来讨论三个重要的五则函数 $T_a x$, $R_a x$ 以及 $E_a x$. 它们可以看作 x^2, $[\sqrt{x}]$以及 Ex 的推广.

$T_a x$ 指 $x+[Sa\cdot x(x-1)/2]$. $T_a x$ 的值也叫做第 x 个 T_a 数.

$R_a x$ 指 x 以下(包括 x)非零 T_a 数的个数（今后约定“x 以下”

永包括 x 在内).

E_ax 指 $x\dot{-}T_aR_ax$. E_ax 的值称为 T_a 数剩余.

由定义可知, T_ax 为五则函数. 下面我们证明 R_ax 也是五则函数, 从而 E_ax 也是五则函数.

在证明 R_ax 为五则函数之前, 先对这三个函数的某些性质作些说明.

当 $a=0$ 时有

$$T_0x=x+[(0+1)\cdot x\cdot(x\dot{-}1)/2]$$

$$=x+x(x\dot{-}1)/2=\frac{x(x+1)}{2}$$

此数为代数中有名的三角数, 故 T_0 数即通常的三角数, 第 0 个 T_0 数即 $T_00=0$, 第 1 个 T_0 数即 $T_01=1$, 第 2 个 T_0 数即 $T_02=3$, 第 3 个 T_0 数即 $T_03=6$, $T_04=10$, $T_05=15$, $T_06=21$, ….

R_0x 即为 x 以下非零三角数的个数.

E_0x 即为 $x\dot{-}T_0R_0x$、称为三角数剩余.

由此可知, 因为 6 以下的非零 T_0 数有 1, 3, 6 三个, 所以

$$T_06=3,\ E_06=6-T_0R_06=6-T_03=6-6=0$$

因为 7 以下的非零 T_0 数也是 1, 3, 6 三个, 所以

$$T_07=3,\ E_07=7-T_0R_07=7-T_03=7-6=1$$

因为 8 以下的非零 T_0 数也是 1, 3, 6 三个, 所以

$$T_08=3,\ E_08=8-T_0R_08=8-T_03=8-6=2$$

因为 9 以下的非零 T_0 数也是 1, 3, 6 三个, 所以

$$T_09=3,\ E_09=9-T_0R_09=9-T_03=9-6=3$$

因为 10 以下的非零 T_0 数有四个, 即 1, 3, 6, 10, 所以

$$T_010=4,\ E_010=10-T_0R_010=10-T_04=10-10=0$$

再看当 $a=1$ 时有

$$T_1x=x+[(1+1)x(x\dot{-}1)/2]=x+x(x\dot{-}1)=x^2$$

此数为平方数.

R_1x 为 x 以下非零平方数的个数，即为$[\sqrt{x}]$.

$$E_1x=x\dot{-}T_1R_1x=x\dot{-}T_1[\sqrt{x}]=x\dot{-}[\sqrt{x}]^2=Ex$$

这三个是大家都熟悉的五则函数.

现在我们来证明 R_ax 是五则函数.

因为　　$T_a0=0,\ T_a1=1$

所以　　$R_a0=0$，且当 $x\neq0$ 时 $R_ax\geqslant1$

故可设　　$R_ax=t+1\qquad(x\neq0)$

于是　　$T_a(t+1)\leqslant x<T_a(t+2)$

即　　$$(t+1)+\frac{bt(t+1)}{2}\leqslant x<(t+2)+\frac{b(t+1)(t+2)}{2}$$

其中　　$b=Sa$

所以

$$8b\left((t+1)+\frac{bt(t+1)}{2}\right)+(b\dot{-}2)^2\leqslant8bx+(b\dot{-}2)^2$$

$$<8b\left((t+2)+\frac{b(t+1)(t+2)}{2}+(b\dot{-}2)^2\right.$$

即　　$$(2bt+b+2)^2\leqslant8bx+(b\dot{-}2)^2<(2bt+3b+2)^2$$

故　　$$2bt+b+2\leqslant[\sqrt{8bx+(b\dot{-}2)^2}]<2bt+3b+2$$

$$2bt+2b\leqslant[\sqrt{8bx+(b\dot{-}2)^2}]+a\dot{-}1<2bt+4b$$

$$t+1\leqslant[([\sqrt{8bx+(b\dot{-}2)^2}]+a\dot{-}1)/2b]<t+2$$

即当 $x\neq0$ 时

$$R_ax=t+1=\left[\frac{[\sqrt{8bx+(b\dot{-}2)^2}]+a\dot{-}1}{2b}\right]$$

$$=\left[\frac{[\sqrt{8xSa+(a\dot{-}1)^2}]+a\dot{-}1}{2Sa}\right]$$

不难看出，当 $x=0$ 时，上式左右两端均为 0，仍然成立. 故对一切 a 和一切 x，均有

$$R_a x=\left[\frac{[\sqrt{8xSa+(a\dot{-}1)^2}]+(a\dot{-}1)}{2a+2}\right]$$

由此式知道 $R_a x$ 为五则函数.

应该提请读者注意的是，虽然 $R_a x$ 可用这个式子表示，但由这个公式推导 $R_a x$ 和 $E_a x$ 的性质却非常困难. 通常还是根据定义来推导 $R_a x$ 和 $T_a x$ 的性质反而更清楚.

关于 $T_a x$, $R_a x$ 和 $E_a x$ 的主要性质有下列八个:

$T_a x\dot{-}T_a(x+y)=0$

$R_a x\dot{-}R_a(x+y)=0$

$R_a T_a x=x$

$T_a R_a x\dot{-}x=0$ （即 $T_a R_a x\leqslant x$）

$Sx\dot{-}T_a S R_a x=0$ （即 $x<T_a(R_a x+1)$）

$x=T_a R_a x+E_a x$

$T_a(x+y)=T_a x+T_a y+x\cdot y\cdot Sa$

$R_a(T_a(x+y)+Sa\cdot x)=x+y$

请读者自行由定义推导这些性质.

习　题

1. 试用五则函数表示下列函数:

1.1 x 以下的最大偶数;

1.2 x 以下的最大奇数;

1.3 x 以下的 y 的最大倍数;

1.4 x 以下的最大平方数.

2. 证明下列各式:

2.1 $1+[\sqrt{8z+1}]\geqslant[\sqrt{8z+8}]$;

2.2 $[([\sqrt{8z+1}]+1)/2]=[([\sqrt{8z+8}]+1)/2]$.

3. 求解下列问题:

3.1 今有物不知其数，三三数之余 2，五五数之余 3，七七数之余 2，问物几何？（此是《孙子算经》中的问题）

3.2 七数剩1，八数剩2，九数剩3，问本数．（此是<杨辉续古摘奇算经>中的问题）

4. 证明本节中给出的找剩余式方程组根的方法是正确的．

5. 验证下列各题：

5.1 本节末尾给出的关于 $T_a x$, $R_a x$ 和 $E_a x$ 的八个等式；

5.2 $T_a(x+1) > T_a x + x$；

5.3 当 Sx 非 T_a 数时，$E_a Sx = E_a x + 1$。

§3.5 配对函数

定义 如果二元函数 $pg(x, y)$ 与一元函数 $K(x)$, $L(x)$ 之间满足下列条件：

对于一切 x, y 均有 $Kpg(x, y) = x$, $Lpg(x, y) = y$，则 pg、K、L 叫做配对函数组，pg 叫做配对合函数，K 叫做配对左函数，L 叫做配对右函数，上面的条件叫做配对条件，$pg(x, y)$ 叫做二元矢量(x, y)的关于该配对函数组的编号．

由定义可知，对于任一配对函数组 pg, K, L 而言，任意一个二元矢量(x, y)都有一个编号，该编号就是 $pg(x, y)$ 之值；反之任给一个编号 a，均可用 K, L 给出其相应的二元矢量，该二元矢量就是(Ka, La)．这样便得

$$f(x, y) = f(Kpg(x, y),\ Lpg(x, y)) = f(K, L)pg(x, y)$$

换言之，二元函数 $f(x, y)$ 可表为一元函数 $f(K, L)$ 与二元配对合函数 $pg(x, y)$ 的迭置．因为配对函数组 pg、K 和 L 为事先给定的已知函数，因此对任意二元函数 $f(x, y)$ 的性质的研究，本质上可化归对一元函数 $f(K, L)$ 的研究．这就是说，在某种意义上，二元函数的研究可化归为一元函数的研究，进而多元函数的研究可化归为一元函数的研究．这就是配对函数的重要性所在．

必须指出，定义中并不要求配对合函数 $pg(x, y)$ 穷尽一切自然数，仅要求对于任意的 x, y，$pg(x, y)$ 均有定义．问题是能否用

常用函数作出配对函数组，回答是可以的，不但可以构造出配对函数组，而且可以造出无穷多的配对函数组，下面介绍一些常用的配对函数.

第一组

$$pg_a(x,y)=T_a(x+y)+x$$

$$K_a(x)=x\dot{-}T_aR_a(x)$$

$$L_a(x)=R_a(x)\dot{-}K_a(x)$$

第二组

$$\widetilde{pg}_a(x,y)=T_a(T_a(x+y)+y)+x$$

$$\widetilde{K}_a(x)=x\dot{-}T_aR_a(x)$$

$$\widetilde{L}_a(x)=\widetilde{K}_aR_a(x)$$

第三组

$$\overline{pg_a}(x,y)=T_a([(x+a)/Sa]+y)+x$$

$$\overline{K_a}(x)=x\dot{-}T_aR_a(x)$$

$$\overline{L_a}(x)=R_a(x)\dot{-}[(\overline{K_a}(x)+a)/Sa]$$

这三组配对函数均为五则函数，用得较多，故对它们给以特殊的符号. 下面也是配对函数，由于用得较少，故不给以特殊符号了.

第四组

$$pg(x,y)=2^x(2y+1)$$

$$K(x)=ep_0x$$

$$L(x)=[x/2^{K(x)+1}]$$

其中 ep_0x 指 x 的质因子分解式中 P_0(即第 0 个质数，也即 2)的幂指数. 一般 ep_nx 指 x 的质因子分解式中 P_n 的幂指数.

第五组

$$pg(x,y)=2^x\cdot 3^y$$

$$K(x)=ep_0x$$

$$L(x)=ep_1x$$

注意，这两组配对函数不是五则函数，而是初等函数.

为了讨论上面五组函数的性质，我们引进下面几个术语.

定义 如果 $pg(0,0)=0$，则说该配对函数组是从零开始的.

如果 $pg(x,y)$ 永不为 0，则说该配对函数组无零编号.

如果 $pg(x,y)$ 对 x,y 均是递增的，则说该配对函数组是递增的.

如果 $pg(x,y)$ 的值穷尽一切自然数，即每个自然数都是一编号，则说该配对函数是一一对应的.

如果当 $K(x+1)\neq 0$ 时必有 $K(x+1)=K(x)+1, L(x+1)=L(x)$，则说该配对函数组是平梯的.

由定义易证，如果 $pg(0,0)=0$，则 $K_0=0, L_0=0$. 还易证配对函数组一一对应当且仅当对任何 x 均有 $pg(Kx,Lx)=x$.

定理 上面的第一组配对函数是从零开始的递增的配对函数，第二组是从零开始的递增平梯配对函数，第三组是从零开始的递增的一一对应的配对函数组，第四组和第五组均是无零编号的递增配对函数组.

[证] 今就第二组证明，其余各组请自行证明.

因为 $\widetilde{pg}_a(0,0)=T_a(T_a(0+0)+0)+0=0$，所以该函数组是从零开始的.

因为当 $x>y$ 时 $T_ax>T_ay$，所以

$$\begin{aligned}\widetilde{pg}_a(x+1,y)&=T_a(T_a(x+1+y)+y)+x+1\\&>T_a(T_a(x+y)+y)+x\\&=\widetilde{pg}_a(x,y)\end{aligned}$$

故该函数组对 x 递增. 同理对 y 也递增. 因此该函数组是递增的.

因为 $\overleftarrow{K}_a(x)=E_ax$，而易知当 $E_a(x)\neq 0$ 时 x 必非 T_a 数，故当 $\overleftarrow{K}_a(x+1)\neq 0$ 时，$x+1$ 必非 T_a 数，又当 $x+1$ 非 T_a 数时，有

$$R_a(x+1)=R_ax$$

所以当 $\bar{K}_a(x+1)\neq 0$ 时有

$$R_a(x+1)=R_ax$$

故当 $\bar{K}_a(x+1)\neq 0$ 时有

$$\bar{K}_a(x+1)=x+1\dot{-}T_aR_a(x+1)=x+1\dot{-}T_aR_ax$$
$$=x\dot{-}T_aR_ax+1=\bar{K}_a(x)+1$$
$$\bar{L}_a(x+1)=\bar{K}_aR_a(x+1)=\bar{K}_aR_ax=\bar{L}_a(x)$$

故该函数组是平梯的.

上面已指出利用配对函数可以对二元矢量进行编号，从而使得二元函数可化归为一元函数. 进一步推广，可 以对 n 元矢量进行编号，从而使得 n 元函数可以化归成一元函数. 化归方法很多，这里我们介绍两种.

第一种方法.

令 $z=J_n(x_0, x_1, \cdots, x_n)$

$$=pg^nx_0x_1\cdots x_n$$
$$\hat{K}_0(x)=K^n(x)$$
$$\hat{K}_i(x)=LK^{n-i}(x)\qquad(1\leqslant i\leqslant n)$$

这时显然有

$$\hat{K}_0z=K^nz=K^nJ_n(x_0, x_1, \cdots, x_n)$$
$$=K^npg^nx_0x_1\cdots x_n$$
$$=x_0$$
$$\hat{K}_iz=LK^{n-i}z=LK^{n-i}J_n(x_0, x_1, \cdots, x_n)$$
$$=LK^{n-i}pg^nx_0x_1\cdots x_n$$
$$=Lpg^ix_0x_1\cdots x_i$$
$$=x_i$$

这样 $J_n(x_0, x_1, \cdots, x_n)$ 便是 $n+1$ 元矢量 $(x_0, x_1, \cdots, x_n)$ 的编号，$\hat{K}_iz$ 便是以 z 为其编号的 $n+1$ 元矢量的第 i 个分 量. 和二元矢

量一样，这里不要求编号 $J_n(x_0, x_1, \cdots, x_n)$ 穷尽一切自然数.

引理 对于任给的 $n+1$ 个数 $a_0, a_1, \cdots, a_n$ 必存在 c 和 d，使得

(1) $rs(c, 1+(i+1)d) = a_i \quad (0 \leqslant i \leqslant n)$

(2) $d \leqslant 2^{s^2}$

(3) $c \leqslant 2^{(n+2)^2+(n+1)s^2}$

其中 $s = \max\limits_{i \to n}(a_i + n)$.

[证] 令 $s = \max\limits_{i \to n}(a_i + n)$

$$d = s!$$

$$d_i = 1 + (i+1)d$$

显然有 $s \geqslant a_i$ 且 $s \geqslant n$

所以 $d_i \geqslant d+1 \geqslant s+1 > a_i$

又因为 $n! \leqslant n^n \leqslant 2^{n^2}$

所以 $d = s! \leqslant s^s \leqslant 2^{s^2}$

(2)得证.

当 $i \neq j$ 时(设 $i > j$)，d_i 与 d_j 的公因子必可除尽 $(i+1)d_j - (j+1)d_i$，因而必可除尽 $i-j$. 因为 $i-j$ 是 $n!$ 的因子，因而是 $s!$ 的因子，所以 d_i 与 d_j 的公因子必可除尽 $s!$(即 d). 但是 d_i 与 d 除 1 外无公因子，故 d_i 与 d_j($i \neq j$ 时)的公因子只有 1，所以 d_i 与 d_j 必互质. 由孙子定理知，方程组

$$\{rs(x, 1+(i+1)d) = a_i \quad (0 \leqslant i \leqslant n)$$

必有解.

令 c 为其最小根. 这样(1)得证.

由孙子定理知

$$c \leqslant d_0 \cdot d_1 \cdot \cdots \cdot d_n$$

$$\leqslant 2d \cdot 3d \cdot \cdots \cdot (n+2)d$$

$$\leqslant (n+2)!d^{n+1}$$
$$\leqslant 2^{(n+2)^2}\cdot(s!)^{n+1}$$
$$\leqslant 2^{(n+2)^2}\cdot(2^{s^2})^{n+1}$$
$$=2^{(n+2)^2+(n+1)s^2}$$

(3)得证.

定理 任给配对左、右函数 K, L 以及 $n+1$ 个数 $a_0, a_1, \cdots, a_n$（或者任给一函数 $a(x)$），恒可找到一数 w，使得

$$rs(Kw, 1+(i+1)Lw)=a_i \quad (\text{或者}=a(i)) \quad (0\leqslant i\leqslant n)$$

又如果相应于 K, L 的合函数是递增的，那么还可使得

$$w\leqslant pg(2^{(n+2)^2+(n+1)s^2}, 2^{s^2})$$

其中 $s=\max\limits_{i\to n}(a_i+n)$.

