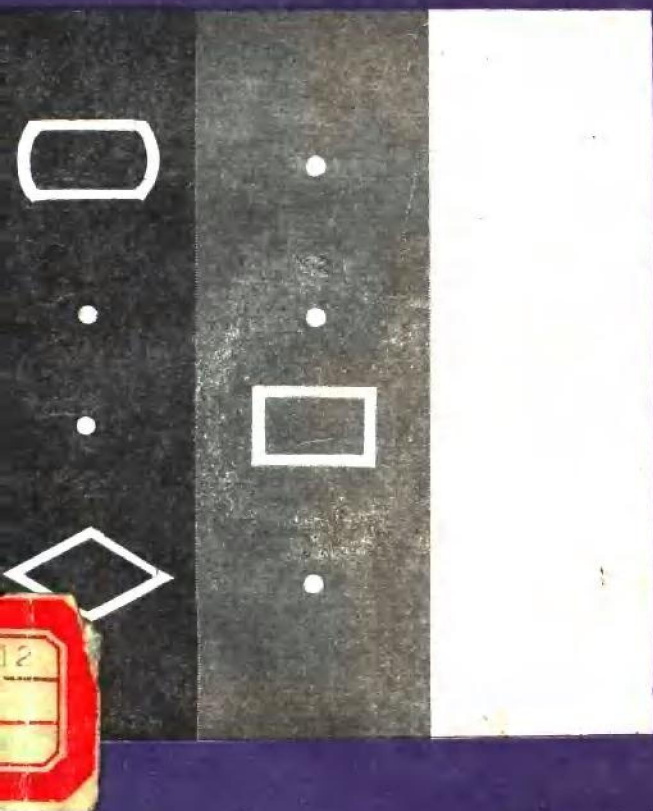


计算机基础自学丛书

数据结构

严蔚敏 陈文博 编著

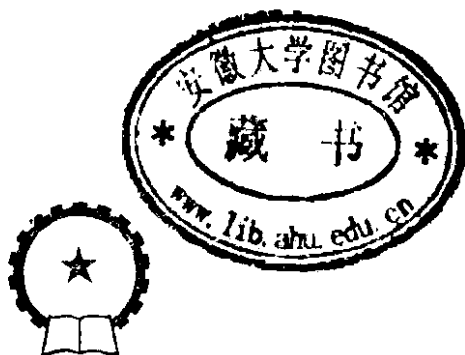


机械工业出版社

计算机基础自学丛书

数 据 结 构

严蔚敏 陈文博 编著



机械工业出版社

本书介绍常用数据结构的特性、存贮表示方法和有关的实用算法。全书共九章。主要内容包括：数据结构的基本概念、算法的描述工具和初步的分析方法；线性表、栈、队、树、图等结构；查找方法、排序方法和文件组织方法；最后介绍了数据结构在实际问题中的使用方法和综合运用的若干程序实例。

本书可作为具有中专以上文化程度、机电专业及计算机应用专业的科技工作者的自学用书，也可作为大专院校相应专业教科书或培训班教材。

数 据 结 构

严蔚敏 陈文博 编著

责任编辑：邱锦来 责任校对：孙志筠
封面设计：方 勇 版式设计：张世琴
责任印制：张俊民

机械工业出版社（北京阜成门外百万庄大街一号）
（北京市新闻出版局登记证出字第117号）

中国农业机械出版社印刷厂印刷

新华书店北京发行所发行·新华书店经售

开本 850×1168¹/₃₂·印张 11·字数 286 千字
1990年2月北京第一版·1990年2月北京第一次印刷
印数 00,001—2,910·定价：11.10 元

ISBN 7-111-01723-4/TP·100

出版说明

计算机的广泛应用极大地提高了人类认识和改造客观世界的能力,特别是微电子及大规模集成电路技术的发展,微型计算机的广泛应用为各国工业技术的改造提供了重要而有效的途径。一场以解放人类智力为标志的新技术革命开始了。其规模和深度都将是空前的,不仅涉及到科学、工业、经济、教育等各个领域,同时也将会引起社会结构和生活方式的改变。

预计90年代将出现以知识处理为基础的智能技术与智能系统,并得到广泛的应用。例如,智能机器人、智能信息处理系统、智能计算机辅助设计系统、智能办公室自动化系统等。它将开创新的计算机应用领域。传统的数据处理、计算机控制等应用更加向前推进。

