

计·算·机·等·级·考·试·丛·书

98 大纲

谭浩强 主编

郑启华 编著

PASCAL 程序设计(二级)教程

- 特 邀计算机教育专家编写
- 有 配套辅导用书和样题汇编
- 用 于考试前系统的学习与培训

计算机等级考试
从书考机



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



计算机等级考试丛书('98 大纲)

谭浩强 主编

PASCAL 程序设计(二级)教程

郑启华 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书是作者根据教育部考试中心 1998 年制定的《全国计算机等级考试 考试大纲》并在长期教学经验的基础上编写的。本书全面介绍了 PASCAL 语言,并讲述了自顶向下逐步求精的结构化程序设计的基本思想和方法。全书共分 12 章,内容涵盖了计算机等级考试所要求的范围,每章之后都指出了一些常见的错误,并给出了小结,还附有精选的练习题。

本书不仅可以作为参加计算机等级考试人员的自学或培训教材,也可以作为大专院校 PASCAL 语言程序设计课程的教学用书,还可以作为工程技术人员的自学参考书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: PASCAL 程序设计(二级)教程
作 者: 郑启华
出 版 者: 清华大学出版社(北京清华大学学研楼,邮编 100084)
<http://www.tup.tsinghua.edu.cn>
印 刷 者: 北京市人民文学印刷厂
发 行 者: 新华书店总店北京发行所
开 本: 787×1092 1/16 印张:22.25 字数:525 千字
版 次: 2000 年 3 月第 1 版 2000 年 5 月第 2 次印刷
书 号: ISBN 7-302-03827-9/TP·2235
印 数: 5001~11000
定 价: 26.50 元

序

进入 20 世纪 90 年代以来,我国掀起了第二次计算机普及高潮。人们已经认识到,要建设一个现代国家,离开计算机将寸步难行。无论从事什么工作,都必须学习计算机知识,掌握计算机应用。不少单位已经把通过计算机考试作为任职条件。

为了适应社会的需要,原国家教委考试中心于 1994 年推出了“全国计算机等级考试”,受到社会各界的热烈欢迎。至 1998 年底,已有 160 多万人报名考试,其中 70 多万人获得了等级证书。全国许多地区和部门也组织了本地区或本系统的计算机统一测试。许多同志认为,“学历”是从整体上反映了一个人的知识水平,而“证书”则反映了一个人在某一方面的能力。证书制度是学历制度的必要补充,是人才市场的需要,因而受到各方面的欢迎。

当然,计算机统一考试无论从内容上还是形式上都有一些问题需要进一步探索,例如,如何更好地反映计算机新技术的发展,怎样才能测试出应试者的实际应用能力等。相信随着时间的推移和经验的累积,这项制度会日趋完善。

全国和地区性的计算机等级考试在内容上、分级上、考试形式上大体相似。全国计算机等级考试目前分为四个等级。一级:具有计算机的初步知识和使用微机系统的初步能力。一级(B):面向公务员和在职干部,要求大致与一级相当,侧重应用能力。二级:具有计算机软件、硬件的基础知识和使用一种高级语言编制程序、上机调试的能力,可以从 QBASIC, FORTRAN, PASCAL, C, FoxBASE 五种语言中任选一种应试。三级分为两类:三级(A):具有计算机应用基础知识和计算机硬件系统开发的初步能力;三级(B):具有计算机应用基础知识和软件系统开发的初步能力。四级:具备深入而系统的计算机知识和较高的计算机应用能力。

为了帮助广大读者准备参加等级考试,1995 年我们组织编写了“计算机等级考试辅导”丛书,由清华大学出版社出版。该丛书包括一级、一级 B、二级(五种)的考试辅导共 8 本书,在出版后三年多时间内已发行 120 多万册。

根据计算机应用的发展,1998 年教育部考试中心修改了全国计算机等级考试大纲。和原大纲相比,新大纲有以下改动:一级除了可以选择 DOS 环境应试外,还可以选择 Windows 环境应试;二级的 BASIC 语言改为 QBASIC 语言;在二级的计算机基础知识部分中增加了对网络和多媒体的要求。