[证] 由上引理先找出 c, d，使得 $w=pg(c, d)$. 显然 w 为所要求的数.

今后我们把方程组

$$rs(Kw, 1+(i+1)Lw)=a_i \qquad (0\leqslant i\leqslant n)$$

的最小根记为 $\underset{i\to n}{seq}\ a_i$

或记为 $seq[a_0, a_1, \cdots, a_n]$

把函数 $rs(Kw, 1+(i+1)Lw)$ 记为 $tm(i, w)$

或记为 $tm_i(w)$

由本定理可知，任给 $n+1$ 个数 $a_0, a_1, \cdots, a_n$，必存在 $\underset{i\to n}{seq}\ a_i$，且永有

$$tm(i, \underset{i\to n}{seq}\ a_i)=tm_i(\underset{i\to n}{seq}\ a_i)$$
$$=a_i \qquad (0\leqslant i\leqslant n)$$

下面给出第二种对多元矢量编号的方法.

令 $Z=J_n(x_0, x_1, \cdots, x_n)=\underset{i\to n}{seq}\ x_i$

$$K_i z=tm(i, z) \qquad (0\leqslant i\leqslant n)$$

这样 $J_n(x_0, x_1, \cdots, x_n)$ 就是 $n+1$ 元矢量 $(x_0, x_1, \cdots, x_n)$ 的编号，

$\hat{K}_i z$ 就是以 z 为其编号的 $n+1$ 元矢量的第 i 个分量.

和二元矢量一样，从某种意义上说，利用函数组 $J_n, \hat{K}_0, \cdots, \hat{K}_n$ 便可把多元函数化归为一元函数. 今后，我们将要利用这个性质，把多元函数简化成一元函数来讨论.

习　题

1. 试证如果一配对函数是一一对应的，则该配对函数不可能是平梯的.

2. 设 pg, K, L 是一组配对函数. 求证：pg^2xyy 和 $pgypgxy$ 均是配对合函数，即求它们相应的配对左右函数.

3. 试用 pg_n, $\widetilde{pg_n}$, $\overline{pg_n}$ 分别写出 (i, j) 的编号 $(0 \leqslant i, j \leqslant 4)$.

4. 选取 pg_1, K_1, L_1 为 J_n 所用的配对函数组，试用上面给的两种方法算出 $J_3(2, 3, 2)$ 之值.

5. 试证下面的函数组可对三元矢量进行编号：

$$J_3(x, y, z) = 2^x \cdot 3^y \cdot (6z+1)$$

$$\hat{K}_1 x = ep_0 x$$

$$\hat{K}_2 x = ep_1 x$$

$$\hat{K}_3(x) = [[x/(2^{ep_1(x)} \cdot 3^{ep_2(x)})]/6]$$

§3.6　初等函数

我们已经知道，初等函数就是由本原函数和 $x \dot{-} y$ 出发，利用迭置、迭加、迭乘而得的函数. 所有初等函数组成的集合称为初等函数集.

3.6.1　若干常见的初等函数

1. $x \cdot y$

$$x \cdot y = \sum_{i \to x} y \dot{-} y$$

2. $x + y$

$$x + y = Sx \cdot Sy \dot{-} S(x \cdot y)$$

3. Nx 与 N^2x

$$Nx=\prod_{i\to x}(i\dot{-}1)$$

$$N^2x=1\dot{-}Nx$$

4. $x\dot{-}y$

$$x\dot{-}y=(x\dot{-}y)\cdot N^2(y\dot{-}x+x\dot{-}y)$$

5. $\left[\frac{x}{y}\right]$

$$\left[\frac{x}{y}\right]=N^2y\cdot\sum_{i\to x}N(y\cdot(i+1)\dot{-}x)$$

6. x'

$$x'=\left[\prod_{i\to y}x/x\right]+NxNy$$

7. $[\sqrt{x}]$

$$[\sqrt{x}]=\sum_{i\to x}N(i^2\dot{-}x)\dot{-}1$$

为了下面讨论方便，我们先引入一个关于一个函数集的控制函数的概念，并介绍它的性质.

3.6.2 函数集的控制函数

定义 设有一个函数集Δ，又有一个二元函数 $g(x,y)$. 如果对于Δ中的任何一个函数 $f(x_1,\cdots,x_n)$，恒有一数 h，使得对于任何 $x_1,\cdots,x_n$，均有

$$f(x_1,\cdots,x_n)<g(h,max(x_1,\cdots,x_n))$$

则说二元函数 $g(x,y)$是函数集Δ的控制函数.

定理1 如果函数 $g(x,y)$ 是某个函数集Δ的控制函数，则该控制函数 g 不在Δ中.

[**证**] 用反证法证之. 设 g 为Δ的控制函数，且 g 在Δ中，则由控制函数的定义，既然 $g(x,y)$是Δ中某个函数，故应有数 h，使

得对任何 x, y, 有

$$g(x, y)<g(h, max(x, y)).$$

现取 x, y 都为 h, 则得

$$g(h, h)<g(h, max, (h, h))=g(h, h).$$

这是一个矛盾. 所以 g 必不在 Δ 中.

定理 2 五则函数集的控制函数为

$$g(x, y)=(y+2)^x.$$

[证] (i) 因 $O(x)=0<1=(x+2)^0$, 故对于 $O(x)$, $h=0$.

(ii) 因 $Sx<x+2=(x+2)^1$, 故对于 Sx, $h=1$.

(iii) 因 $I_{mn}(x_1, \cdots, x_m)=x_n<x_n+1<max(x_1, \cdots, x_m)+2$, 故对于 I_{mn}, $h=1$.

(iv) 因 $x+y\leqslant 2max(x, y)<(max(x, y)+2)^2$, 故对于 $x+y$, $h=2$.

(v) 因 $x\dot{-}y\leqslant x$, 故对于 $x\dot{-}y$, $h=1$.

(vi) 因 $x\cdot y\leqslant(max(x, y))^2<(max(x\cdot y)+2)^2$, 故对于 $x\cdot y$, $h=2$.

(vii) 因 $\left[\frac{x}{y}\right]\leqslant x$, $[\sqrt{x}]\leqslant x<x+2$, 故对于 $\left[\frac{x}{y}\right]$ 可取 $h=1$, 对于 $[\sqrt{x}]$ 也可取 $h=1$.

(viii) 设 $f(x_1, \cdots, x_n)=A(B_1, \cdots, B_m)(x_1, \cdots, x_n)$, 而对于 A, $B_1, \cdots, B_m$, 均有相应的 $h_0, h_1, \cdots, h_m$, 使得

$$A(y_1, \cdots, y_m)<(max(y_1, \cdots, y_m)+2)^{h_0}$$

$$B_i(x_1, \cdots, x_n)<(max(x_1, \cdots, x_n)+2)^{h_i} \quad (1\leqslant i\leqslant m)$$

则这时

$$f(x_1, \cdots, x_n)=A(B_1, \cdots, B_m)(x_1, \cdots, x_n)<(max(B_1, \cdots, B_m)+2)^{h_0}$$

又

$$max(B_1, \cdots, B_m)\leqslant max((max(x_1, \cdots, x_n)+2)^{h_1}, \cdots, (max$$

$$(x_1, \cdots, x_n)+2)^{h_m})$$

$$=(max(x_1, \cdots, x_n)+2)^{max(h_1, \cdots, h_m)}$$

故 $f(x_1, \cdots, x_n)<[(max(x_1, \cdots, x_n)+2)^{max(h_1, \cdots, h_m)}+2]^{h_0}$

$$\leqslant[2((max(x_1, \cdots, x_n)+2)^{max(h_1, \cdots, h_m, 2)})]^{h_0}$$

$$\leqslant[(max(x_1, \cdots, x_n)+2)^{max(h_1, \cdots, h_m, 2)+1}]^{h_0}$$

$$=(max(x_1, \cdots x_n)+2)^{h_0 \cdot (max(h_1, \cdots, h_m, 2)+1)}$$

于是由 $A, B_1, \cdots, B_m$ 经(m, n)迭置而得的函数 f 亦可有相应的 $h=h_0 \cdot (1+max(h_1, \cdots, h_m, 2))$.

按照我们在引论中所讨论的一般意义下的数学归纳法，本定理得证.

3.6.3 初等函数与五则函数的关系

定理 3 五则函数集是初等函数集的真子集.

[证] 由 3.6.1 中讨论知，所有五则函数 $x+y$，$x \dot{-} y$，$x \cdot y$，$\left[\frac{x}{y}\right]$与$[\sqrt{x}]$都是初等函数，又因$(y+2)^x$是五则函数集的控制函数，故它不是五则函数；但已知 y^x 为初等函数，故由它迭置而得的 $(y+2)^x$亦是初等函数，从而定理得证.

3.6.4 初等函数与受限幂状式的关系

定理 4 受限幂状算子均可在初等函数集中表示之.

[证] (i) 受限幂状算子 $\underset{i \to n}{rtn}$ 可在初等函数集中表示之. 这是因为

$$\underset{i \to n}{rtn}\, f(t)=\sum_{i \to n} Nf(t)$$

(ii) 受限幂状算子 $\underset{i \to n}{rti}$ 可在初等函数集中表示之. 这是因为

$$\underset{i \to n}{rti}\, f(t)=\sum_{k \to n} \prod_{i \to k} N^2 f(t) \dot{-} \prod_{i \to n} N^2 f(t)$$

(iii) 受限幂状算子 $\underset{t\to n}{rtu}$ 可在初等函数集中表示之. 这是因为

$$\underset{t\to n}{rtu}\, f(t)=\underset{t\to n}{rti}f(t)\cdot N(\underset{t\to n}{rtn}f(t)\dot{-}1)+n\cdot N^2(\underset{t\to n}{rtn}f(t)\dot{-}1)$$

(iv) 受限幂状算子 $\underset{t\to n}{rta}$ 也可在初等函数集中表示之. 这是因为

$$\underset{t\to n}{rta}f(t)=n\dot{-}\underset{t\to n}{rti}f(n\dot{-}t)\cdot N\prod_{t\to n}N^2f(t)$$

由(i)～(iv)知, 各种受限幂状算子均可在初等函数集中表示之. 故定理得证.

推论 由初等函数出发利用迭置和受限幂状算子而得的函数, 仍为初等函数.

3.6.5. 另外一些有用的初等函数

1. "x 因子的个数"(记为 $\mathscr{F}(x)$).

$$\mathscr{F}(x)=\underset{t\to x}{rtn}\, rs(x,t)=\sum_{t\to x}Nrs(x,t)$$

2. "x 为质数"的特征函数(记为 $\mathscr{P}(x)$).

$\mathscr{P}(x)$可以随质数定义的描述方式的不同而有各种不同的表达式, 下面我们给出四种不同的定义以及与它们对应的特征函数, 当然使用时只要用其中的一种就够了.

2.1 "x 的因子个数为 2".

$$\mathscr{P}(x)=N^2\left(2\dot{-}\sum_{t\to x}Nrs(x,t)\right)$$

2.2 "$x\geqslant 2$ 并且由 a 除尽 x 可推出 $a=1$ 或 $a=x$".

$$\mathscr{P}(x)=N^2\left((2\dot{-}x)+\sum_{a\to x}(Nrs(x,a)\cdot(1\dot{-}a)\cdot(x\dot{-}a))\right)$$

2.3 $x\geqslant 2\wedge \underset{a\to x}{\forall}\underset{b\to x}{\forall}((a\cdot b=x)\supset(a=1\vee b=1))$

$$\mathscr{P}(x)=N^2\left((2\dot{-}x)+\sum_{a\to x}\sum_{b\to x}(N(x\dot{-}a\cdot b)\cdot(1\dot{-}a)\cdot(1\dot{-}b))\right)$$

2.4 "$x\geqslant 2\wedge x$ 除不尽$(x\dot{-}1)!^{2n}$"

$\mathscr{P}(x)=N^2((2\dot{-}x)+Nrs((x\dot{-}1)!^2,x))$

3. "x 以下质数的个数"(记为 $\mathscr{J}(x)$).

$$\mathscr{J}(x)=\underset{t\to x}{rtn}\mathscr{P}(t)=\sum_{t\to x}N\mathscr{P}(t)$$

4. "第 n 个质数"(记为 P_n).

P_n 可如下定义:"使得 'x 以下质数的个数$=n+1$' 成立的最小 x",故得

$$P_n=\underset{x}{rti}\left((n+1)\dot{-}\sum_{t\to x}N\mathscr{P}(t)\right)$$

但是这里使用不受限摹状算子,而不受限摹状算子是不能在初等函数集中表示的,因此必须对它加限.在§3.0中曾证明:$P_n\leqslant 2^{2^n}$,现暂把 2^{2^n} 记为 a,这样便得

$$P_n=\underset{x\to a}{rti}\left((n+1)\dot{-}\sum_{t\to x}N\mathscr{P}(t)\right)=\sum_{k\to a}\prod_{x\to k}N^2\left((n+1)\dot{-}\sum_{t\to x}N\mathscr{P}(t)\right)$$

于是我们便得到了一个非常精确的关于 P_n 的公式,

5. "x 的质因子分解式中第 a 个质数的方幂"(记为 ep_ax 或者 $ep(a,x)$).ep_ax 可如下定义:"在$(0,x)$中,满足 $rs(x,P_a^{t+1})\neq 0$的最小 t 根".故得

$$ep_ax=ep(a,x)=\underset{t\to x}{rti}\ Nrs\ (x,P_a^{t+1})=\sum_{k\to x}\prod_{t\to k}Nrs(x,P_a^{t+1})$$

这里约定 $ep(a,0)=0$.

6. "x 的最大质因子的足码"(记为 $H(x)$). $H(x)$ 可如下定义:"在$(0, x\dot{-}1)$中,满足 $rs(x, P_t)=0$ 的最大 t 根",故得

$$H(x)=\underset{t\to Dx}{rta}\, rs(x, P_t)=Dx\dot{-}\sum_{k\to Dx}\prod_{t\to k} rs(x, P_{Dx\dot{-}t})\cdot N\prod_{t\to Dx} N^2 rs(x, P_t)$$

注意 $H(0)=H(1)=0$.

7. "x 的质因子分解式"可如下表示

$$x=\prod_{t\to x} P_t^{\,\beta(t,\,x)} \qquad (x\neq 0)$$

3.6.6 与加限原始递归式的关系

定义 设有
$$\begin{cases} f(n, u, 0)=A(u) \\ f(n, u, x+1)=B(u, min(n, x), \\ min(n, f(n, u, x))) \end{cases}$$
则说新函数 $f(n, u, x)$ 是由 A, B 利用加限原始递归式而得. 其中 x 为递归变元, u 称为参变元(可以有多个), n 称为限制变元.

这个式子与原始递归式基本相同, 仅当用到下列三种值之一时才作相应的修改:

① 当$(t>n)\wedge(y\leqslant n)$时把 $B(u, t, y)$ 换为 $B(u, n, y)$;

② 当$(t>n)\wedge(x\leqslant n)$时, 把 $B(u, x, t)$ 换为 $B(u, x, n)$;

③ 当 $t_1, t_2>n$ 时, 把 $B(u, t_1, t_2)$ 换为 $B(u, n, n)$.

下面我们证明加限原始递归式可在初等函数集中表示.

引理 设
$$\begin{cases} f(n, u, 0)=A(u) \\ f(n, u, x+1)=B(u, min(n, x), min(n, f(n, u, x))) \end{cases}$$
则 $f(n, u, x)\leqslant A(u)+\sum_{t_1\to n}\sum_{t_2\to n} B(u, t_1, t_2)$

[证] 因为 $f(n, u, x+1)=B(u, min(n, x), min(n, f(n, u, x)))$

$$\leqslant \max_{t_1 \to n} \max_{t_2 \to n} B(u, t_1, t_2) \leqslant \sum_{t_1 \to n} \sum_{t_2 \to n} B(u, t_1, t_2)$$

所以 $f(n, u, x) \leqslant max(A(u), \sum_{t_1 \to n} \sum_{t_2 \to n} B(u, t_1, t_2))$

$$\leqslant A(u) + \sum_{t_1 \to n} \sum_{t_2 \to n} B(u, t_1, t_2)$$

定理 5 由初等函数出发，利用加限原始递归式而得的函数必为初等函数.

[**证**] 设 $A(u)$, $B(u, x, y)$ 均为初等函数，$f(n, u, x)$ 为由 A, B 利用加限原始递归式而得的函数，根据引理有：

$$f(n, u, x) \leqslant A(u) + \sum_{t_1 \to n} \sum_{t_2 \to n} B(u, t_1, t_2)$$

(记为 $c(u, n)$)(注意 $c(u, n)$ 是初等函数)，现定义 $w(n, u, x)$ 如下：

$$w(n, u, x) = \underset{y \to G}{rti}((A(u) \dot{-} ep_0 y) + \sum_{t \to x} (ep_{t+1} y \dot{-} B(u, min(n, t), min(n, ep_t y))))$$

其中 G 表示 $P_{x+1}^{(x+2) \cdot c(u, n)}$. 显然 G 仍为初等函数，而且 $w(n, u, x)$ 也为初等函数. 容易验证

$$w(n, u, x) = \prod_{t \to x+1} P_t^{f(n, u, t)}$$

因此得 $$f(n, u, x) = ep_x w(n, u, x)$$

所以 $f(n, u, x)$ 为初等函数. 定理得证.