为了适应新技术革命的需要,帮助各个领域的科技人员进一步普及和提高计算机基础科学知识,使计算机得到广泛的更有效的应用,我们特为全国机电专业及计算机应用专业中专以上文化水平的读者编写了这一套丛书。广大读者只要具有基本的数字电路及逻辑设计基础知识,均可自学这套丛书。“丛书”着眼于计算机的应用,是在搜集归纳了国内外大量的资料,并在长期的教学和科研工作基础上进行编写的。编写时力求做到选材上的先进性、系统性和完整性。书中由浅入深全面系统地阐述了计算机科学的基础知识,同时对计算机的主要应用领域的先进技术也作了必要的介绍。

“丛书”编写者是清华大学计算机科学与技术系长期在一线从事教学、科研工作的副教授、高级工程师。全书由清华大学周远清主编,叶乃萃为全套丛书的编写、定稿和出版作了大量的工作。

这套丛书共分九册。

第一册为计算机概论，阐述计算机的体系结构及各部分工作原理。

第二册为高级语言程序设计，介绍PASCAL语言及C语言的基本结构以及语言的使用。

第三册为PDP-11机器语言与汇编语言程序设计，介绍PDP-11机的基本组成及编程技术。

第四册为数据结构，阐述非数值程序中有关结构以及排序和查找等问题。

第五册为操作系统，阐述操作系统的有关基本概念以及管理系统资源，提供用户应用环境。

第六册为汉字微型计算机与汉字识别，介绍汉字微型计算机系统的组成、输入编码方案、汉字处理和识别技术以及汉字微型计算机系统的应用。

第七册为计算机控制系统及应用，主要分析计算机控制系统的原理、设计方法（连续、离散、状态空间），通道接口及应用实例。

第八册为计算机辅助图形设计，介绍计算机辅助设计基础——二维三维图形设计的基本算法以及交互式图形软件的设计。

第九册为办公室自动化基础，阐述数据库的基本概念、办公室自动化基础知识及技术。

在编写这套丛书时，我们既注意到它的系统性和完整性，同时又使各册之间有相对的独立性。便于读者能够根据自己从事工作的领域以及自己的兴趣，阅读全套丛书或选读其中几册。

对于需要使用语言进行程序设计的初学者，可以首先阅读第一、二、三册。在有关领域中从事计算机应用的科技人员，可以分别选读第六、七、八、九册。如果需要系统深入了解和掌握计算机科学知识的读者，建议进一步阅读第四、五册。

书中所选实例均经过上机验证，有的还附有习题，希望读者能完成练习，并上机进行实践。

▼

这套丛书既可作广大读者的普及读物和科技参考书，又可作为大学教科书。我们希望，这套丛书能为我国计算机的普及与应用起到一定的作用。

由于编者水平有限，书中定有不当之处和错误的地方，欢迎读者批评指正。

清华大学计算机科学与技术系
计算机基础自学丛书编写组

前 言

随着计算机在各行各业中的普及,《数据结构》逐渐成为人们注目的课目。虽然,任何一个使用计算机的技术人员,在了解自己本专业的知识基础上,又掌握了某高级语言,也能编制出程序上机调试。但在学习了数据结构之后,可以有更明确的算法思想来指导编程,使程序的效率更高,结构更合理,并减少出错机会,以提高程序的可读性。

数据结构讨论的对象是非数值性的数据以及它们之间的关系。根据这些关系的不同数学特性,可分为线性结构(包括线性表、栈和队列)和非线性结构(包括二叉树、树和图),本书将分别讨论它们的逻辑结构和存贮结构。查找和排序是文献检索、数据管理和档案管理等问题中经常遇到的两种运算,它们也是数据结构讨论的主要对象。在当前数据库流行的同时,文件系统仍是一种有效的常用数据管理手段,而且它是学习数据库的基础,因此我们也为文件单列一章进行讨论。由于数组几乎是所有语言工具都已具备的数据类型;很多高级语言都扩充了字符串及其运算的功能,本书将不对数组和字符串结构进行讨论。广义表是一种较为重要的数据结构,但从初学者的实际应用机会考虑,我们没有把广义表的内容列入。这样安排的目的是为了突出常用数据结构的应用效果。