根据广大读者的要求,我们在原来的“计算机等级考试辅导”丛书的基础上,组织出版

这套“计算机等级考试丛书('98大纲)”。该丛书由以下三个系列构成:(1)计算机等级考试教程:全面而系统地介绍考试大纲所规定的内容,是便于自学的教材。(2)计算机等级考试辅导:用来帮助已学过该课程的读者复习和准备考试,每本书的内容均包括各章要点、各章难点、例题分析、思考题,并附有模拟试题。(3)计算机等级考试样题汇编:按照计算机等级考试的内容和试题形式,提供了500~600道样题,供应试者准备考试参考。

本丛中各书的作者都是高等学校或计算机应用部门中具有丰富教学经验并对计算机等级考试有较深入研究的教授、专家。相信该丛书的出版一定会受到广大准备参加计算机等级考试的读者的欢迎。

欢迎读者对本丛书提出宝贵意见。

“计算机等级考试丛书('98大纲)”主编

全国计算机等级考试委员会副主任

谭浩强

1999年3月

前 言

PASCAL 语言是由瑞士的沃斯(N. Wirth)教授于 1971 年提出来的。它的命名是为了纪念法国数学家 Pascal。

PASCAL 语言的建立基于两个主要目的:第一,提供一种能够清晰、自然地表述某些基本概念的语言,使其成为基本概念系统训练的工具,适合于程序设计教学。第二,使新定义的语言能在现有计算机上可靠、有效地加以实现。

PASCAL 语言是系统地体现由戴克斯特拉(E. W. Dijkstra)和霍尔(C. A. R. Hoare)定义的结构化程序设计概念的第一个语言,因此它是程序设计语言发展史中的一个里程碑。由于它结构清晰、便于学习和有较丰富的数据类型和语句,而且编译、运行效率高,便于移植,它已广泛地用于大学程序设计语言的教学和许多应用软件、系统软件的开发中。

本书是作者根据教育部考试中心 1998 年制定的《全国计算机等级考试 考试大纲》并在长期教学经验的基础上编写的。它不仅可以作为参加计算机等级考试人员的培训教材,也可以作为大学计算机软件、硬件专业和其他专业的程序设计教学用书,还可以作为工程技术人员的自学参考书和培训教材。

本书不仅全面介绍了标准 PASCAL 语言,还讲述了自顶向下逐步求精的结构化程序设计的基本思想和基本方法。学习程序设计必须理论联系实际。本书在介绍基本概念的同时,列举了典型而有意义的例题和习题。读者通过阅读这些例题和自己动手完成习题,并尽可能地上机通过,就一定可以学好程序设计。本书还强调培养良好的程序设计风格和习惯,这对于一个优秀的程序设计者是很重要的。

全书共分 12 章。第 1 章介绍了必要的基本知识。第 2 章到第 5 章介绍了结构化程序的四种基本结构(顺序结构、选择结构、重复结构、函数与过程)及其设计方法,此外还介绍了 PASCAL 的四种标准数据类型和一些基本语句。这四章是本书的核心,掌握了这四章的内容也就掌握了程序设计的基本方法。第 6 章到第 11 章介绍了 PASCAL 的各种用户定义数据类型(枚举类型、子界类型、数组类型、集合类型、记录类型、文件类型、指针类型),它们的引入扩大了 PASCAL 的应用领域。通过这几章的学习,不仅能了解 PASCAL 的丰富的数据类型和处理语句,而且对前几章所讲述的程序设计基本方法是一次新的应用和提高。第 12 章是一些余留问题。在每章之后都指出了一些常见错误,给出了小结并附有一些精选的习题。书末提供了“全国计算机等级考试二级(基础知识和 PASCAL 语言程序设计)考试大纲”和“1999 年 4 月全国计算机等级考试二级笔试试卷(基础知识和 PASCAL 语言程序设计)”及其答案,还有各章习题的参考答案。