3.6.7 初等函数集的控制函数

定理 6 $g(n, a) = \underset{t \to (1, n)}{itr} (a+2)^t$ 是初等函数集的控制函数，因而不在初等函数集之中.

易见，$g(0, a) = 1$, $g(1, a) = (a+2)$, $g(2, a) = (a+2)^{(a+2)}$, $g(3, a) = (a+2)^{(a+2), (a+2)}$…于是 $g(n, a)$ 对于 a 是递增函数.

设 $f(x_1, x_2, \cdots, x_n)$ 是任一初等函数，现就 f 的组成过程施行

归纳法证明，有 h 使得

$$f(x_1, \cdots, x_n) < g(h, u), \quad (u = max(x_1, \cdots, x_n))$$

奠基 $O(x_n) \leqslant I_{mn}(x_1, \cdots, x_m) = x_n < S(x_n) < (u+2)$

$$= g(1, u) \qquad (u = max(x_1, \cdots, x_n))$$

$$x \dot{-} y \leqslant max(x, y) < (u+2) = g(1, u) \qquad (u = max(x, y))$$

因此有 $h = 1$ 使得初等函数集的所有开始函数均 $< g(h, u)$.

归纳 对迭置，迭加，迭乘三种情况分别证明之.

i) 对于迭置，设 $f(x_1, \cdots, x_n) = A(B, \cdots, B_m)(x_1, \cdots, x_n)$，依归纳假设有 $h_0, h_1, \cdots, h_m$（均 $\geqslant 1$），使得

$$A(x_1, \cdots, x_m) < g(h_0, u) \qquad (u = max(x_1, \cdots, x_n))$$

$$B_i(x_1, \cdots, x_n) < g(h_i, u) \quad (u = max(x_1, \cdots, x_n)) \quad (1 \leqslant i \leqslant m)$$

故 $f(x_1, \cdots, x_n) = A(B_1, \cdots, B_m)(x_1, \cdots, x_n)$

$$< g(h_0, \max_{i \to (1, m)} B_i(x_1, \cdots, x_n)) \qquad \text{(归纳假设)}$$

$$< g(h_0, \max_{i \to (1, m)} g(h_i, u)) \quad (u = max(x_1, \cdots, x_n))$$

（归纳假设和 g 对 a 递增）

$$= g(h_0, g(k, u)), \quad (k = max(h_1, \cdots, h_m))$$

$$\leqslant g(k + 2h_0, u) \qquad \text{(可用归纳法证明)}$$

ii) 对于迭加设 $f(x_1, \cdots, x_n, y) = \sum_{i \to y} A(x_1, \cdots, x_n, t)$，依归纳假设有 h_0，使得

$$A(x_1, \cdots, x_n, t) < g(h_0, u) \qquad (u = max(x_1, \cdots, x_n, t))$$

故 $f(x_1, \cdots, x_n, y) = \sum_{t \to y} A(x_1, \cdots, x_n, t)$

$$< \sum_{t \to y} g(h_0, max(x_1, \cdots, x_n, t)) \quad \text{(归纳假设)}$$

$$\leqslant \sum_{t \to y} g(h_0, max(x_1, \cdots, x_n, y)) = (y+1) g(h_0, max(x_1, \cdots, x_n, y))$$

$<g(h_0+1, max(x_1, \cdots, x_n, y))$　　(请自行证明)

iii) 对迭乘，设 $f(x_1, \cdots, x_n, y)=\prod_{t\to y} A(x_1, \cdots, x_n, t)$　依归纳假设有 h_0，使得

$$A(x_1, \cdots, x_n, t)<g(h_0, max(x_1, \cdots, x_n, t))$$

故 $f(x_1, \cdots, x_n, y)=\prod_{t\to y} A(x_1, \cdots, x_n, t)$

$<\prod_{t\to y} g(h_0, max(x_1, \cdots, x_n, t))$　　(归纳假设)

$\leqslant\prod_{t\to y} g(h_0, max(x_1, \cdots, x_n, y))=(g(h_0, u))^{(y+1)}$

$(u=max(x_1, \cdots, x_n, y))$

$<g(h_0, u)^{g(h_0, u)}=(u+2)^{g(h_0-1, u)g(h_0, u)}<(u+2)^{g(h_0, u)^2}$

$\leqslant(u+2)^{g(h_0+1, u)}=g(h_0+2, u)$

因此，利用迭置，迭加，迭乘所得的新函数 $f(x_1, \cdots, x_n)$ 仍有 h 使得

$$f(x_1, \cdots, x_n)<g(h, max(x_1, \cdots, x_n))$$

定理得证.

习　题

1. 将下列数论函数表成初等函数：

1.1　$\sigma_1(x)$：x 的约数之和；

1.2　$v(x)$：x 的质因子的个数(相同的重复计算)；

1.3　$\nu(x)$：大于 x 的第一个质数；

1.4　$\xi(x)$：

$$\xi(x)=\begin{cases}1, & \text{当 } x=1 \text{ 时},\\ 0, & \text{当 } x\neq 1 \text{ 且为平方数的倍数时},\\ a^k, & \text{此外，其中 } k \text{ 为 } x \text{ 的质因子个数};\end{cases}$$

1.5　$dv(a, b)$：a, b 的最大公约数(约定 $a, b=0$ 时 $dv(a, b)=a+b$)；

1.6　$lm(a, b)$：a, b 的最小公倍数(约定 $a\cdot b=0$ 时 $lm(a, b)=0$).

2. 求下列谓词的特征函数：

2.1 x 为完全数(即 x 的所有因子之和为 x 的两倍，例如 6 为完全数)。

2.2 m, n 为亲和数(即 m 及 n 的因子之和均为 $m+n$，例如 6 与 6 就是亲和数)。

3. 试用简单的语言把下列函数叙述出来：

3.1 $\sum\limits_{t\to n} rs(t+1,2)$；

3.2 $\sum\limits_{t\to n} t^2$；

3.3 $\sum\limits_{t\to n} NE_a(t+1)$；

3.4 $\prod\limits_{t\to n} N^2 f(t)$；

3.5 $\sum\limits_{k\to n}\prod\limits_{t\to k} N^2 f(t)$。

4. 求证下列各题：

4.1 迭大算子和迭小算子均可在初等函数集中表示之；

4.2 $$\sum_{t\to n} nf(t)\leqslant \sum_{t\to(n+1)} (n+1)f(t),$$

$$\sum_{t\to n} n\cdot f(t)+f(n)\leqslant \sum_{t\to(n+1)} (n+1)f(t)+f(n+1);$$

4.3 任给 a, b 恒有 x, y 使得 $a\cdot x\dot{-}b\cdot y=dv(a,b)$，试将 x, y 表示为 a, b 的初等函数；

4.4 如果 $$g(x,n)=\mathop{itr}_{t\to(1,n)}(x+2)^t$$

则有

i) $(x+1)g(x,n)\leqslant g(x,n+1)$；

ii) $g(g(x,n),m)\leqslant g(x,n+2m)$。

§3.7 原始递归函数

我们知道原始递归式的标准形式是

$$\begin{cases} f(u_1,\cdots,u_r,0)=A(u_1,\cdots,u_r) \\ f(u_1,\cdots,u_r,n+1)=B(u_1,\cdots,u_r,n,f(u_1,\cdots,u_r,n)). \end{cases}$$

$f(u_1,\cdots,u_r,x))$ 称为由已知函数 $A(u_1,\cdots,u_r), B(u_1,\cdots, u_r,$

x, y)出发，利用原始递归式而得. f 中的 x 称为递归变元，诸 u_i 称为参数. 这里参数有 r 个，但利用配对函数恒可化为一个. 因此，今后我们永把下列只含一个参数的式子作为原始递归式的标准式：

$$\begin{cases} f(u, 0) = A(u) \\ f(u, n+1) = B(u, n, f(u, n)) \end{cases}$$

前面曾定义，凡由本原函数出发，经过有限次迭置与原始递归式所作成的函数称为原始递归函数，所有原始递归函数组成的集合称为原始递归函数集.

本节讨论三个方面的问题：

1. 原始递归函数与初等函数的关系.

2. 标准原始递归式与其它各种原始递归式的关系.

3. 原始递归函数集的控制函数.

3.7.1 原始递归函数与初等函数的关系

定理 1 初等函数集是原始递归函数集的真子集.

[证] (i) 所有初等函数均是原始递归函数，也即初等函数集是原始递归函数集的子集.

这可以根据初等函数的定义施行证明：

首先本原函数是共有的，其次，$x \dot{-} y$ 可如下定义：

$$\begin{cases} D0 = 0 \\ D(n+1) = n \end{cases}$$

$$\begin{cases} x + 0 = x \\ x + (n+1) = S(x+n) \end{cases}$$

$$\begin{cases} x \dot{-} 0 = x \\ x \dot{-} (n+1) = D(x \dot{-} n) \end{cases}$$

$$\begin{cases} x \cdot 0 = 0 \\ x \cdot (n+1) = x \cdot n + x \end{cases}$$

故 $Dx, x+y, x\dot{-}y, x\cdot y$ 均是原始递归函数.

因为 $x\ddot{-}y=(x\dot{-}y)+(y\dot{-}x)$，即 $x\ddot{-}y$ 可由原始递归函数利用迭置而得，所以 $x\ddot{-}y$ 是原始递归函数，这样初等函数集的开始函数都是原始递归函数.

再次，迭置是共有的. 最后，迭加算子迭乘算子可如下定义：

$$\begin{cases}\sum\limits_{t\to 0} f(t)=f(0)\\ \sum\limits_{t\to(n+1)} f(t)=f(n+1)+\sum\limits_{t\to n} f(t)\end{cases}$$

$$\begin{cases}\prod\limits_{t\to 0} f(t)=f(0)\\ \prod\limits_{t\to(n+1)} f(t)=f(n+1)\cdot\prod\limits_{t\to n} f(t)\end{cases}$$

故迭加，迭乘算子可以由加法，乘法以及原始递归式表出.■ 于是(i)得证.

(ii) 有些原始递归函数不是初等函数.

上节定理 5 已证明了 $itr_{t\to(1,n)}(a+2)^t$ 是初等函数集的控制函数，因而不是初等函数，但是显然为原始递归函数，于是(ii)得证.

由(i)和(ii)便知，本定理成立.

3.7.2 原始递归式与原始复迭式的关系

设 $$f(x)=B^x(0)=itr_{t\to(0,x)} B(t)$$

即

$$\begin{cases}f(0)=0\\ f(n+1)=B(f(n))\end{cases}$$

显然它也是原始复迭式的一种特殊情形，当然更是原始递归式的特例，称为弱原始复迭式.

下面将证明原始递归式的作用(在允许增加开始函数的条件下)可

由弱原始复迭式来代替.

定理 2 只要增加两个开始函数 $x+y, Ex$ 便可以利用迭置和弱原始复迭式，把原始递归式表示出来.

[证] (i) 由本原函数以及 $x+y, Ex$，利用迭置和弱原始复迭式可定义出下列函数

① $x+y$ (开始函数)

② Ex (开始函数)

③ $N^2x=\underset{t\to(0,x)}{itr}1$ 即

$$\begin{cases}N^20=0\\N^2(x+1)=1\end{cases}$$ (由本原函数和弱复迭式而得)

④ $Nx=N^2E(3+N^2x)$ (由本原函数，$x+y$，Ex，N^2x 利用迭置而得)

⑤ $G(x)=\underset{t\to(0,x)}{itr}(t+2NE(t+4)+1)$

即

$$\begin{cases}G(0)=0\\G(x+1)=G(x)+2NE(G(x)+4)+1\end{cases}$$

注意，$G(x)=x+2[\sqrt{x}]$

⑥ $x^2=\underset{t\to(0,x)}{itr}S(G(t))$

即

$$\begin{cases}0^2=0\\(x+1)^2=G(x^2)+1\end{cases}$$

⑦ $x\dot{\odot}y=E((x+y)^2+3x+y+1)$

注意

$$x\dot{\odot}y=\begin{cases}x\dot{-}y, & \text{当}x\geqslant y\text{时}\\3x+y+1, & \text{当}x<y\text{时}\end{cases}$$

⑧ $rs(x,3)=\underset{t\to(0,x)}{itr}(Nt+2N(t\dot{\odot}1))$

即

$$\begin{cases} rs(0,3)=0 \\ rs(x+1,3)=Nrs(x,3)+2N(rs(x,3)\dot{-}1) \end{cases}$$

⑨ $H_3(x)=\underset{t\to(0,x)}{itr}(t+1+rs(s,3))$

即

$$\begin{cases} H_3(0)=0 \\ H_3(x+1)=H_3(x)+1+rs(H_3(x),3) \end{cases}$$

注意，$H_3(x)=x+[x/2]$.

⑩ $[x/2]=H_3(x)\dot{-}x$

⑪ $x\cdot y=[(((x+y)^2\dot{-}x^2)\dot{-}y^2)/2]$

⑫ $xNy=x\cdot Ny$

⑬ $[\sqrt{x}]=[(G(x)\dot{-}x)/2]$

(ii) 利用上面定义的 $x+y, Ex, x^2, [\sqrt{x}]$可作出下列配对函数组

$$\widetilde{pg}(x,y)=((x+y)^2+y)^2+x$$

$$\tilde{K}x=Ex, \qquad \tilde{L}x=E([\sqrt{x}])$$

§5 中曾证明，该配对函数组是从 0 开始的，而且是平梯的.

(iii) 利用这个配对函数组以及(i)中定义的 $Nx, x\cdot y$ 等函数可以把原始递归式化为原始复迭式，即任一利用原始递归式定义的函数，均可改用原始复迭式来定义.

设 $f(u,x)$是由 A、B利用原始递归式定义的函数，即

$$\begin{cases} f(u,0)=A(u) \\ f(u,x+1)=B(u,x,f(u,x)) \end{cases}$$

现作下列函数

$$\begin{cases} g(v,u,0)=v \\ g(v,u,x+1)=B(u,x,g(v,u,x)) \end{cases}$$

则有

$$f(u,x)=g(A(u),u,x)\qquad\text{(可用归纳法验证之)}$$

再作下列函数

$$h(u)=\widetilde{pg}(u,g(\tilde{L}^2u,\tilde{K}\tilde{L}u,\tilde{K}u))$$

易见，$h(u)$具有下列三种性质:

(1) $\tilde{K}h(u)=u,\quad \tilde{L}h(u)=g(\tilde{L}^2u,\tilde{K}\tilde{L}u,\tilde{K}u)$

(2) $h(0)=\widetilde{pg}(0,g(\tilde{L}^2(0),\tilde{K}\tilde{L}(0),\tilde{K}(0)))$

$$=\widetilde{pg}(0,g(0,0,0))=\widetilde{pg}(0,0)=0$$

$$h(u+1)=\widetilde{pg}(u+1,g(\tilde{L}^2(u+1),\tilde{K}\tilde{L}(u+1),\tilde{K}(u+1)))$$

$(u+1)=0$ 时

$$h(u+1)=\widetilde{pg}(u+1,g(\tilde{L}^2(u+1),\tilde{K}\tilde{L}(u+1),0))$$

$$=\widetilde{pg}((u+1),\ \tilde{L}^2(u+1))=\widetilde{pg}(\tilde{K}h(u)+1,$$

$\tilde{L}^2(\tilde{K}h(u)+1))$ (暂记为 $B_1(h(u)))=B_1(h(u))$

当 $\tilde{K}(u+1)\neq 0$ 时

$$\tilde{K}(u+1)=\tilde{K}(u)+1,\ \tilde{L}(u+1)=\tilde{L}(u)$$

因此

$$h(u+1)=\widetilde{pg}((u+1),g(\tilde{L}^2(u+1),\tilde{K}\tilde{L}(u+1),\tilde{K}(u+1)))$$

$$=\widetilde{pg}(u+1,g(\tilde{L}^2(u),\tilde{K}\tilde{L}(u),\tilde{K}(u)+1))$$

$$=\widetilde{pg}(u+1,B(\tilde{K}\tilde{L}(u),\tilde{K}(u),g(\tilde{L}^2(u),\tilde{K}\tilde{L}(u),$$

$$\tilde{K}(u))))$$

$$=\widetilde{pg}(\tilde{K}h(u)+1,B(\tilde{K}\tilde{L}\tilde{K}h(u),\tilde{K}^2h(u),\tilde{L}h(u)))$$