学习本书要求读者具有一般程序设计的能力和PASCAL语言的知识,如果具有离散数学和概率论的初步知识,将更有助于对本书的学习和内容的掌握。

本书强调数据结构在软件工程中的应用,在内容组织上分为两大部分:前八章为数据结构的基本内容,每章都有丰富的例题讲解;第九章列举了数据结构的各种应用实例,目的在于提高读者分析处理综合问题的能力。为了便于读者学习,我们在文字叙述

上尽量做到由浅入深,通俗易懂,对各种基本运算给以详尽描述,并写出算法,对有的算法还加以跟踪演示。对一些较难理解的运算,如二叉树的遍历等还列举了多个应用例子。除第九章外,所有算法都用拟PASCAL语言描述,算法中的标识符都用大写字母表示,但同一变量名在文字叙述或算法功能说明中出现时,除数组名外,均用小写字母表示,其含义不变。

本书各章均由严蔚敏和陈文博共同编写,全书由周远清审阅。

本书在编写过程中还得到了北京计算机学院HP机房同志的大力支持,在此表示衷心感谢。

由于水平所限,加之时间仓促,书中难免存在缺点和疏漏,欢迎广大读者予以批评指正。

作 者

第一章 绪 论

§ 1-1 学习数据结构的意义

数据结构是计算机和计算机应用专业的重要课程之一，现在又成为其它理工科专业学生的“热门”选修课。事实说明，要想有效使用计算机，仅掌握计算机语言而缺乏数据结构和算法的有关知识，难以应付众多复杂的应用课题。为此，让我们简单回顾一下数据结构学科形成的历史。

在电子计算机发展的初期阶段，人们使用计算机主要是处理数值性问题，解决人们用手工或机械计算机难于胜任的数值计算。当时所涉及的操作对象还都比较简单，不外乎是整型、实型和布尔型数据。以此为加工对象的程序设计可称为数值型程序设计，对应的软件称“数学软件”。随着电子计算机使用领域的扩大和深入，解决“非数值性问题”越来越引起人们的关注。例如文献检索、金融系统和商业系统的数据处理、档案管理以及以图论为基础的各种应用问题和图象的模式识别等等。解决此类问题使用的数学工具已不是分析数学及其计算方法，而更多地用到离散数学的知识，所涉及的数据也更复杂，其突出的特点是，数据元素之间所具有的特定联系不能用分析数学的方程式来描述。

为了帮助读者建立对数据结构的感性认识，下面先举几个简单例子。

例1-1 识别手写体汉字问题。

计算机识别一个手写体的汉字时，一般采用分层次的办法，自顶向下进行。先用汉字的某些局部特征与识别主体相匹配，尽量容忍畸变和干扰，以逐步缩小识别范围直至最后确认为止。如识别“体”字时，先判部首，“体”属于“单人旁”，因部首为“单

人”的汉字还很多,故尚需按四角号码进行匹配,相同的只有“体”和“休”二字,最后再进行局部的细节匹配,被确认为“体”字。上述识别过程可用图1-1所示的“树”来描述。