本书如有不妥之处,希望读者批评指正。

郑启华

2000 年于清华大学

目 录

| | |
|------------------------------------|--------|
| 第 1 章 PASCAL 语言程序设计介绍 | (1) |
| 1.1 PASCAL 语言的特点 | (1) |
| 1.2 基本符号、保留字、标识符 | (2) |
| 1.2.1 基本符号 | (2) |
| 1.2.2 保留字 | (2) |
| 1.2.3 标识符 | (2) |
| 1.3 程序结构 | (4) |
| 1.4 小结 | (9) |
| 习题 | (9) |
| 第 2 章 顺序结构程序设计 | (12) |
| 2.1 引言 | (12) |
| 2.2 用计算机解题的基本方法 | (12) |
| 2.2.1 问题分析 | (12) |
| 2.2.2 问题解的描述 | (13) |
| 2.3 标准数据类型 | (14) |
| 2.3.1 实型(real) | (14) |
| 2.3.2 整型(integer) | (17) |
| 2.3.3 字符型(char) | (19) |
| 2.3.4 布尔型(boolean) | (20) |
| 2.3.5 标准类型小结 | (21) |
| 2.4 表达式与赋值语句 | (23) |
| 2.5 READ 语句 | (25) |
| 2.6 WRITE 语句 | (27) |
| 2.7 顺序程序设计举例 | (29) |
| 2.8 常见的错误 | (33) |
| 2.9 小结 | (33) |
| 习题 | (34) |
| 第 3 章 选择结构程序设计 | (39) |
| 3.1 引言 | (39) |
| 3.2 IF 语句 | (39) |
| 3.2.1 IF 语句的两种形式 | (39) |
| 3.2.2 复合语句 | (42) |

| | |
|------------------------------|--------------|
| 3.2.3 复合 IF 语句 | (45) |
| 3.3 CASE 语句 | (48) |
| 3.4 常见的错误 | (52) |
| 3.5 小结 | (53) |
| 习题 | (53) |
| 第 4 章 循环结构程序设计 | (57) |
| 4.1 引言 | (57) |
| 4.2 FOR 语句 | (57) |
| 4.3 WHILE 语句 | (68) |
| 4.4 REPEAT 语句 | (71) |
| 4.5 多重循环 | (73) |
| 4.6 常见的错误 | (85) |
| 4.7 小结 | (85) |
| 习题 | (86) |
| 第 5 章 函数与过程程序设计 | (93) |
| 5.1 引言 | (93) |
| 5.2 自顶向下程序设计方法 | (93) |
| 5.3 函数 | (94) |
| 5.3.1 标准函数回顾 | (94) |
| 5.3.2 定义新函数 | (94) |
| 5.3.3 函数调用 | (96) |
| 5.4 过程 | (99) |
| 5.4.1 过程与函数的区别 | (99) |
| 5.4.2 过程说明 | (101) |
| 5.4.3 过程调用 | (102) |
| 5.5 嵌套与递归 | (105) |
| 5.5.1 嵌套 | (105) |
| 5.5.2 递归 | (108) |
| 5.6 标识符的作用域 | (113) |
| 5.7 常见的错误 | (119) |
| 5.8 小结 | (119) |
| 习题 | (120) |
| 第 6 章 枚举与子界类型 | (132) |
| 6.1 引言 | (132) |
| 6.2 枚举类型 | (132) |