(暂记为 $B_2(h(u)))=B_2(h(u))$

故

$$h(u+1)=B_1(h(u))N\tilde{K}(\tilde{K}h(u)+1)+B_2(h(u))$$

$$\cdot N^2\bar{K}(\bar{K}h(u)+1)$$

因此可改用弱原始复迭式来定义 $h(u)$，即

$$\begin{cases} h(0)=0 \\ h(u+1)=B_1(h(u))N\bar{K}(\bar{K}h(u)+1) \\ \qquad\qquad +B_2(h(u))N^2\bar{K}\ (\bar{K}h(u)+1) \end{cases}$$

(3) $g(v,\ u,\ x)=\bar{L}h\widetilde{pg}(x,\ \widetilde{pg}(u,v))$

因此 $g(v,u,\ x)$可用 $h(u)$来定义．又因为 $f(u,\ x)$可表示为 $g(Au,u,x)$，于是 $f(u,\ x)$可用 $h(u)$来定义，即可用弱原始复迭式来定义．故(iii) 得证．因此定理得证．

3.7.3 与原始复迭式的关系

以前我们曾提到原始复迭式呈如下形式

$$\begin{cases} g(x,0)=x \\ g(x,n+1)=f(g(x,n)) \end{cases}$$

或记为

$$g(x,n)=\underset{t\to(x,\ n)}{itr}\ f(t)$$

今证明任何原始递归式在不增加任何开始函数情况下恒可化归为原始复迭式．

定理 3 不增加任何开始函数，仅利用本原函数，迭置和原始复迭式可生成一切原始递归式．

[证] 下面我们使用原始复迭式依次生成有关函数(对有些函数的直观意义读者可作为练习加以阐述)。

① $x+y$

$$\begin{cases} x+0=x \\ x+Sy=S(x+y) \end{cases}$$

② $a\cdot x$ （a 为固定常数，下同）

$$\begin{cases} a\cdot 0=0 \\ a\cdot Sx=a\cdot x+a \end{cases}$$

③ xNy

$$\begin{cases} xN0=x \\ xN(Sy)=0 \end{cases}$$

④ $N(y)=1N(y)$

⑤ $xN^2(y)=xN(Ny)$

⑥ $rs(x,2)$

$$\begin{cases} rs(0,2)=0 \\ rs(Sx,2)=Nrs(x,2) \end{cases}$$

⑦ $rs(x,c)$　　（c 为大于 2 的固定常数）

$$\begin{cases} rs(0,c)=0 \\ rs(Sx,c)=[Srs(x,c)]N[Nrs(rs(x,c),c-1) \\ \qquad\qquad N^2rs(x,c)] \end{cases}$$

⑧ $H_b(x)$　（b 为大于 0 的固定常数）

$$\begin{cases} H_b(0)=0 \\ H_b(Sx)=SH_b(x)+Nrs(H_b(x)+2,b+1) \end{cases}$$

（注意，$H_b(x)$实为 $x+\left[\frac{x}{b}\right]$）

⑨ $p(x)$

$$\begin{cases} p(0)=0 \\ p(Sx)=46Np(x)+P_1p(x)+P_2p(x)+P_3p(x)+P_4p(x) \end{cases}$$

其中

$P_1(x)=(33\cdot x+H_2(x))Nrs(x,5)Nrs(x,2)N^2x$

$P_2(x)=(8\cdot x+H_5(x))Nrs(x,5)N^2rs(x,2)$

$P_3(x)=(14\cdot x+H_3(x))N^2rs(x,5)Nrs(x,3)$

$P_4(x)=(171\cdot x+H_2(x))N^2rs(x,5)N^2rs(x,3)$

（注意，x 为平方数当且仅当 $x=0$ 或 $p(x)$非 3 的倍数）

⑩ $G(x)$

$$\begin{cases} G(0)=0 \\ G(Sx)=SG(x)+2N^2rs(p(G(x)+4),3) \end{cases}$$

⑪ $T(x)$

$$\begin{cases} T(0)=0 \\ T(Sx)=SGT(x) \end{cases}$$

⑫ $F(u,x)$

$$\begin{cases} F(u,0)=u \\ F(u,Sx)=(F(u,x)+2)N[rs(F(u,x),2)N^2rs(pF(u,x),3)] \end{cases}$$

⑬ Dx

$$Dx=[SF(T(x)+x+1,x)]Nrs(x,2)N^2x+F(T(Sx)+Sx+1,Sx)N^2rs(x,2)$$

⑭ $x\dot{-}y$

$$\begin{cases} x\dot{-}0=x \\ x\dot{-}Sy=D(x\dot{-}y) \end{cases}$$

⑮ $R(x)=H_2(G(x)\dot{-}x)\dot{-}(G(x)\dot{-}x)$

⑯ $pg(x,y)=T(x+y)+x$

⑰ $Kx=x\dot{-}TRx$

⑱ $Lx=Rx\dot{-}Kx$

到此，配对函数组已被作出，有了任何一组配对函数组，就可把原始递归式化归为原始复迭式了.

设 $f(u,x)$ 是由 A，B 利用原始递归式定义的函数，即

$$\begin{cases} f(u,0)=A(u) \\ f(u,Sx)=B(u,x,f(u,x)) \end{cases}$$

现令 $g(u,x)=pg(pg(u,x),f(u,x))$

则 $K^2g(u,x)=u,\ LKg(u,x)=x,\ Lg(u,x)=f(u,x)$

$$\begin{cases} g(u,0)=pg(pg(u,0),f(u,0))=pg(pg(u,0),A(u))\equiv A_1(u) \\ g(u,Sx)=pg(pg(u,Sx),f(u,Sx))=pg(pg(u,Sx),B(u,x, \end{cases}$$

$$
\begin{aligned}
&f(u,x)))\\
&=pg(pg(K^2g(u,x),SLKg(u,x)),B(K^2g(u,x),\\
&\quad LKg(u,x),Lg(u,x)))\\
&\equiv B_1(g(u,x))
\end{aligned}
$$

再令

$$
\begin{cases}
g_1(u,0)=u\\
g_1(u,Sx)=B_1g_1(u,x)
\end{cases}
$$

则 $g_1(u,x)$由原始复迭式定义. 有了 $g_1(u,x)$后, 显然有

$$g(u,x)=g_1(A_1(u),x)$$

又 $$f(u,x)=Lg(u,x)=Lg_1(A_1(u),x)$$

于是知 $f(u,x)$可由 $g_1(u,x)$, $A_1(u)$, $L(x)$使用迭置而得, 从而定理得证.

3.7.4 原始递归式与串值递归式的关系

在原始递归式标准形式的第二个定义式的右端, 被定义的函数值只出现一次, 而且这个值是左端的直接前驱(即左端为 $f(u,x+1)$, 右端为 $f(u,x)$), 如果右端出现有多个前驱值 $f(u,x_1)$, $f(u,x_2)$, $\cdots$, $f(u,x_i)$, $\cdots$, $f(u,x_p)$(其中诸 $x_i\leqslant x$, $1\leqslant i\leqslant p$)这种递归式便叫做串值递归式.

串值递归式可用下式作代表

$$
\begin{cases}
f(u,0)=A(u)\\
f(u,x+1)=B(u,x,f(u,x_1),\cdots,f(u,x_p)) \qquad (x_i\leqslant x \quad i=1,\cdots,p)
\end{cases}
$$

定理 4 串值递归式可化归为原始递归式及迭置, 即任何由串值递归式定义的函数均可改用原始递归式和迭置来定义.

[证] 命 $g(u,x)=\prod_{i\to x}P_i^{f(u,i)}$

则有 $B(u,x,f(u,x_1),\cdots,f(u,x_p))=B(u,x,ep_{x_1}g(u,x),\cdots,ep_{x_p}$

$g(u,x))\equiv\tilde{B}(u,x;g(u,x))$

故 $g(u,x)$ 可用下面的原始递归式来定义

$$\begin{cases}g(u,0)=P_0^{f(u,0)}=2^{A(u)}\\ g(u,x+1)=g(u,x)\cdot P_{x+1}^{f(u,x+1)}=g(u,x)\cdot P_{x+1}^{\tilde{B}(u,x,g(u,x))}\end{cases}$$

$f(u,x)$ 可用 $g(u,x)$ 来定义：$f(u,x)=ep_xg(u,x)$

定理得证.

3.7.5 原始递归式与联立递归式的关系

可用原始递归式同时定义多个函数，例如三个函数：

$$\begin{cases}f_1(u,0)=A_1(u)\\ f_2(u,0)=A_2(u)\\ f_3(u,0)=A_3(u)\\ f_1(u,x+1)=B_1(u,x,f_1(u,x),f_2(u,x),f_3(u,x))\\ f_2(u,x+1)=B_2(u,x,f_1(u,x),f_2(u,x),f_3(u,x))\\ f_3(u,x+1)=B_3(u,x,f_1(u,x),f_2(u,x),f_3(u,x))\end{cases}$$

这种式子称为联立递归式，仿此可用串值递归式定义多个函数而得串值联立递归式.

定理 5 联立递归式可化归为原始递归式及迭置.

[证] 今就上面的三个函数的联立递归式来证明，多个函数的联立递归式仿此.

命 $g(u,x)=pg^2f_3(u,x),f_2(u,x),f_1(u,x)$

则诸 f_i 可用 $g(u,x)$ 来定义：

$$f_1(u,x)=Lg(u,x)$$
$$f_2(u,x)=LKg(u,x)$$
$$f_3(u,x)=K^2g(u,x)$$

而 $g(u,x)$ 又可用原始递归式定义如下：

$$\begin{cases}g(u,0)=pg^2f_3(u,0),f_2(u,0),f_1(u,0)\\ \qquad=pg^2A_3(u),A_2(u),A_1(u)\equiv A(u)\\ g(u,x+1)=pg^2f_3(u,x+1),f_2(u,x+1),f_1(u,x+1)\end{cases}$$

$$= pg^2 B_3(u, x, Lg(u, x), LKg(u, x), K^2 g(u, x)),$$
$$B_2(u, x, Lg(u, x), LKg(u, x),$$
$$K^2 g(u, x)), B_1(u, x, Lg(u, x), LKg(u, x),$$
$$K^2 g(u, x)) \equiv B(u, x, g(u, x))$$

因而定理得证.

3.7.6 参数变异递归式

上面介绍的几种递归式有一个共同的特点，即出现在右端的诸 f 中的参数 u 都保持不变，仅仅递归变元 x 在变化，当然参数 u 是可以变化的. 如果第二个定义式的右端出现有 $f(u_1, x), f(u_2, x)$ 等等，而 u_1, u_2 等均是 u, x 的已知函数，那么这种递归式称为**参数变异递归式**. 它也有串值的，联立的等等.

下列递归式就是串值参数变异递归式:

$$\begin{cases} f(u, 0) = A(u) \\ f(u, x+1) = B(u, x, f(u_1, x_1), \cdots, f(u_r, x_r)) \end{cases}$$

其中诸 u_i 及诸 x_i 均为 u, x 的已知函数，诸 $x_i \leqslant x$，即诸 u_i 是指 $u_i(u, x)$，x_i 是指 $x_i(u, x)$，它们都是已知函数，且 $x_i(u, x) \leqslant x$.

可以证明这种递归式可化为原始递归式与迭置.

定理 6 参数变异递归式可化归为原始递归式.

[证] 令 $w(u, x) = u + \max\limits_{i \to u} \max\limits_{j \to x}(u_1(i, j), \cdots, u_r(i, j))$

因为 $u_1, \cdots, u_r$ 均为已知函数，所以 $w(u, x)$ 是由已知函数利用迭大算子构造而得.

显然，当 $y \geqslant x$ 时，$w(u, y) \geqslant w(u, x)$

又令 $$h(u, x) = \prod_{i \to u} \prod_{j \to x} P_{f(i,j)}^{f(i,j)}$$

显然 $h(w(u, y), x) \geqslant h(u, x)$

$$f(i, j) = ep_{f(i,j)}(h(w(u, y), x))$$
$$f(i, x+1) = B(i, x, f(u_1(i, x), x_1(i, x)), \cdots,$$

$$f(u,_r(i,x),x,_r(i,x)))$$

$$=B(i,x,ep(pg(u_1(i,x),x_1(i,x)),h(w(u,y),x),\cdots,ep(pg(u,_r(i,x),x,_r(i,x)),$$

$$h(w(u,y),x))\equiv B_1(i,x,h(w(u,y),x))$$

由此可得

$$h(u,0)=\prod_{i\to u}\prod_{j\to x}P_{pg(i,j)}^{f(i,j)}=\prod_{i\to u}P_{pg(i,0)}^{f(i,0)}=\prod_{i\to u}P_{pg(i,0)}^{A(i)}\equiv A_1(u)$$

$$h(u,x+1)=\prod_{i\to u}\prod_{j\to x}P_{pg(i,j)}^{f(i,j)}\cdot\prod_{i\to u}P_{pg(i,x+1)}^{f(i,x+1)}$$

$$=\prod_{i\to u}\prod_{j\to x}P_{pg(i,j)}^{ep(pg(i,j),h(w(u,y),x))}\cdot\prod_{i\to u}P_{pg(i,x+1)}^{B_1(i,x,h(w(u,y),x))}$$

$$\equiv B_2(u,x,h(w(u,y),x))$$

由此可知，$h(u,x)$ 可用参数变异递归式来定义，但这个递归式中只有一个在前的 h 之值来定义当前的 h 之值.

今作 $\underset{t\to(u,n)}{itr}\ w(t,x)=w^nux\cdots x$

再作 $g(u,y,x)=\begin{cases}h(w^{y\dot{-}x}uy\cdots y,x) & \text{当 } y\geqslant x \text{ 时}\\ 0 & \text{当 } y<x \text{ 时}\end{cases}$

则有 $g(u,y,0)=h(w^yuy\cdots y,0)=A_1(w^yuy\cdots y)\equiv A_2(u,y)$

$$g(u,y,x+1)=\begin{cases}0 & \text{当 } y<x+1 \text{ 时}\\ h(w^{y\dot{-}x\dot{-}1}uy\cdots y,x+1)=B_2(w^{y\dot{-}Sx}uy\cdots y,\\ \quad x,h(w^{y\dot{-}x}uy\cdots y,x))=B_2(w^{y\dot{-}Sx}uy\cdots y,x,\\ \quad g(u,y,x))\equiv B_3(u,y,x,g(u,y,x)) & \text{当 } y\geqslant x+1 \text{ 时}\end{cases}$$

即

$$\begin{cases}g(u,y,0)=A_2(u,y)\\ g(u,y,x+1)=B_3(u,y,x,g(u,y,x))N(Sx\dot{-}y)\end{cases}$$

所以 $g(u,y,x)$ 可用原始递归式来定义

因为 $$h(u,x)=g(u,x,x)$$

$$f(u,x)=ep(pg(u,x),h(u,x))$$

所以用参数变异递归式定义的函数 $f(u,x)$ 可用 $h(u,x)$ 来定义，$h(u,x)$ 又可用原始递归函数 $g(u,y,x)$ 来定义，于是定理得证.

参数变异递归式是很有用的，有些函数很难用原始递归式来定义，但用参数变异递归式来定义却很容易，例如:

$$f(u,x)=B(A_0(u),B(A_1(u),B(A_2(u),\cdots$$
$$B(A_{(x-1)}(u),A_x(u)))\cdots))$$

是很难用原始递归式来定义的. 现用参数变异递归式来定义下列函数 $g(v,u,x)$

$$\begin{cases}g(v,u,0)=A_v(u)\\ g(v,u,x+1)=B(A_v(u),g(v+1,u,x))\end{cases}$$

容易看出

$$g(v,u,x)=B(A_v(u),B(A_{v+1}(u),B(A_{v+2}(u),\cdots,$$
$$B(A_{v+x-1}(u),A_{v+x}(u))\cdots)$$

因此 $f(u,x)=g(0,u,x)$.

3.7.7 多重递归式

以上各递归式都可以叫做单重递归式，其特点是只含有一个递归变元，而且出现在递归式右端的“在前的”f 值都具有这样的性质：它们的递归变元的变值都比左端的 f 值的递归变元的变值小，即所谓“在前的 f 值”，“在后的 f 值”是按递归变元的变值大小为准的.

但是有这样的递归式，它含有多个递归变元，而且决定 f 值的前后关系须同时比较几个递归变元的变值，比较时还须采用“字典次序法”. 设须比较变元组 (x_1,x_2,x_3) 的大小，则字典次序法是:

$$(x_1,x_2,x_3)\text{在}(x_1',x_2',x_3')\text{之前}$$

当且仅当 或者 $x_1<x_1'$，或者 $x_1=x_1'$，$x_2<x_2'$

或者 $x_1=x_1'$，$x_2'=x_2$，$x_3<x_3'$

凡比较 f 值的前后关系时须比较几个变元的大小，且依字典次序法进行比较的递归式叫做*多重递归式*. 如果递归变元有 n 个，便叫做 n *重递归式*.