象这样按分支和层次组织的数据,称为“树形结构”。识别“体”字的过程是对图1-1所示的树进行某种操作或运算。

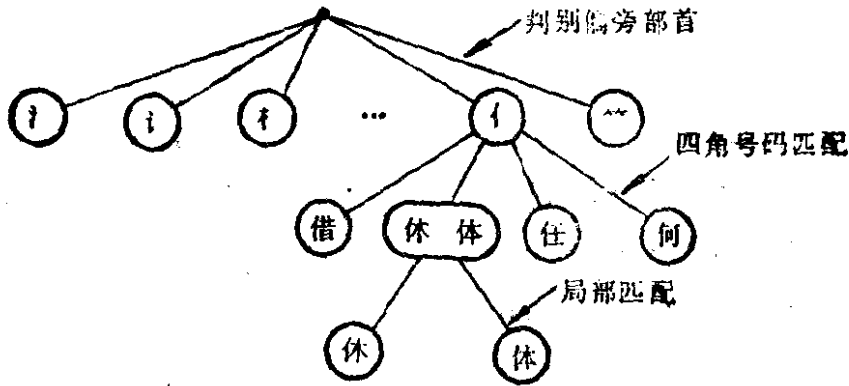


图1-1 手写体汉字的识别过程

例1-2 计算机换房系统中的“多角互换问题”。

在登记的换房户中,如果乙所具条件恰好满足甲的要求,而甲所具条件恰好满足乙的要求,则称甲乙两户为一对可行的“对换对”。这种关系可用图 1-2a 表示,图中从甲到乙的一条弧表示甲可以换到乙(即乙的条件满足甲的要求),反之从乙到甲的一条弧表示乙可以换到甲。但实际情况往往是,在两户之间很难凑成彼此都满意的“对换对”,如果多考虑几户,则有可能找到多户之间的一个“换房环”,即甲可以换到乙,乙可以换到丙,丙可以换到丁,而丁可以换到甲。这在换房的行话中叫做“多角互换”。图1-2中所示的图就是描述上述“对换对”和“多角互换”的数据结构。

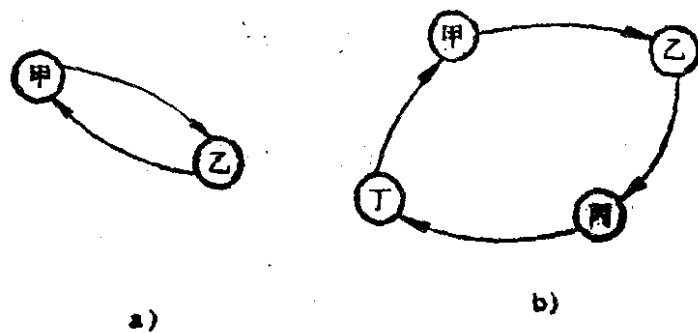


图1-2 换房系统中的数据结构模型

a) 对换对 b) 多角互换

由所有表示换房户的顶点和表示换房户之间关系（条件满足与否）的有向弧构成的有向图称做“换房网络”，如图1-3所示。计算机换房系统就是对这个网络进行操作，试图在这个网络中寻找一个有向环，为换房户提供“多角互换”的依据。这种有向图也是数据结构中讨论的一类非线性结构。

例1-3 饭店服务系统中的客房预订问题。

客房预订是饭店计算机系统的内容之一，它根据预订日期的先后将预订客房的客户排成一个队列，队列中的每个元素包括客户姓名、开始使用日期和房间使用要求等信息，如图1-4所示。计算机系统

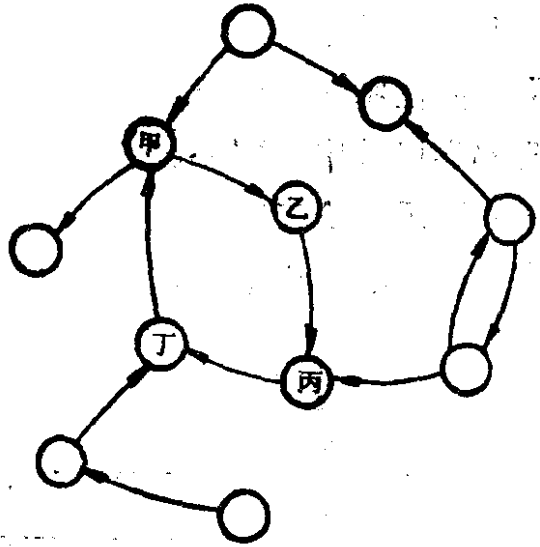


图1-3 在换房网络中通过有向环找多角互换

将定期扫视这个队列，打印临近预订期客户的有关信息，为服务人员提供事先准备客房的有效信息。计算机系统还可以为撤消预订的客户服务，即从队列中删除该客户的预订信息。这种队列就是一种元素间先后次序性很强的数据结构，扫视查询、预订客房和撤消预订都是对队列进行某种操作。

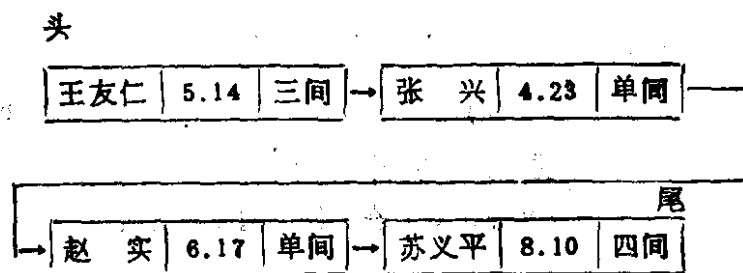


图1-4 房间预订系统中使用的队列结构

例1-4 管理系统中的查询问题。

在各种计算机的管理系统中，相关的信息（记录）组织成一个文件，查询就是对文件进行检索操作。如学籍管理系统中的学

生选修课文件记载着每个学生选修课的情况，如图1-5所示。当教务部门要查询选修“算法分析”课的同学时，需对文件进行检索。文件也是一类重要的数据结构，文件中的记录可如图1-5a所示按顺序方式组织。在有些数据量很大或时间性很强的系统中，为了提高检索效率，还需要对文件中经常被查询的数据项建立一种联系。如在图1-5a的文件中添加一项，将所有选修“算法分析”课的同学记录串接到一起，这种串接称为“加链”，加链的一项如图1-5b所示。检索时，只要从链头开始顺链便可找到全部选修“算法分析”的学生记录。

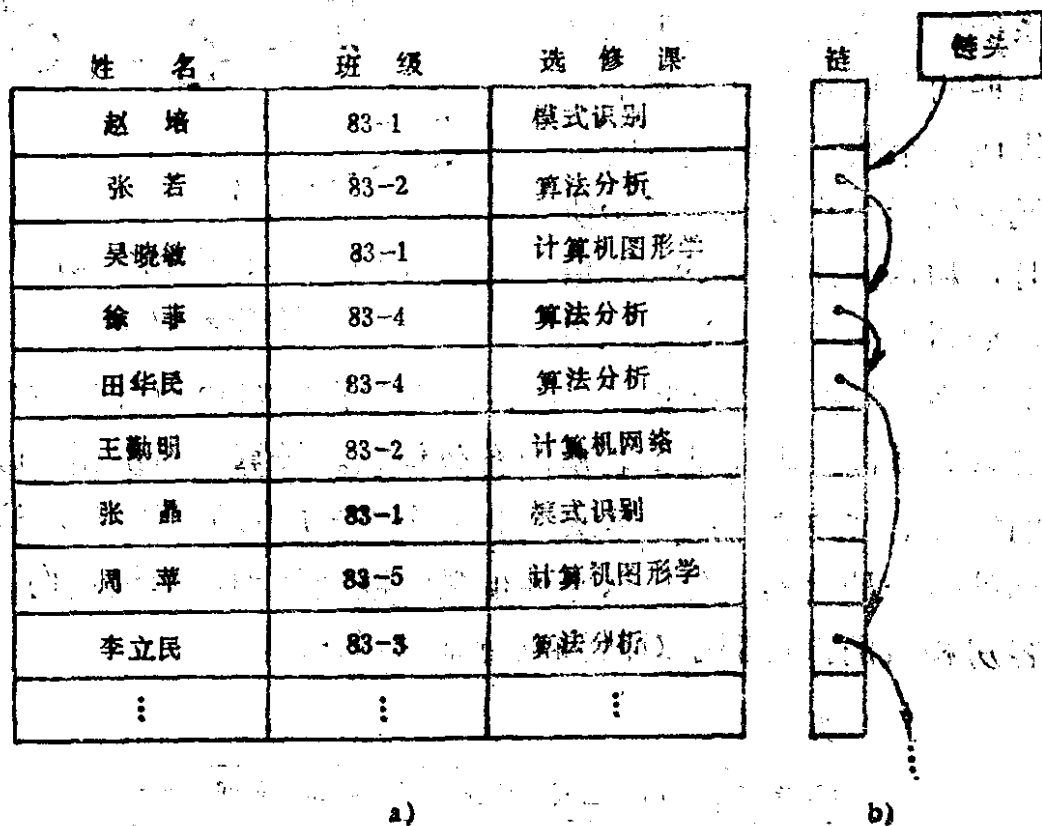


图1-5 文件结构示例

a) 顺序文件 b) 在文件中增加的链项

综合以上几个例子不难看出，大量的实际问题中涉及的数据，元素之间存在着非数值性的关系。我们称这种关系为“结构”，目前，多数高级语言尚不能直接操作这些带结构的数据。本书将在高级语言的层次上讨论如何表示这些数据和如何进行对它们的操作。