下面是三重递归式的例子:

$$\begin{cases} f(u,0,0,0)=A_1(u) \\ f(u,0,0,x_3+1)=A_2(u,x_3,f(u,0,0,x_3)) \\ f(u,0,x_2+1,0)=A_3(u,x_2,f(u,0,x_2,x_3')) \\ f(u,0,x_2+1,x_3+1)=A_4(u,x_2,x_3,f(u,0,x_2,x_3'),f(u,0,x_2+1,x_3)) \\ f(u,x_1+1,0,0)=A_5(u,x_1,f(u,x_1,x_2',x_3')) \\ f(u,x_1+1,0,x_3+1)=A_6(u,x_1,x_3,f(u,x_1',x_2',x_3'),f(u,x_1+1,0,x_3)) \\ f(u,x_1+1,x_2+1,0)=A_7(u,x_1,x_2,f(u,x_1,x_2',x_3'),f(u,x_1+1,x_2,x_3')) \\ f(u,x_1+1,x_2+1,x_3+1)=B(u,x_1,x_2,x_3,f(u,x_1,x_2',x_3'), \\ \qquad f(u,x_1+1,x_2,x_3'), \\ \qquad f(u,x_1+1,x_2+1,x_3)) \end{cases}$$

其中 x_1', x_2', x_3' 可为任意值. 前面七个式子称为*开始值定义式*. 最后一个式子称为*递归定义式*.

多重递归式也有串值的，联立的，参数变异的等等.

有些多重递归式可以化归成原始递归式及迭置，有些则不能. 关于这方面的讨论，较为繁杂，这里就从略了. 下面我们给出一个不能化归为原始递归函数的多重递归式.

3.7.8 原始递归函数集的控制函数和非原始递归函数的例子.

定理 7 下列嵌套二重递归式所定义的函数 $g(x,y)$ 是原始递归函数集的控制函数，因而不是原始递归函数.

$$g(0,n)=n+1$$

$$g(m+1, 0)=g(m, 1)$$

$$g(m+1, n+1)=g(m, g(m+1, n))$$

[证] (i) $g(1, n)=n+2$, $g(2, n)=2\cdot n+3$(可用归纳法证之)

(ii) 当 $m\neq 0$ 时, $g(m, n)>n+1$

先对 m 施行归纳法.

奠基 $m=1$ 时, 左端 $=g(1, n)=n+2$ (根据(i))

$$>n+1=\text{右端}$$

归纳 m 处取 $m+1$ 时, 即证: $g(m+1, n)>n+1$

为证该式需再对 n 施行归纳法.

奠基 $n=0$ 时

左端 $=g(m+1, 0)=g(m, 1)>2$ (关于归纳假设)

$$>1=\text{右端}$$

归纳 n 处取 $n+1$ 时

左端 $=g(m+1, n+1)=g(m, g(m+1, n))$

$>g(m+1, n)+1$ (关于 m 的归纳假设)

$>(n+1)+1$ (关于 n 的归纳假设)

$=n+2=$ 右端

故 $g((m+1), n)>n+1$ 成立, 因而(ii)得证.

(iii) $g(m, n)\geqslant n+1>n$(可由(ii)及第一个定义式证明之)

(iv) $g(m, n+1)>g(m, n)$即 $g(m, n)$是关于 n 的严格递增函数, 这是因为

当 $m=0$ 时, 左端 $=g(0, n+1)=n+2>n+1$

$$=g(0, n)=\text{右端}$$

当 $m\neq 0$ 时, 左端 $=g(m-1, g(m, n))>g(m, n)$(根据(iii))

$$=\text{右端}$$

(v) $g(m+1, n)>g(m, n)$即 $g(m, n)$ 是关于 m 的严格递增函

数，这是因为

当 $n=0$ 时

$$\begin{aligned}\text{左端}&=g(m+1,0)=g(m,1)\\&>g(m,0)\qquad\text{(根据(iv))}\\&=\text{右端}\end{aligned}$$

当 $n\neq 0$ 时

$$\begin{aligned}\text{左端}&=g(m+1,n)=g(m,g(m+1,n\dot{-}1))\\&>g(m,(n\dot{-}1)+1)\qquad\text{(根据(iv)和(ii))}\\&=g(m,n)=\text{右端}\end{aligned}$$

(vi) $g(m+1,n)\geqslant g(m,n+1)$

奠基 $n=0$ 时，左端 $=g(m+1,0)=g(m,1)=$ 右端

归纳 n 处取 $n+1$ 时

$$\begin{aligned}\text{左端}&=g(m+1,n+1)=g(m,g(m+1,n))\\&\geqslant g(m,g(m,n+1))\qquad\text{(根据归纳假设)}\\&\geqslant g(m,n+2)\qquad\text{(根据(iii))}\\&=\text{右端}\end{aligned}$$

(vii) 现证 $g(m,n)$ 为原始递归函数集的控制函数，即证：任给一个原始递归函数 $f(x_1,x_2,\cdots,x_r)$，恒有一数 h，使得对于所有的 $x_1,x_2,\cdots,x_r$ 均有

$$f(x_1,x_2,\cdots x_r)<g(h,u)\qquad(u=max(x_1,\cdots,x_r))$$

由定理 2 可知，任意一个原始递归函数，均可由本原函数及 $x+y$, Ex 出发，经有限次迭置以及弱原始复迭式而得．因此可根据原始递归函数的定义过程施行归纳法证明之．

奠基 如果 f 为本原函数或开始函数，则 h 可如下找出：

$$I_{mn}(x_1,\cdots,x_m)=x_n\leqslant u+1=g(0,u)\quad(u=max(x_1,\cdots,x_m))$$

$$O(x)=0<x+1=g(0,x)$$

$$Sx=x+1=g(0,x)<g(1,x)$$

$$x+y\leqslant 2u<2u+3=g(2,u)\qquad (u=max(x,y))$$

$$Ex\leqslant x<x+1=g(0,x)$$

归纳　如果 f 由迭置而得，即设

$$f(x_1,\cdots,x_n)=A(B_1,\cdots,B_m)(x_1,\cdots,x_n)$$

根据归纳假设有 $h_0,h,\cdots,h_m$ 使得

$$A(x_1,\cdots,x_m)<g(h_0,u),\ (u=max(x_1,\cdots,x_m))$$

$$B_i(x_1,\cdots,x_m)<g(h_i,u),\ (u=max(x_1,\cdots,x_n),\ (i=1,\cdots,m))$$

令

$$\tilde{h}=max(h_0,h_1,\cdots,h_m),$$

则有

$$\begin{aligned}
f(x_1,\cdots,x_n)&=A(B_1,\cdots,B_m)(x_1,\cdots,x_n)\\
&<g(h_0,max(B_1,\cdots,B_m)(x_1,\cdots,x_n))\quad\text{(归纳假设)}\\
&<g(h_0,max(g(h_1,u),\cdots,g(h_m,u)))\\
&\qquad\qquad (u=max(x_1,\cdots,x_n))\\
&\qquad\qquad\qquad\text{(归纳假设以及(iv))}\\
&\leqslant g(\tilde{h},g(\tilde{h},u))\qquad\text{(根据(iv)和(v))}\\
&<g(\tilde{h},g(\tilde{h}+1,u))\qquad\text{(根据(iv)和(v))}\\
&=g(\tilde{h}+1,u+1)\\
&<g(\tilde{h}+2,u)\qquad\text{(根据(vi))}
\end{aligned}$$

于是 $max(h_0,h_1,\cdots,h_m)+2$ 即为所求之 h.

如果 f 由弱原始复迭式而得，即设

$$\begin{cases}f(0)=0\\ f(x+1)=Bf(x)\end{cases}$$

根据归纳假设有 h_1，使得 $B(x)<g(h_1,x)$

容易验证 $f(x)<g(h_1+1,\ x)$（可对 x 施行归纳法证之）这样(vii)便得证.

我们已经知道函数集 Δ 的控制函数决不属于该 Δ，因此 $g(m,$

n)决不是原始递归函数. 定理得证.

3.7.9 递归式与数学归纳法

最后必须指出,递归式与数学归纳法之间有着密切的关系,这是因为,"归纳假设"实际上相当于出现在递归定义式右端的被定义函数的"前行值",几乎可以说,它们之间有着一一对应关系,例如:

简单归纳法对应于原始递归式;

强归纳法对应于串值递归式;

参变归纳法对应于参数变异递归式.

与多重递归式相对应的有多重归纳法. 本节的 3.7.8 中的(ii)所用的证明方法就是二重归纳法. 它分以下几步:

(1) $g(0,n)$真;

(2) 如果 $g(m,n)$对一切 n 真,那么 $g(m+1,0)$真;

(3) 如果 $g(m,n)$对一切 n 真,且 $g(m+1,n)$真,那么 $g(m+1,n+1)$真;

(4)如果(1)～(3)均得证,则 $g(m,n)$真. 这与二重递归式的对应是很明显的.

另外,还有一般递归式相对应的归纳法,这里就不讨论了.

习 题

1. 从本原函数以及 $x+y, x\dot{-}y, x\cdot y$ 出发,利用原始递归式定义下列函数:

1.1 $rs(x,y)$;　　1.2 $\left[\frac{x}{y}\right]$;

1.3 $[\sqrt{x}]$;　　1.4 2^x(用复迭式);

1.5 a^x(a 为常数,用复迭式);　　1.6 $T_a x$;

1.7 $R_a x$;　　1.8 $a(0)+(u\dot{-}a(1))+\cdots+(u\dot{-}a(n))$.

2. 求证：由本原函数以及 $x\dot{-}y$，NEx 出发，利用迭置和弱原始复迭式可以作出所有的原始递归函数.

3. 试问下列递归式是什么递归式？并把它们化为原始递归式.

$$\begin{cases} f_1(u,0)=A_1(u) \\ f_2(u,0)=A_2(u) \\ f_3(u,0)=A_3(u) \\ f_1(u,Sx)=B_1(u,x,f_1(u,x),f_2(u,x),f_3(u,x)) \\ f_2(u,Sx)=B_2(u,x,f_1(u,Sx),f_2(u,x),f_3(u,x)) \\ f_3(u,Sx)=B_3(u,x,f_1(u,Sx),f_2(u,Sx),f_3(u,x)) \end{cases}$$

4. 把下列递归式化归为原始递归式以及迭置（如有比正文中给出的通用化归法更为简捷的化归法，请尽量使用）：

4.1 $$\begin{cases} f(u,0)=A(u) \\ f(u,x+1)=B(u,x,\sum\limits_{t\to x}f(u,t)) \end{cases}$$

提示：令 $g(u,x)=\sum\limits_{t\to x}f(u,t)$ 然后进行化归.

4.2 $$\begin{cases} f(u,0)=u+1 \\ f(u,x+1)=2\prod\limits_{t\to x}f(u,t)\cdot\sum\limits_{t\to x}f(u,t) \end{cases}$$

4.3 $f(0)=a(0)$

$f(1)=a(0)^{a(1)}$

$f(2)=a(0)^{a(1)^{a(2)}}$

…

4.4 $f(0)=a(0)$

$f(1)=a(0)\dot{-}a(1)$

$f(2)=a(0)\dot{-}(a(1)\dot{-}a(2))$

…

5. 用多重递归式定义下列函数：

5.1 $m\dot{-}n$

5.2 $\binom{m}{n}=C_m^n$ 即 $\dfrac{m!}{n!(m-n)!}$

6. 设 $f(0,n)=1$

$f(m+1,0)=m+1$

$$f(m+1, n+1) = f(m, m \cdot n) \cdot f(m+1, n)$$

求 $f(2, n)$, $f(3, n)$, $f(4, n)$, $f(5, 5)$ 诸值.

§3.8 一般递归函数与摹状函数

所谓一般递归函数就是由本原函数出发，经过有限次迭置与一般递归式所作成的函数，所有一般递归函数组成的集合称为一般递归函数集. 由本原函数以及 $x+y, x \cdot y, eq(x, y)$ 出发，经过有限多次迭置以及摹状算子 rti 而得的函数称为摹状函数 ①，所有摹状函数组成的集合称为摹状函数集.

一般递归式的标准形式如下:

$$f(u, 0) = A(u)$$

$$f(u, x+1) = B(u, x, f(u, g(u, x+1)))$$

其中 A, B 为已知函数, $g(u, x)$ 为已知的归宿于 0 的函数. 上式中 u 为参数. 一般说, 参数可以有若干个. 这里只写出一个, 因为利用配对函数, 多个参数必可化归为一个参数.

所谓归宿于 0 的函数 $g(u, x)$ 是指该函数具有下列性质: 对于任何 x, 恒有一数 m, 使得

$$\underset{t \to (x, m)}{itr}\, g(u, t) (= g_u^m(x)) = 0$$

且有这种性质的最小的 m, 叫做 g 在 x 处的归宿步骤, 记为 $d(u, x)$.

从形式上看, 一般递归式与原始递归式相差甚微而与摹状算子完全不同, 但是实质上, 一般递归式的力量比之原始递归式要强得多, 而与摹状算子差不多. 下文我们将证明, 一般递归函数集与

---

① 为了简便起见, 本节中的摹状函数均指正常摹状函数, 摹状算子均指正常摹状算子.

幂状函数集相等，凡处处有定义且能够在有限步骤内计算其值的函数都是一般递归函数，也是幂状函数.

3.8.1 一般递归式与原始递归式的关系

定理 1 原始递归函数集是一般递归函数集的真子集.

[证] (i) 原始递归函数集是一般递归函数集的子集.

在 § 3.2.4 中已经证明了原始递归式是一般递归式的特例，即当把一般递归式中的归宿函数 $g(u,x)$取为 $D(x)$时(显然，Dx 为一般递归函数，其归宿函数为 Ox)，一般递归式便为原始递归式. 而本原函数和迭置是两者共有的. 故原始递归函数集是一般递归函数集的子集.

(ii) § 3.7.8 中给出的非原始递归函数，即下列函数

$$\begin{cases} f(0,n)=n+1 \\ f(m+1,0)=f(m,1) \\ f(m+1,n+1)=f(m,f(m+1,n)) \end{cases}$$

是一般递归函数.

首先，引进下列函数

$$a(x)=\begin{cases} P_0^{ep_0(x)}P_1^{ep_1(x)}\cdots P_{H(x)-1}^{ep_{H(x)-1}(x)\div 2(x)}, & \text{当 } H(x)\geqslant 2 \text{ 时} \\ 1, & \text{当 } H(x)\leqslant 1 \text{ 时} \end{cases}$$

其中 $H(x)$表示 x 的最大质因子的足码，且 $H(0)=H(1)=0$，注意，在 § 3.6.4 中已经证明 $H(x)$为初等函数；实际上 $a(x)$是从 x 中删去最后两个质因子的结果. 例如:

若 $x=99=2^0\cdot 3^2\cdot 5^0\cdot 7^0\cdot 11^1=P_0^0P_1^2P_2^0P_3^0P_4^1$

则 $H(99)=4,\quad a(99)=P_0^0P_1^2P_2^0=9$

又若 $x=36=2^2\cdot 3^2=P_0^2P_1^2$

则 $H(36)=1a(36)=1$

其次，$h(x)$的定义如下:

$$h(0)=0$$

$$h(x+1)=\begin{cases}h(g_1(x+1))=h(a(x+1)P^{ep}_{(0)}{}^{(x+1)+1}_{(-1)}),\\ \qquad\text{当}H(x+1)\neq 0\wedge ep_{(-1)}(x+1)=1\text{时}\\ h(g_2(x+1))=h(a(x+1)\cdot P^{ep}_{(-1)}{}^{(x+1)\dot{-}1}_{(0)}P^2_{(0)}),\\ \qquad\text{当 }H(x+1)\neq 0\wedge ep_{(-1)}(x+1)\\ \qquad\geqslant 2\wedge ep_{(0)}(x+1)=1\text{ 时}\\ h(g_3(x+1))=h(a(x+1)P^{ep}_{(-1)}{}^{(x+1)\dot{-}1}_{(0)}\\ \qquad P^{ep}_{(0)}{}^{(x+1)}_{(-1)}\cdot P^{ep}_{(+1)}{}^{(x+1)\dot{-}1}_{(0)},\\ \qquad\text{当 }H(x+1)\neq 0\wedge ep_{(-1)}(x+1)\\ \qquad\geqslant 2\wedge ep_{(0)}(x+1)\geqslant 2\text{ 时}\\ ep_0(x+1)\dot{-}1\qquad\text{此外, 即 }H(x+1)=0\vee ep_{(-1)}\\ \qquad(x+1)=0\text{时}\end{cases}$$

$$\text{令}g(x)=\begin{cases}g_1(x)=a(x)\cdot P^{ep}_{(-1)}{}^{(x)+1}_{(0)},\ \text{当 }H(x)\neq 0\wedge ep_{(-1)}x=1\text{时}\\ g_2(x)=a(x)\cdot P^{ep}_{(-1)}{}^{(x)\dot{-}1}_{(0)}\cdot P^2_{(0)},\text{当 }H(x)\neq 0\wedge ep_{(-1)}\\ \qquad x\geqslant 2\wedge ep_{(0)}x=1\text{ 时}\\ g_3(x)=a(x)\cdot P^{ep}_{(-1)}{}^{(x)\dot{-}1}_{(-1)}\cdot P^{ep}_{(0)}{}^{(x)}_{(-1)}\cdot P^{ep}_{(+1)}{}^{(x)\dot{-}1}_{(0)},\\ \qquad\text{当 }H(x)\neq 0\wedge ep_{(-1)}x\\ \qquad\geqslant 2\wedge ep_{(0)}x\geqslant 2\text{ 时}\\ 0,\qquad\text{此外, 即 }H(x)=0\vee ep_{(-1)}(x)=0\text{ 时}\end{cases}$$

(其中(−1)表 $H(x)\dot{-}1$, (0)表 $H(x)$, (+1)表 $H(x)+1$)

显然 $h(x)$可表为

$$h(0)=0$$

$$h(x+1)=\begin{cases}h(g(x+1)),\ \text{当 }H(x+1)\neq 0\wedge ep_{(-1)}(x+1)\geqslant 1\text{时}\\ ep_0(x+1)\dot{-}1,\ \text{当}H(x+1)=0\vee ep_{(-1)}(x+1)=0\text{时}\end{cases}$$

因而可表为

$$\begin{cases}h(0)=0\\ h(x+1)=B(x,h(g(x+1)))\end{cases}$$

可以证明 $g(x)$是归宿于 0 的函数(请自行证明), 所以 $h(x)$是

一般递归函数.