§ 1-2 有关数据结构的基本概念和术语

为今后学习方便，这一节先介绍有关数据结构的基本概念和术语^①。

数据 (Data) 数据是信息的载体，它可以在机内表示並加工处理。

数据元素 (Data Element) 数据元素是数据的基本单位。有时一个数据元素也可以由若干数据项组成。例如，在计算机的换房系统中，一个登记户的全部信息（包括登记号、姓名、地址、住房条件和换房要求）组成一个数据元素，其中每项信息称数据项。通常，数据元素在除文件以外的所有数据结构中是操作的最小对象。

数据对象 (Data Object) 具有相同性质的数据元素的集合称为数据对象。任何实际问题中讨论的数据元素都是具有相同性质的，因此属于同一数据对象。例如在计算机换房系统中的数据元素都属于“登记换房的居民户”这个数据对象。

结构 (Structure) 结构是指同一数据对象中各数据元素之间存在的关系。如客房预订系统中的登记次序关系，换房系统中的条件满足关系等等。这种关系是以问题为背景的一种特定联系。

数据结构 (Data Structure) 简单地说，数据结构是带结构的数据对象，它包括一个（具有共同特性的）数据元素的集合和定义在这个集合上的一组关系。从 § 1-1 所举例子可见，对每个具体问题而言，其数据元素和关系均具有特定的含义，但对数据结构学科而言，待研究的不是这些具体的含义，而是它们所具有的数学特性。根据不同的结构特性，本书讨论的数据结构有：表或文件、线性表（包括栈和队列）、树和图。表中的元素具有松散的集合关系，或者说，元素之间除了同属某个集合外，不存在任何关系；线性表中的元素之间存在着一对一的线性关系；树的元素

^① 数据结构是一门年青的学科，并且仍在发展之中。因此，有关数据结构的各种术语在计算机科学界尚未有统一的定义，这里所述仅是当前通行的一种说法。

之间存在着一对多的分支或层次关系；图的元素之间存在着多对多的网络关系。数据结构包括数据的逻辑结构和数据的物理结构。数据的逻辑结构研究的是数据元素及其关系的数学特性，而数据的物理结构研究的是它们在计算机内的实现。对机器语言来说，这种实现是具体的，本书仅在高级语言的层次上来讨论这种实现，需用高级语言中的数据类型来描述这种实现。为简明起见，以后我们将简称数据的逻辑结构为数据结构，简称数据的物理结构为存贮结构。

数据类型 (Data Type) 数据类型指的是程序设计语言中允许的变量种类。在用程序设计语言书写的程序中出现的每个变量，必须与一个、而且仅仅与一个数据类型相联系，它不仅规定了该变量可以设定的值的集合，还规定了在这个集合上的一组运算。每一种程序设计语言中都有一组它所允许的基本数据类型，不同的语言提供的基本数据类型不同。如FORTRAN语言提供了五种基本数据类型：整数、实数、双精度数、复数和布尔量，PASCAL语言提供了整型、实型、布尔型和字符型四种基本类型。除了基本类型之外，程序设计语言中还提供结构的数据类型（简称结构类型），如数组等。基本类型变量的值是不可分的，而结构类型变量的值是可分的，或者说它是带结构的数据。因此，结构类型也可看成是语言中已经实现的数据结构。

算法 (Algorithm) 一个算法，就是解决某一特定类型问题的有限运算序列。显然，一个算法必须是在执行有限步之后结束；算法中的每一步必须有确切定义，并且每一步有待实现的运算都是相当基本的，换句话说，它们原则上都是能够精确地进行的，一个算法有0个或多个的输入，也必然有一个或多个的输出。算法的设计可以避开具体的计算机和程序设计语言，但算法的实现必须借助程序设计语言中提供的数据类型及其运算。讨论算法是数据结构课程中的重要内容之一，著名瑞士计算机科学家沃斯(N. Wirth) 提出一个精辟论点：算法+数据结构=程序。可见算法和数据结构是利用计算机解决实际问题时不可缺少的两个方

面。因此，本书在讨论数据结构的同时，也将介绍若干和数据结构有关的算法。

§ 1-3 算法的描述工具——拟PASCAL语言

在当前各种版本的数据结构的书中，描述算法的方式很多，本书采纳当前比较通行的拟PASCAL语言。拟PASCAL语言在可读性和严密性方面比早期使用的自然语言、框图等要优越得多。这一方面因为PASCAL语言本身拥有丰富的数据类型和控制结构，书写的程序结构性强；另一方面拟PASCAL语言不拘泥于某些细节规定，对PASCAL语言进行了技术上的简化处理，以便更方便地编制算法和增加算法的可读性。

简化处理表现在两个方面。第一，省略了形式化的类型说明和变量说明，代之以用较为自由的自然语言作非形式化的描述，且省略说明一些容易理解的整型变量；第二，增加了一些语句的功能，如扩充了CASE语句的范围，增添了出错处理语句ERROR，将语句组的一对括弧BEGIN-END改为一对黑体方括弧【】等等。如读者需对某些算法进行上机调试，只须稍加处理便可将用拟PASCAL语言书写的算法转换成其它语言编写的程序。

本书中采用的拟PASCAL语言的非形式化说明如下：

(1) 语言中允许整型、实型、字符型、布尔型等基本类型，还允许数组、指针等结构类型。数组可用 $A(L_1 : H_1, L_2 : H_2, \dots, L_n : H_n)$ 表示，其中A是数组名称，n是维数， L_i 和 H_i 分别是第i维的下界和上界， $i = 1, 2, \dots, n$ 。

(2) 表达式可以是算术表达式、字符表达式和布尔表达式。在布尔表达式中允许使用关系运算符 $<$ 、 \leq 、 $=$ 、 \geq 、 $>$ 、 \neq 和逻辑运算符AND、OR和NOT。

(3) 赋值语句的形式为 $\langle \text{变量} \rangle := \langle \text{表达式} \rangle$ 。语句间用“;”隔开，若干语句构成的语句组用一对括弧“【”和“】”相括。

(4) 条件语句有下列两种格式：

IF C THEN S

IF C THEN S₁ ELSE S₂

其中 C 表示条件 (布尔表达式), S 表示语句或语句组 (以下同)。

(5) 情况语句也有两种格式:

CASE <表达式> OF	CASE
<常量 1> : S ₁ ;	C ₁ : S ₁ ;
<常量 2> : S ₂ ;	C ₂ : S ₂ ;
:	:
<常量 n> : S _n ;	C _n : S _n ;
END	ELSE S _{n+1} ;
	END

(6) 循环语句有下列四种格式:

“当”语句:

WHILE C DO S

“重复语句:

REPEAT S₁; S₂; ... S_n UNTIL C

“步进循环”语句:

FOR 变量 := 初值 TO 终值 DO S

或

FOR 变量 := 初值 DOWNTO 终值
DO S

前者的工作条件是初值 ≤ 终值; 后者的工作条件是终值 ≤ 初值, 步进循环的步长分别为 1 和 -1。

(7) 过程调用语句 CALL 过程名 (参量表) 过程允许递归调用和嵌套调用。

(8) 出错处理语句 ERROR (字符串)

在过程中出现异常情况时使用。在过程执行中, 遇到 ERROR 语句时提前结束过程的执行, 直接转移至过程结束处, 并可通通过字符串带回相应的出错信息。

(9) 读语句 READ (变量或表达式或字符串);

写语句 WRITE (变量或表达式或字符串)。

(10) 注解{字符串}可在算法中任何位置上出现, 等价于一个空格, 不予执行, 只是为了帮助读者阅读而插入程序的。

(11) 所有的算法都以如下所示过程或函数的形式出现:

PROCEDURE 过程名 (形式参量表);

BEGIN

〈语句组〉

END; {过程名}

FUNCTION 函数名 (形式参量表);

BEGIN

〈语句组〉

RETURN (函数值)

END; {函数名}

形式参数表中的参数不加类型说明, 但在每个变参之前冠以VAR。凡是需将过程加工中的结果传递到过程外的参数必须是变参。PASCAL语言中约定, 结构类型的参数通常设为变量参数, 此约定对拟PASCAL语言也适用。

对语言中允许的字符集和标识符都不加任何限制。

下面举几个例子介绍拟PASCAL语言的使用。

例1-5 拉丁方游戏。在 $n \times n$ 的方阵中每一格都填上 1, 2, ..., n 中的一个数, 使每行、每列中的数均不同。如 $n = 5$

1	2	3	4	5
2	3	4	5	1
3	4	5	1	2
4	5	1	2	3
5	1	2	3	4

5 x 5

a)

1	2	3	4	5	1
2	3	4	5	1	2
3	4	5	1	2	3
4	5	1	2	3	4
5	1	2	3	4	5

b)

图1-6 拉丁方游戏图

a) 5 x 5 的拉丁方 b) 算法模型