其三, $h(x)$有如下性质: 当 $H(x)\geqslant 1 \wedge ep_{(-1)}x\geqslant 1$ 时

$$h(a(x)\cdot P_{(-1)}^{m+1}\cdot P_{(0)}^{n+1})=h(a(x)\cdot P_{(-1)}^{f(m,n)+1}) \qquad (1)$$

这可用归纳法证明之, 先对 m 施行归纳

奠基 $m=0$ 时

$$左端=h(a(x)P_{(-1)}P_{(0)}^{n+1})=h(a(x)P_{(-1)}^{n+2})=右端$$

归纳 m 取 $m+1$ 时, 这时归纳假设为: 情形 m 对于任何 n 均成立, 须证: 情形 $m+1$ 对于任何 n 均成立, 即证:

$$h(a(x)\cdot P_{(-1)}^{m+2}\cdot P_{(0)}^{n+1})=h(a(x)\cdot P_{(-1)}^{f(m+1,n)+1})$$

为此再对 n 施行归纳

奠基 $n=0$ 时

$$左端=h(a(x)\cdot P_{(-1)}^{m+2}\cdot P_{(0)})=h(a(x)\cdot P_{(-1)}^{m+1}\cdot P_{(0)}^{2})$$
$$=h(a(x)\cdot P_{(-1)}^{f(m,1)+1}) \quad (关于m的归纳假设)$$
$$=右端$$

归纳 n 为 $n+1$ 时, 这时归纳假设为: 情形 $m+1$ 对于 n 成立.

$$左端=h(a(x)\cdot P_{(-1)}^{m+2}P_{(0)}^{n+2})=h(a(x)\cdot P_{(-1)}^{m+1}\cdot P_{(0)}^{m+2}\cdot P_{(+1)}^{n+1}$$
$$=h(a(x)\cdot P_{(-1)}^{m+1}\cdot P_{(0)}^{f(m+1,n)+1}) \quad (关于n的归纳假设)$$
$$=h(a(x)\cdot P_{(-1)}^{f(m,f(m+1,n))+1}) \quad (关于m的归纳假设)$$
$$=h(a(x)\cdot P_{(-1)}^{f(m+1,n+1)+1})=右端$$

故依归纳法(1)得证.

其四, 容易证明: $f(m,n)=h(2^{m+1}\cdot 3^{n+1})$ (2)

这是因为

$$右端=h(2^{m+1}\cdot 3^{n+1})=h(2^{f(m,n)+1}) \quad (由(1)而得)$$
$$=(f(m,n)+1)\dot{-}1(由\ h(x)的定义)=f(m,n)$$

所以(2)得证.

这就是说, $f(m,n)$ 可由一般递归函数 $h(x)$ 利用迭置而得. 故

$f(m,n)$是一般递归函数，因此 ii)得证，于是定理得证.

3.8.2 一般递归式与不受限幕状式的关系

虽然表面上幕状式(或者幕状算子)的形式与一般递归式很不相同，但是实际上，两者的力量是差不多的，这里我们将证明幕状函数集与一般递归函数集是相等的.

定理 2 幕状函数集是一般递归函数集的子集，即所有幕状函数都是一般递归函数.

[证] 我们知道，所谓幕状函数是由本原函数以及$x+y$, $x\cdot y$, $eq(x,y)$出发，经过有限次迭置，以及幕状算子 rti 而得的函数.

因为本原函数和迭置是两者共有的，而 $x+y, x\cdot y, eq(x,y)$ 是初等函数，因而是一般递归函数. 为了证明本定理，只须证明幕状算子可在一般递归函数集中表示之.

设 $f(u)$是由 $A(u,x)$利用幕状算子而得，即

$$f(u)=rti\ A(u,t)$$

其中 u 是参数，一般情形可有多个参数，但是利用配对函数后恒可化为一个.

今证明:

$$(*)\qquad f(u)=\underset{t\to N^2A(u,0)}{stp}(St)N^2A(u,t)$$

设 $f(u)=m$, 即 $A(u,t)$的最小 t 零点为 m.

① 若 $m=0$, 则 $A(u,0)=0$, $N^2A(u,0)=0$, 所以(*)式右边为$\underset{t\to 0}{stp}(St)N^2A(u,t)=0$ 即(*)式成立.

② 若 $m\neq 0$, 则 $A(u,i)\neq 0$, $i=0,\cdots,m-1$; $A(u,m)=0$.

令 $g(u,t)=(St)N^2A(u,t)$; 这时

$$\underset{t\to N^2A(u,0)}{stp}g(u,t)=\underset{t\to 1}{stp}\,g(u,t)$$

$$g(u,1)=(S1)N^2A(u,1)=S1=2$$

$$g^2(u,1)=g(u,g(u,1))=g(u,2)=(S2)N^2A(u,2)=S2=3$$

……

$$g^{m-1}(u,1)=g(u,g^{m-2}(u,1))=g(u,m-1)=S(m-1)$$

$$N^2A(u,m-1)=S(m-1)=m$$

最后，$g^m(u,1)=g(u,g^{m-1}(u,1))=g(u,m)=(Sm)N^2A(u,m)=(Sm)N^20=0$，也即 $g(u,t)$ 在 $t=1$ 处的归宿步骤为 m.

所以 $\underset{t\to 1}{stpg}(u,t)=m=f(u)$.

于是(*)式得证.

定理 3 一般递归函数集是幂状函数集的子集.

[证] (i) $x+y, x\cdot y, eq(x,y), x\dot{-}y, [x/y], [\sqrt{x}], rs(x,y)$均是幂状函数，这是因为

① $x+y$ 是开始函数

② $x\cdot y$ 是开始函数

③ $eq(x,y)$是开始函数

④ $x\dot{-}y=\underset{t}{rti}eq(t^2+4\cdot x\cdot y,(x+y)^2)$

$N^2x=\underset{t}{rti}eq(x\cdot t,x)$, $Nx=1\dot{-}N^2x$

$x\dot{-}y=(x\dot{-}y)N^2(y\dot{-}x+x\dot{-}y)$

⑤ $[x/y]=\underset{t}{rti}((x+1)N^2y\dot{-}(t+1)\cdot y)$

⑥ $[\sqrt{x}]=\underset{t}{rti}((x+1)\dot{-}(t+1)^2)$

⑦ $rs(x,y)=x\dot{-}y[x/y]$

所以，以上七个函数均是幂状函数.

(ii) 因为 $pg(x,y)=(x+y)^2+x$, $Kx=x\dot{-}[\sqrt{x}]^2$, $Lx=[\sqrt{x}]\dot{-}Kx$，所以该配对函数组都是幂状函数.

(iii) 因为$tm(i,w)=rs(Kw,1+(i+1)Lw)$，所以 $tm(i,w)$是幂状函数.

注意，§3.5 中已证明，任给一数列 $a_0,a_1,\cdots,a_n$，必有一数 w，使得

$$tm(i,w)=a_i \qquad (0\leqslant i\leqslant n)$$

设 w_0 是所有这些 w 中的最小者，则有

$$w_0 = rti_i \max_{i \to n} eq(tm(i, t), a_i)$$

因为 $\max_{i \to n} N^2 f(i) = N^2 f(rti_i((i \dot{-} n) N f(i)))$，即 $\max_{i \to n} N^2$ 可用幂状式和迭置表示. 注意到 $\max_{i \to n} eq(x_i, y_i) = \max_{i \to n} N^2 eq(x_i, y_i)$，便可知 $\max_{i \to n} eq(tm(i, t), a_i)$ 是幂状函数，因而

$$rti_i \max_{i \to n} eq(tm(i, t), a_i)$$

是幂状函数.

(iv) 设 $f(u, x)$ 是由函数 A, B 和归宿函数 g，利用一般递归式而得的函数，即

$$\begin{cases} f(u, 0) = A(u) \\ f(u, x+1) = B(u, x, f(u, g(x+1))) \end{cases}$$

今证 $f(u, x)$ 必为幂状函数.

首先，令 $a_i = g^i(x) \qquad (i = 0, 1, \cdots, n+1)$

由(iii)知，必有 $v(x, n)$，使得

$$tm(i, v) = g^i(x) \qquad (i = 0, 1, \cdots, n+1)$$

并且 $v(x, n) = rti_i(\max_{i \to n+1} eq(tm(i, t), g^i(x)))$

$$= rti_i(eq(tm(0, t), x) + \max_{i \to n} eq(tm(i+1, t), g^{i+1}(x)))$$

$$= rti_i(eq(tm(0, t), x) + \max_{i \to n} eq(tm(i+1, t), g(tm(i, t))))$$

所以 $v(x, w)$ 是幂状函数. 从而有

$$g^i(x) = tm(i, v(x, n))$$

所以 $g^i(x)$ 是幂状函数.

因为 $g(x)$ 在 x 处的归宿步骤 $d(x) = rti_i\, g^i(x)$，

所以有 $\quad d(x) = rti_i\, tm(i, v(x, n))$

故 $d(x)$是摹状函数.

其次, 又令 $a_i=f(u, g^i(x))\qquad (i=0, 1, \cdots, d(x))$

由(iii)知, 必有 $w(u, x)$, 使得

$$tm(i, w(u, x))=f(u, g^i(x))\qquad (i=0, 1, \cdots, d(x))$$

而且 $w(u, x)=\underset{t}{rti}(eq(tm(d(x), t), f(u, 0))$

$$+N^2d(x)\cdot \max_{i\to d(x)\dot{-}1} eq(tm(i, t), f(u, g^i(x))))$$

$$=\underset{t}{rti}(eq(tm(d(x), t), A(u))$$

$$+N^2d(x)\cdot \max_{i\to d(x)\dot{-}1} eq(tm(i, t), B(u, g^i(x)\dot{-}1, tm(i+1, t))))$$

所以 $w(u, x)$是摹状函数.

显然有 $\qquad f(u, x)=tm(0, w(u, x))$

所以 $f(u, x)$是摹状函数, (iv)得证, 于是定理得证.

由定理 2 和定理 3 可知, 一般递归函数集与摹状函数集完全相等. 因此今后我们不再区别摹状函数和一般递归函数, 在讨论它们时, 摹状式和一般递归式均可使用.

习　题

1. 证明下列函数为归宿于 0 的函数:

(a) $g(x)=\begin{cases} a(x)P_{(-1)}^{ep_{(0)}(x)+1}, & \text{当}H(x)\neq 0\wedge ep_{(-1)}(x)=1\text{ 时} \\ a(x)P_{(-1)}^{ep_{(-1)}(x)\dot{-}1}P_{(0)}^{ep_{(-1)}(x)}, & \\ \quad \text{当}H(x)\neq 0\wedge ep_{(-1)}(x)\geqslant 2\wedge ep_{(0)}(x)=1\text{ 时} & \\ a(x)P_{(-1)}^{ep_{(-1)}(x)\dot{-}1}P_{(0)}^{ep_{(-1)}(x)}P_{(+1)}^{ep_{(0)}(x)\dot{-}1}, & \\ \quad \text{当}H(x)\neq 0\wedge ep_{(-1)}(0)\geqslant 2\wedge ep_{(0)}(x)\geqslant 2\text{ 时} & \\ 0, \quad \text{此外, 即 } H(x)=0\vee ep_{(-1)}(x)=0\text{ 时} \end{cases}$

(关于 $a(x)$, $H(x)$, (-1), (0), $(+1)$等的约定见书中定理)

(b) $g(x)=\begin{cases}pg(Kx, 4Lx+5), & \text{当 } Lx<(Kx)^3 \text{ 时}\\ pg(Kx\dot{-}2, (Lx)^2\cdot Kx), & \text{当 } Lx\geqslant(Kx)^3 \text{ 时}\end{cases}$

(c) $g(x)=\begin{cases}pg(Kx, Lx\dot{-}1), & \text{当 } Kx \text{ 为偶数且 } Lx\neq 0 \text{ 时}\\ pg(Kx\dot{-}1, (Lx)^2), & \text{当 } Kx \text{ 为奇数或 } Lx=0 \text{ 时}\end{cases}$

(d) $g(x)=E_0x$

2. 试用塞状式表示下列函数:

$f(0, n)=n+1$;

$f(m+1, 0)=f(m, 1)$;

$f(m+1, n+1)=f(m, f(m+1, n))$.

§3.9 能行可计算函数

通常我们所碰到的有定义的函数有两种类型，一种是尽管其"对应法则"已经给定，但按照这种对应法则却无法计算在有定义的地方的值．例如下面的函数:

$$f(n)=\begin{cases}1, & \text{当 } n\geqslant 3 \text{ 且方程 } x^n+y^n=z^n \text{ 有整数解时}\\ 0, & \text{此外情形}\end{cases}$$

尽管在理论上说，此函数对任何自然数 n 都有定义，但由于无法判定当 $n\geqslant 3$ 时 $x^n+y^n=z^n$ 是否有整数解，故对任何 $n\geqslant 3$, $f(n)$ 的值实际上是无法计算出来的．又例如下面的函数:

$$g(n)=\begin{cases}1, & \text{当在 } \pi \text{ 的十进制展式中有相继 } n \text{ 个 } 0 \text{ 时}\\ 0, & \text{此外情形}\end{cases}$$

也是无法计算 $g(n)$的值(尽管 $g(n)$对任何自然数都有定义).

另一种函数是可以按照定义的过程在有限步内计算它(有定义)的函数值．这种函数通常称为能行可计算函数或算法可计算函数．例如上面讨论的一般递归函数就是能行可计算函数．当然能行可计算函数与"实际"可计算函数还有区别，但是把那些能行可计算函数从一般函数中区分开来是具有很大的理论意义和实际意义的.

3.9.1 形式系统

任意一个计算，即使是“心算”，人们总可以设法把它记录下来，在记录计算过程时，必须使用一些符号，如数字，变元，函数，等号，括号，逗号等等．不同符号的个数可以有限也可以无限．对所有这些符号我们恒可以赋以不同的编号．不失一般性，可以假定所使用的符号只限于下列这些(顺便列出对它们的编号)：

计算中所使用的符号	=	(	)	,	数字 n	变元 x_n	函数 σ_n
编　　号	1	2	3	4	$3n+5$	$3n+6$	$3n+7$

这里假定：每个数都用不同的数字表示，而不是只由十个数字表示，即这里把象“489”那样的数看作一个新符号，而不看成由三个数字 4, 8, 9 并列而成的符号．这里所给的编号显然与所使用的符号是一一对应的，任给一符号立即可求得其编号，反之，任给一编号也立即可找出它所代表的符号．

我们知道，一个计算过程，总是由一连串的式子组成，而每个式子总是由一系列符号组成．对于“式子”(即符号系列)及“式子序列”，我们也可以进行编号，使得不同的式子有不同的编号，不同的式子序列有不同的编号．编号方法很多，我们采用下列编号法．

设式子 E 由符号 $a_1, a_2, \cdots, a_k$ 依次毗连而成，又设诸 a_i 的编号为 n_i $(i=1, \cdots, k)$．则式子 E 的编号约定为 $\underset{i\to k}{seq}\, n_i$ (并指定 $n_0=k$)，也可写为 $seq[k, n_1, \cdots, n_k]$．其次设式子序列 P 由 h 个式子 $E_1, E_2, \cdots, E_h$ 依次组成，又设诸 E_i 编号为 e_i $(i=1, \cdots, h)$，则式子序列 P 的编号约定为 $\underset{i\to h}{seq}\, e_i$ (指定 $e_0=h$)，也可写为 $seq[h, e_1, \cdots, e_h]$．这样，每个符号，每个式子，每个式子序列，都对应于一数，例如，式子

$$x_0 = 3+2$$

对应于数: $seq[5, 6, 1, 14, 10, 11]$

其中设 σ_1 表示“+”. 又如果一式子的编号为 360, 则该式子可如下求出: 先求 $tm(0, 360)$, 得出式子中符号的个数(设该数为k). 然后依次求 $tm(1, 360)\,tm(2, 360)$, …, $tm(k, 360)$. 便得相应的符号序列.

同样, 下列式子序列

$$x_0 = 3+2, \qquad x_0 = 5$$

的编号是 $\mathrm{seq}[2, \mathrm{seq}[5, 6, 1, 14, 10, 11], \mathrm{seq}[3, 6, 1, 20]]$. 反之, 如欲求编号为 360 的式子序列, 亦可同法进行. 易知, “360”不是任何式子序列的编号.

显然, 当给出一编号时, 必须指明是什么东西的编号, 是符号的编号还是式子的编号, 还是式子序列的编号.

在计算过程中, 必须使用一些“运算”. 这是计算过程的主要内容. 所谓运算, 是指由若干个式子得到一个新式子所使用的方法. 例如, 由式子 $x_1^2+3x_1+2$ 使用代入运算可得新式子$4^2+3.4+2$(把 x_1 代入以数字 4).

显然, 运算可以看作是以式子为变域而以式子为值的函数。因此, 设有一运算 Φ, 它把 r 个式子变成一个新式子, 则可表为

$$G = \Phi(F_1, F_2, \cdots, F_r)$$

其中诸 F_i 及 G 都是式子.

既然对每个式子都进行了编号, 因此极易使每一种运算对应于一数论函数. 设诸 F_i 的编号为 x_i, G 的编号为 y, 则 y 与诸 x_i 之间显然有函数关系. 设该关系的特征函数为 $f(x_1, \cdots, x_r, y) = 0$, 则 f 便叫做相应于运算Φ的数论函数. 当符号及式子的编号确定后, 相应于运算Φ的数论函数也就确定了.

最常用而又最重要的运算有两个, 即代入运算及替换运算.

现讨论相对于这两个运算的数论函数.

代入 设原式为 E, 把 E 中所有变元 x_j 的出现均代入以数字 a, 得到式子 E', 则说 E' 是在 E 中把 x_j 代以 a 而得. 记为

$$E'=\mathrm{sub}(E, j, a)$$

设 E 的编号为 e, E' 的编号为 e', 则应有一数论函数 sb, 使得

$$sb(e, j, a, e')=0$$

今求该数论函数 sb.

我们知道, e 与 e' 之间的关系应是:

(1) $tm(0, e)=tm(0, e')$ (即 E 和 E' 中符号个数相等).

(2) $\underset{i\to tm(0,\ e)}{\forall}[(i\neq 0\wedge tm(i, e)=3\cdot j+6)\supset(tm(i, e')=3a+5)]$

(3) $\underset{i\to tm(0,\ e)}{\forall}[(i\neq 0\wedge tm(i, e)\neq 3\cdot j+b)\supset(tm(i, e')=tm(i, e))]$

作合取之后, 其特征关系显然为 e, j, a, e' 的初等函数(这是因为初等函数集对命题联结词和受限量词是封闭的).

替换 设有一式子 F(编号为 f)呈 $A=B$ 形, 又有式 E(编号为 e), 在 E 中把从第 $t+1$ 个符号开始的部分公式 A 代换以 B 所得的式子为 E'(编号为 e'), 则说 E' 是在 E 中第 t 个符号处依 F 作替换的结果. 记为

$$E'=rep(F, E, t)$$

同样应有一数论函数 rp, 使得

$$rp(f, e, t, e')=0.$$

今求该函数 rp.

我们知道, e, f, e' 之间应有下面的关系:

(1) $\underset{n\to lh(0,\ f)}{\exists}(tm(n+1, f)=1)$ (即 F 中必有一等号)

令 $L=\underset{n\to tm(0,f)}{rti}[tm(n+1,f)\doteq 1]$(即 E 中第 $L+1$ 个符号为等号)

因为等号的左端为 A, 故 A 的符号个数为 L, 等号的右端为 B, 故 B 的符号个数为 $tm(0,f)\doteq(L+1)$, 记为 q.

因为 L 是由初等函数利用受限幂状式而得, 故为初等函数, 因而 q 也是初等函数.

(2) E 中第 $t+1,\cdots,\ t+L$ 个符号分别与 F 中第 $1,\cdots,L$ 个符号(即 A)相同, 故

$$\underset{i\to(L\doteq 1)}{\forall}(tm(i+1,f)=tm(t+i+1,e))$$

(3) E' 中第 $t+1,\cdots,t+q$ 个符号分别与 F 中第 $L+2,\cdots,L+q+1$ 个符号(即 B)相同.

故 $\underset{i\to(q\doteq 1)}{\forall}(tm(t+i+1,e')=tm(L+i+2,f))$

(4) E 中第 $1,\cdots,\ t$ 个符号分别与 E' 中第 $1,\cdots,\ t$ 个符号相同, 即 $\underset{i\to t\doteq 1}{\forall}(tm(i+1,e)=tm(i+1,e'))$

(5) E 中第 $t+L+1,\cdots,tm(0,e)$ 个符号分别与 E' 中第 $t+q+1,\cdots,tm(0,\ e')$ 个符号相同, 即(把 $tm(0,e)\doteq(t+L+1)$ 记为 α)

$$\underset{i\to\alpha}{\forall}(tm(t+L+1+i,e)=tm(t+q+1+i,e'))$$

(6) $tm(0,e')=tm(0,e)+q\doteq L$

将(1)～(6)作合取, 其特征函数显然是初等函数.

可见, 相应于代入运算和替换运算的数论函数为初等函数, 对于别的运算可能为更简单的函数(如为五则函数). 也可能为更复杂的函数(如原始递归函数, 或一般递归函数).

上面我们详细讨论了计算时所使用的"原料"(即符号, 式子)及"工具"(即运算), 下面便可讨论什么是计算过程了.

定义 设有一组等式 E(叫做定义等式组)及若干个运算(叫做容许运算) $\Phi_1,\ \cdots,\ \Phi_s$, 如果式子序列 $F_1,\ \cdots,F_k$ 满足下列条

件:

(i) 诸 F_i 或者为定义等式组 E 中某等式之一.

(ii) 或者为由前面若干个 $F_j(j<i)$ 根据某一容许运算而得.

(iii) 最后一个式子 F_k 是一等式，其左端是 $\sigma_0(a_1, \cdots, a_r)$ 而右端是一数字. 则说该式子系列是由定义等式组 E 根据所给的容许运算而对 $\sigma_0(a_1, \cdots, a_r)$ 而作的计算过程，F_k 右端的数字称为 σ_0 在 $(a_1, \cdots, a_r)$ 处的值.

定义 如果定义等式组 E 及容许运算是不随 $a_1, \cdots, a_r$ 的改变而更改，即用同样的 E 及容许运算可以对 σ 的一切有定义的地方而计算其值，则说 σ 是可半计算的. 如果 σ 处处有定义，又是可半计算的，则说 σ 可完全计算的或说 σ 是能行可计算的.

这样我们便对能行可计算性下了一个精确的定义，从而可回答前面所提出的问题，即所有能行可计算的函数是否都是一般递归函数.

定理 4 如果函数 σ 可以由定义等式组 E 及容许运算 $\Phi_1, \cdots, \Phi_s$ 而能行计算，而诸 Φ_i 的相应数论函数为 g_i，则 σ_0 必为一般递归于 $g_1, \cdots, g_s$ 的函数.

[证] 设定义等式组 E 的编号为 m，则 E 中每个等式的编号是

$$tm(i, m) \qquad (i=1, 2, \cdots, tm(0, m))$$

因为 σ_0 是能行可计算的，所以任给一组值 $(a_1, \cdots, a_n)$，均有一计算 $\sigma_0(a_1, \cdots, a_n)$ 之值的计算过程 $F_1, \cdots, F_k$.

设该计算过程的编号为 w，则诸 F_i 的编号便是

$$tm(i, w) \qquad (i=1, 2, \cdots, tm(0, w))$$

今把 w 所应满足的条件表示如下:

(1) 由"F_i 为定义等式组中某一等式"可得

$$\underset{t\to(lm(0,\ m)\dot{-}1)}{\exists}(tm(i,w)=tm(t+1,m))$$

该语句的特征函数为下列初等函数

$$\underset{t\to(lm(0,\ m)\dot{-}1)}{min}\ eq(tm(i,w),tm(t+1,m))$$

(记为 $Q_1(i,m,w)$)

(2) 由"F_i 为由前面若干个 F_j(设为 $F_{j1}, F_{j2}, F_{j3}, \cdots, F_{j_k}$)根据运算 Φ_r(相应的数论函数为 g_r)而得", 可得

$$\underset{j_1\to(i\dot{-})}{\exists}\ \underset{j_2\to(i\dot{-}1)}{\exists}\cdots\underset{j_k\to(i\dot{-}1)}{\exists}g_r(tm(j_1,w),\cdots,tm(j_k,w),tm(i,w))=0$$

该语句的特征函数为下列初等于 g_r 的函数

$$\underset{j_1\to(i\dot{-}1)}{min}\ N^2\underset{j_2\to(i\dot{-}1)}{min}\ N^2\cdots\underset{j_k\to(i\dot{-}1)}{min}\ N^2g_r(tm(j_1,w),\cdots,tm(j_k,w),tm(i,w))$$

(记为 $Q_{2r}(i,w)$), 这里 r 必满足: $1\leqslant r\leqslant s$(s 为定数), 也即有

$$\underset{r\to(s\dot{-}1)}{\exists}Q_{2(r+1)}(i,w)=0$$

其特征函数为

$$Q_{21}(i,w)\cdot Q_{22}(i,w)\cdots Q_{2s}(i,w)\ \text{(记为 } Q_2(i,w))$$

显然 $Q_2(i,w)$ 是初等于 $g_1, g_2, \cdots, g_s$ 的函数.

因为对于一切非零的 i, F_i 或满足(1)或满足(2), 故得

$$\underset{i\to(lm(0,\ w)\dot{-}1)}{\forall}(Q_1(i+1,m,w)\vee Q_2(i+1,w))$$

其特征函数为

$$\underset{i\to(lm(0,\ w)\dot{-}1)}{max}\ (Q_1(i+1,m,w)\cdot Q_2(i+1,w))$$

此函数仍为初等于 $g_1, \cdots, g_s$ 的函数.

(3) 由"最后一个式子 F_k 是一等式, 是 $\sigma_0(a_1,\cdots,a_n)=L$ 形"(这里 n 为定数, L 为某一数字)而得(把 F_k 的编号即 $tm(\dot{tm}(0,w),w)$ 记为 Tw)

$$\underset{L\to w}{\exists}\ [tm(0,Tw)=2n+4\wedge tm(1,Tw)$$
$$=7\wedge tm(2,Tw)=2$$
$$\wedge tm(3,Tw)=3a_1+5\wedge tm(4,Tw)=4$$
$$\wedge\cdots\wedge tm(2n,Tw)=4\wedge tm(2n+1,Tw)=3a_n+5$$
$$\wedge tm(2n+2,Tw)=3\wedge tm(2n+3,Tw)=1$$
$$\wedge tm(2n+4,Tw)=3L+5]$$

其特征函数显然是初等函数，记为 $Q_3(n,a_1,\cdots,a_n,w)$

故知，w 应为下列函数的零点

$$\underset{i\to(tm(0,w)\dot{-}1)}{max}\ (Q_1(i+1,m,w)\cdot Q_2(i+1,w))$$
$$+Q_3(n,a_1,\cdots,a_n,w)$$

该函数记为 $R(m,n,a_1,\cdots,a_n,w)$.

因为 σ_0 是能行可计算的，亦即 σ_0 是处处有定义的函数，$R(m,n,a_1,\cdots,a_n,w)$ 必有关于 w 的零点(因为必有计算过程，而该计算过程的编号 w 满足该条件)，设其零点为 w_0，则有

$$w_0=\underset{w}{rti}R(m,n,a_1,\cdots,a_n,w)$$

故 w_0 为一般递归于 $g_1,\cdots,g_s$ 的函数.

既得 w_0，便可由下法求得 L(即 $\sigma_0(a_1,\cdots,a_n)$)之值:

F_h 的编号为 $tm(tm(0,w_0),w_0)$，记为 y_0，

F_h 的最后一个符号的编号为 $tm(tm(0,y_0),y_0)$，

因而有 $\quad tm(tm(0,y_0),y_0)=3L+5$

故得 $\quad L=[(tm(tm(0,y_0),y_0)\dot{-}5)/3]$

显然，L 是关于 w_0 的初等函数，记为 $\nabla(w_0)$

故得
$$L=\sigma_0(a_1,\cdots,a_n)=\nabla(w_0)$$
$$=\nabla(\underset{w}{rti}R(m,n,a_1,\cdots,a_n,w))$$

由此可知，σ_0 是一般递归于 $g_1, g_2, \cdots, g_n$ 的函数，定理得证.

由此定理可知，只要诸容许运算 Φ_i 的相应的数论函数 g_i 是一般递归函数，则能行可计算的函数必为一般递归函数.

现在的问题是：有没有这样的容许运算，其相应的数论函数不是一般递归函数？根据经验知道，这样的容许运算是不存在的. 因此邱吉提出了一个假设，除了一般递归函数外，再没有能行可计算的函数了. 这个假设称为"邱吉论题"，其严格的叙述如下：

邱吉论题 能行可计算的数论函数与一般递归函数等同. 这个论题是没有证明的，只是根据经验而作的猜测，直到目前为止，还未发现这个论题的反例.

3.9.2 图灵机器

图灵机是一种非常简单但力量极强的理想的"计算机". 这里我们不准备介绍图灵机的通常描述方法，而是用一种类似FORTRAN 程序设计语言的语言来刻划它的功能. 首先非形式地描述一下这种语言.

我们用字母 X, Y, Z, X_1, Y_1, Z_1, X_2, Y_2, Z_2, …表示变元. 这些变元都可取自然数为值. 在这些变元中约定用带或不带下标的X看作输入，它们常在程序开始时接受某初始值；变元Y 看作输出，当程序结束时它含有计算的结果；带或不带下标的 Z 常被看作临时变元，它们含有计算的中间结果. 程序开始时除接受了输入数的变元外，另外一切变元的初值都为 0. 另外我们恒用字母 A, B, C, …, A_1, A_2, …等等表示"标号"，它们用来标记某个指令语句的号数；特别在计算进行中遇到标号为E 的指令时，意味着计算结束(因此我们对任何指令永不使用标号 E).

指令有三种类型，第一种是

$$\alpha \Longleftarrow \alpha + 1$$

其中 α 为任意变元. 它表示把变元 α 的值增加 1. 第二种是

$$\alpha \Leftarrow\!= \alpha \dot{-} 1$$

它表示当变元α的值不为0时把α的值减去1，否则的话α的值仍取0. 以上两种指令是计算指令. 第三种指令是

$$\text{To}\quad \beta\quad \text{If}\quad \alpha \neq 0$$

其中α为任意变元，而β为任意标号. 它表示当变元α的值不为0时去执行标号为β的指令；否则执行下条指令. 它是一种条件转移指令，指示下一步计算将要按那条指令进行. 它不是一条计算性质的指令.

以上三种指令称为基本指令.

理论上讲，有上述三种基本指令已经够了，但通常还有两种指令经常用到，它们的功能可由上述三种基本指令经适当组合后完成，故可称为导出指令或宏指令. 其中一种是所谓的赋值指令：

$$\alpha_1 \Leftarrow\!= \alpha_2$$

即把变元α_2的值赋给变元α_1，而α_2的值不变. 另一种是所谓无条件转移指令：

$$\text{To}\quad \beta$$

即要程序(无条件地)转去执行标号为β的指令.

程序就是由三种基本指令组合而成的一个有序(有限条)指令的集合. 一个程序刻划了一个计算. 程序的计算过程（或称为程序的执行过程)是对给定的初始输入数据从第一条指令开始顺序地执行程序中的指令，除非遇到转移指令. 当遇到转移指令时，则按转移指令中指出的指令执行，这个过程一直执行到转移标号为E的转移指令时为止. 程序执行结束时，变元Y的值就是该程序关于给定的初始输入数据的计算结果.

例 1　下列程序计算函数值 $Y=Z\cdot X$.

```
        To  C  If  X≠0
        Z₁ ⇐ Z₁+1
        To  E  If  Z₁≠0
[C]     To  A  If  X≠0
[B]     Y ⇐ Y+1
        Z ⇐ Z∸1
        To  B  If  Z≠0
        To  E  If  Y≠0
[A]     X ⇐ X∸1
        Y ⇐ Y+1
        Z ⇐ Z+1
        To  C  If  Z≠0
        Z₁ ⇐ Z₁+1
        TO  E  If  Z₁≠0
```

例 2　导出指令

$$\text{To}\quad \beta$$

可用基本指令如下实现:

```
Z ⇐ Z+1
To  β  If  Z≠0
```

例 3　导出指令

$$\alpha_1 \Longleftarrow \alpha_2$$

可用基本指令以及导出指令“To　β”如下实现:

$[A_1]$	To A_2 If $\alpha_1 \neq 0$
	To A_3
$[A_2]$	$\alpha_1 \Leftarrow \alpha_1 \dot{-} 1$
	To A_1
$[A_3]$	To A_4 If $\alpha_2 \neq 0$
	To A_5
$[A_4]$	$\alpha_2 \Leftarrow \alpha_2 \dot{-} 1$
	$Z \Leftarrow Z+1$
	To A_3
$[A_5]$	To A_6 If $Z \neq 0$
	To E
$[A_6]$	$Z \Leftarrow Z \dot{-} 1$
	$\alpha_1 \Leftarrow \alpha_1 + 1$
	$\alpha_2 \Leftarrow \alpha_2 + 1$
	To A_5

在上述程序中我们使用了无条件转移指令六次. 如果我们不怕麻烦, 它们完全可用相应的基本指令组代替, 这时唯一要注意的是若在相应的基本指令组中亦使用了标号和临时变元, 则在代替时必须改变这些标号和临时变元的名称以使它们不与"主"程序中的标号和临时变元的名称相混淆.

为了使程序的书写简短, 我们允许随意可引入"宏指令", 并在其它更复杂的程序中使用它们(今后, 我们可随时使用上述两条导出指令). 如果注意消去标号和临时变元名字的混乱, 使用宏指令的程序原则上都可恢复为只用三条基本指令的程序.

下面再举几个例子.

例 4 计算和的程序 $Y = X_1 + X_2$

```
        Y ⇐ X1
[A]     To B If X2≠0
        To E
[B]     X2 ⇐ X2−1
        Y ⇐ Y+1
        To A
```

例5 计算函数 $Y=X_1\cdot X_2$ 的程序

```
[A]     To B If X2≠0
        To E
[B]     X2 ⇐ X2−1
        Y ⇐ Y+X1
        To A
```

这里使用了加法宏指令 $Y\Leftarrow\!=Y+X_1$.

现在我们来形式地描述这种语言.

定义 字母 $X, X_i, Y, Z, Z_i(i=1,2,\cdots)$ 称为变元. 字母 $A, B, C, A_i, E(i=1,2,\cdots)$ 称为标号. 其中字母 X, X_i 称为输入变元; Z, Z_i 称为临时变元; Y 称为输出变元. 字母 E 称为出口标号.

定义 语句(或指令)是如下三种形式之一的符号串:

$$\alpha\Leftarrow\!=\alpha+1$$

$$\alpha\Leftarrow\!=\alpha-1$$

$$\text{To}\ \ \beta\ \ \text{If}\ \ \alpha\neq 0$$

其中 α 为任意变元, β 为任意标号.

上述的第一种指令称为加 1 指令, 第二种称为减一指令, 第三种称为非 0 (条件)转移指令. 每种语句前也可以有放在方括号

对“[ ]”中的标号. 在非 0 转移语句

.[A] To B If X≠0

中, B 称为此语句中的标号, X 称为此语句中的变元, 而 A 称为此语句的标号.

定义 一个程序(或计算)是一个语句的有限序列, 在其中不会有两个语句附有相同的标号(这个条件称为协调性条件).

注意, E 仅用作语句中的标号, 而不作语句的标号, 非 0 转移语句中的标号或为 E, 或为程序中出现的某语句的标号.

定义 一个程序 P 的计算状态是对程序中所用的每个变元用矢量等式的形式给出的当前数值. 程序开始于一个初始(计算)状态, 结束于一个结束(计算)状态. 从初始状态开始, 程序每执行一条指令, 就出现一个相应的新的状态. 因此给定一个有序对

〈P 的当前状态, P 的执行指令〉

可引出新的有序对

〈P 的新状态, P 的新执行指令〉.

定义 一个特定程序 P 的某个有序对〈P 的当前状态, P 的执行指令〉称为 P 的一个瞬时描述.

例子 考虑下列计算函数 $Y=X_1+X_2$ 的程序 P:

```
        Y ⇐ X1
[A]     To B If X2≠0
        Z ⇐ Z+1
        To E If Z≠0
[B]     Y ⇐ Y+1
        X2 ⇐ X2-1
        To A If Y≠0
        To E
```

令初始状态为

$$(X_1, X_2, Y, Z) = (3, 5, 0, 0)$$

则 P 的初始瞬时描述为

$$\langle (X_1, X_2, Y, Z) = (3, 5, 0, 0), Y \Longleftarrow X_1 \rangle,$$

它将引出一个新瞬时描述为

$$\langle (X_1, X_2, Y, Z) = (3, 5, 3, 0), \text{To B If } X_2 \neq 0 \rangle.$$

我们可严格定义它们如下:

定义 给定一程序 P. 设 Q_1, Q_2 为 P 的状态, S_1, S_2 为 P 的语句. 称瞬时描述$\langle Q_1, S_1 \rangle$转向(或产生)瞬时描述$\langle Q_2, S_2 \rangle$, 记为 $\langle Q_1, S_1 \rangle \rightarrow \langle Q_2, S_2 \rangle$, 如果有下列三种情况之一发生:

情况 1: S_1 为加 1 指令, S_2 在 P 中紧跟 S_1 之后, Q_2 是把Q_1中的关于 S_1 中的变元加 1 而得.

情况 2: S_1 为减 1 指令, S_2 在 P 中紧跟 S_1 之后, Q_2 是把 Q_1 中的关于 S_1 中的变元算术减 1 (即当 S_1 变元不为 0 时减 1, 为 0 时仍为 0)而得.

情况 3: S_1 为非 0 转移指令, Q_1 与 Q_2 相同, 且当 S_1 中的变元在 Q_1 中非零时, S_2 为以 S_1 中的标号为标号的语句, 而当 S_1 中的变元在 Q_1 中为零时, S_2 在 P 中紧跟 S_1 之后.

定义 瞬时描述$\langle Q, S \rangle$称为程序 P 的终止瞬时描述, 如果不存在被$\langle Q, S \rangle$产生的瞬时描述.

易见, 只有当 S 中的标号为 E, 而 S 中的变元在 Q 中不为零时, 则$\langle Q, S \rangle$为 P 的终止瞬时描述.

显然, 每个程序P 对某个瞬时描述只产生一个瞬时描述(若有的话). 因此若

$$\langle Q_1, S_1 \rangle \rightarrow \langle Q_2, S_2 \rangle$$
$$\langle Q_1, S_1 \rangle \rightarrow \langle Q_3, S_3 \rangle$$

则

$$S_2 = S_3, Q_2 = Q_3.$$

定义 程序 P 称为是 n 元的，如果 P 中恰好含有 n 个输入变元 $X_1, \cdots, X_n$. 当 n＝1 时，输入变元常用X表示.

设 P 为 n 元，输入变元为 $X_1, \cdots, X_n$，输出变元为Y，临时变元为 Z_j，则下列一个矢量等式称为 P 的初始状态

$$(X_1, \cdots, X_n, Y, Z_j) = (a_1, \cdots, a_n, 0, 0)$$

其中各 a_i 为自然数.

有序对$\langle Q_1, S_1\rangle$称为 P 的初始瞬时描述，若 S_1 为 P 的第一个语句，Q_1 为 P 的初始状态.

定义 程序 P 的一个计算是有限个瞬时描述的序列:

$$\langle Q_1, S_1\rangle \to \langle Q_2, S_2\rangle \to \cdots \to \langle Q_n, S_n\rangle,$$

简记为

$$\langle Q_1, S_1\rangle \Rightarrow \langle Q_n, S_n\rangle$$

其中 $\langle Q_1, S_1\rangle$为初始瞬时描述，而$\langle Q_n, S_n\rangle$为终止瞬时描述.

设 P 为一个 n 元程序，则令 P 对应下列一个 n 元函数 $\Psi_P(x_1, \cdots, x_n)$，称为 P 的对应函数:

对于每一组自变量值$(a_1, \cdots, a_n)$，取初始状态 Q_1 为

$$(X_1, \cdots, X_n, Y, Z_j) = (a_1, \cdots, a_n, 0, 0)$$

把 P 作用于初始瞬时描述$\langle Q_1, S_1\rangle$(S_1 为 P 的第一个语句)，这时(i)若存在终止瞬时描述$\langle Q_m, S_m\rangle$，使得$\langle Q_1, S_1\rangle \Rightarrow \langle Q_m, S_m\rangle$，则 Q_m 中变元Y的值取为函数值 $\Psi_P(a_1, \cdots, a_n)$；(ii)若不存在终止瞬时描述，则 $\Psi_P(a_1, \cdots, a_n)$无定义.

这样，每个程序 P 可定义一个函数 Ψ_P. 但每个函数是否有计算它的程序呢?

定义 函数 $f(x_1, \cdots, x_n)$称为(程序)部分可计算的，如果有一

个程序 P, 使得

$$\Psi_P(x_1, \cdots, x_n) \approx f(x_1, \cdots, x_n).$$

其中"≈"指或者当一边无定义时另一边亦无定义；或者当一边有定义时，另一边亦有定义，且值相等.

定义 函数 $f(x_1, \cdots, x_n)$称为全函数，如果它对任意 $x_1, \cdots, x_n$都有定义.

定义 若一个全函数是(程序)部分可计算函数，则该函数称为(程序)可计算函数.

于是一个(程序)可计算函数除了有一个计算它的程序外，还处处有定义.

由定义不难看出，一个处处无定义的一元函数是(程序)部分可计算的. 这是因为可用下列程序 P计算它:

```
[A]    X ⇐ X+1
       To A
```

甚至处处无定义的任意 n 元函数是(程序)部分可计算的.

```
[A]    X1 ⇐ X1+1
       ⋮
       Xn ⇐ Xn+1
       To A
```

下面我们讨论原始递归函数，一般递归函数，幂状函数等怎样用"程序"来计算.

引理 1 本原函数是程序可计算的.

[证] (1) 后继函数 Y=SX 是全的，且可用下列程序计算:

```
X ⇐ X+1
Y ⇐ X
To E If X_i≠0
```

故SX是程序可计算的.

(2) 零函数 $Y=O(X)$是全的,且可用下列程序计算(因Y的初值为零):

```
X ⇐ X+1
To E If X≠0
```

(3) 广义么函数 $Y=I_{mn}(X_1, \cdots, X_m)$是全的，且可用下列程序计算:

```
X_1 ⇐ X_1+1
⋮
X_{n-1} ⇐ X_{n-1}+1
Y ⇐ X_n
X_{n+1} ⇐ X_{n+1}+1
⋮
X_m ⇐ X_m+1
To E
```

引理 2 设函数 $f(Z_1, \cdots, Z_m)$和 $g_1(X_1, \cdots, X_n), \cdots, g_m(X_1, \cdots, X_n)$程序可计算(程序部分可计算)，则由它们经(m, n)迭置所得的函数:

$$h(X_1, \cdots, X_n)=f(g_1(X_1, \cdots, X_n), \cdots, g_m(X_1, \cdots, X_n))$$

亦程序可计算(程序部分可计算).

[证] 设计算函数 $f, g_1, \cdots, g_m$ 的程序分别缩写为

$$Y \Leftarrow f(X_1, \cdots, X_m)$$
$$Y \Leftarrow g_1(X_1, \cdots, X_n)$$
$$\vdots$$
$$Y \Leftarrow g_m(X_1, \cdots, X_n)$$

则计算函数 h 的程序是

```
Z1 ⇐ g1 (X1, ···, Xn)
⋮
Zm ⇐ gm (X1, ···, Xn)
Y  ⇐ f  (Z1, ···, Zm)
To E
```

其中每个"宏指令"(共有 m+1 个)都表示计算相应函数的程序，但对其中的输入变元，输出变元，临时变元和标号等都必须作相应的调整，以避免混乱.

因此函数 h 是程序部分可计算的. 又当 f 和各 g_i 为程序可计算时，它们是全函数，这时 h 亦为全函数，从而 h 亦程序可计算.

引理 3 若 $f(X_1, \cdots, X_n)$, $g(X_1, \cdots, X_n, X_{n+1}, X_{n+2})$ 程序可计算，则由下列原始递归式定义的函数 $h(X_1, \cdots, X_n, X_{n+1})$ 亦程序可计算:

$$\begin{cases} h(X_1, \cdots, X_n, 0) = f(X_1, \cdots, X_n) \\ h(X_1, \cdots, X_n, X_{n+1}+1) \\ \quad = g(X_1, \cdots, X_n, X_{n+1}, h(X_1, \cdots, X_n, X_{n+1})) \end{cases}$$

[证] 设计算 f, g 的程序分别缩写为

$$Y \Leftarrow f(X_1, \cdots, X_n)$$
$$Y \Leftarrow g(X_1, \cdots, X_n, X_{n+1}, X_{n+2})$$

则计算 h 的程序为

```
        Y ⇐ f (X1, ⋯, Xn)
[A]     To B If Xn+1≠0
        To E
[B]     Y⇐g(X1, ⋯, Xn, Z, Y)
        Z ⇐ Z+1
        Xn+1 ⇐ Xn+1∸1
        To A
```

当 n=0 时，上述引理 3 可改述为：若 $g(X_1, X_2)$程序可计算，则由下列原始递归式定义的函数 h(X)亦程序可计算：

$$\begin{cases} h(0)=K \\ h(X+1)=g(X, h(X)) \end{cases}$$

其中K 为任意自然数．这时计算 h的程序可为

```
        Y ⇐ Y+1
        ⋮          } K次
        Y ⇐ Y+1
[A]     To B If X≠0
        To E
[B]     Y ⇐ g(Z, Y)
        Z ⇐ Z+1
        X ⇐ X∸1
        To A
```

引理 4　若 $f(X_1, \cdots, X_n, X_{n+1})$程序可计算，则函数

$$h(X_1, \cdots, X_n)=rti_t\{f(X_1, \cdots, X_n, t)\}$$

也程序部分可计算．

[证]　设计算 $f(X_1, \cdots, X_n, X_{n+1})$的程序简记为

$$Y \Leftarrow f(X_1, \cdots, X_n, X_{n+1})$$

则计算h的程序为

```
[A]  Z⇐f(X1, ⋯, Xn, Y)
     To B If Z≠0
     To E
[B]  Y ⇐ Y+1
     To A
```

在上述程序中，若有某个Y使得Z得到零值，则计算停止；否则的话，计算一直继续下去，从而函数h是部分可计算的.

显然当幕状算子"rti"为正常算子(即 $\mu_t(f(X_1, \cdots, X_n, t)=0)$ 恒存在)时，上述函数h是程序可计算的(这是因为此时的h必为全函数).

由上面几个引理即可得:

定理 每个一般递归函数是程序可计算的. 每个部分递归函数是程序部分可计算的.

[证] 由§3.8知，正常幕状函数集与一般递归函数集等价，因此一般递归函数亦可看作由本原函数以及初始函数 $x+y$, $x\cdot y$, $eq(x, y)$出发经有限次使用(m, n)迭置以及正常幕状算子而得，但所述三个初始函数都是原始递归函数，故一般递归函数亦可看作由本原函数出发经有限次使用(m, n)迭置、原始递归算子以及正常幕状算子而得. 于是由上述几个引理知，一般递归函数是程序可计算的. 又部分递归函数可看作由本原函数出发经有限次使用迭置，原始递归算子以及不加限制(即不要求 $\mu_t(f(x_1, \cdots, x_n, t)=0)$ 恒存在)的幕状算子而得，从而部分递归函数是程序部分可计算的. 于是定理得证.

关于这个定理的逆，即每个程序可计算的函数是一般递归的(和每个程序部分可计算的函数是部分递归的)，实质上是§3.8中

的定理 4，这里不再进一步详细论述了。

习 题

1. 求出对应于下列运算的数论函数：

1.1 将一数的数字颠倒次序（例如把 2437 变成 7342）；

1.2 将两数的数字从左端开始依次交错地排成一数（例如把 459 与 1782 合并变成 4157982）。

2. 试用三条基本指令写出计算下列函数的程序：

2.1 $Y=X_1+X_2+X_3$；

2.2 $Y=X_1\cdot(X_2+X_3)$；

2.3 $Y=X_1\cdot X_2+X_3\cdot X_4$；

2.4 $Y=\begin{cases}X_1+X_2, & \text{当 } X_3\neq 0 \text{ 时；}\\ X_1\cdot X_2, & \text{当 } X_3=0 \text{ 时。}\end{cases}$