| | | |
|--------------|-------------------|--------------|
| 6.2.1 | 枚举类型说明 | (132) |
| 6.2.2 | 枚举类型运算 | (134) |
| 6.2.3 | 读和打印枚举值的方法 | (134) |
| 6.3 | 子界类型 | (136) |
| 6.3.1 | 子界类型说明 | (136) |
| 6.3.2 | 子界运算 | (137) |
| 6.4 | 常见的错误 | (138) |
| 6.5 | 小结 | (139) |
| | 习题 | (139) |
| 第 7 章 | 数组类型 | (142) |
| 7.1 | 引言 | (142) |
| 7.2 | 一维数组 | (142) |
| 7.3 | 多维数组 | (158) |
| 7.4 | 紧缩字符数组 | (163) |
| 7.5 | 常见的错误 | (167) |
| 7.6 | 小结 | (168) |
| | 习题 | (168) |
| 第 8 章 | 集合类型 | (175) |
| 8.1 | 引言 | (175) |
| 8.2 | 集合类型说明 | (175) |
| 8.3 | 集合运算 | (176) |
| 8.3.1 | 赋值空集和全集 | (176) |
| 8.3.2 | 集合的并交差 | (177) |
| 8.3.3 | 集合关系运算 | (178) |
| 8.4 | 类型间的关系 | (184) |
| 8.4.1 | 定义 | (184) |
| 8.4.2 | 应用 | (186) |
| 8.5 | 常见的错误 | (188) |
| 8.6 | 小结 | (189) |
| | 习题 | (189) |
| 第 9 章 | 记录类型 | (193) |
| 9.1 | 引言 | (193) |
| 9.2 | 记录说明 | (193) |
| 9.3 | WITH 语句 | (195) |
| 9.4 | 记录数组 | (196) |

| | | |
|---------------|------------------------|--------------|
| 9.5 | 层次记录 | (201) |
| 9.6 | 记录变体 | (204) |
| 9.7 | 常见的错误 | (208) |
| 9.8 | 小结 | (209) |
| | 习题 | (209) |
| 第 10 章 | 文件类型 | (212) |
| 10.1 | 引言 | (212) |
| 10.2 | 建立和使用文件 | (212) |
| 10.3 | 文件的处理 | (220) |
| 10.3.1 | 文件比较 | (220) |
| 10.3.2 | 文件修改 | (222) |
| 10.3.3 | 文件合并 | (226) |
| 10.4 | 正文文件 | (230) |
| 10.5 | 常见的错误 | (234) |
| 10.6 | 小结 | (235) |
| | 习题 | (235) |
| 第 11 章 | 指针和动态数据结构 | (241) |
| 11.1 | 引言 | (241) |
| 11.2 | NEW 语句和指针 | (241) |
| 11.3 | 链表 | (244) |
| 11.3.1 | 建立链表 | (244) |
| 11.3.2 | 删除一个结点 | (248) |
| 11.3.3 | 插入一个结点 | (250) |
| 11.3.4 | 建立有序链表 | (253) |
| 11.4 | 常见的错误 | (254) |
| 11.5 | 小结 | (254) |
| | 习题 | (255) |
| 第 12 章 | 其它问题 | (257) |
| 12.1 | GOTO 语句 | (257) |
| 12.2 | 形式语法描述 | (260) |
| 12.2.1 | 引言 | (260) |
| 12.2.2 | 语法图 | (261) |
| 12.2.3 | 巴科斯-瑙尔范式(BNF) | (262) |
| 12.3 | TURBO PASCAL 简介 | (264) |
| | 习题 | (265) |

| | |
|---|-------|
| 附录 | (266) |
| 附录 1 PASCAL 的保留字、标识符及标点符号 | (266) |
| 附录 2 标准标识符 | (268) |
| 附录 3 PASCAL 语法图 | (271) |
| 附录 4 ASCII 码 | (277) |
| 附录 5 全国计算机等级考试二级(基础知识和 PASCAL 语言 程序设计)考试大纲 | (278) |
| 附录 6 1999 年 4 月全国计算机等级考试二级笔试试卷 (基础知识和 PASCAL 语言程序设计) | (280) |
| 附录 7 各章习题参考答案 | (297) |

第 1 章 PASCAL 语言程序设计介绍

PASCAL 语言是由瑞士的沃斯(N. Wirth)教授于 1971 年提出来的。它的命名是为了纪念法国数学家 Pascal。

本章将介绍 PASCAL 语言的特点、基本符号、保留字、标识符与程序结构。这些内容是很重要的,是今后学习和正确编写程序所必需的知识。

1.1 PASCAL 语言的特点

PASCAL 语言是在 ALGOL60 基础上发展起来的,它有如下特点:

(1) 它是世界上第一个结构化程序设计语言

由戴克斯特拉(E. W. Dijkstra)和霍尔(C. A. R. Hoare)提出的结构化程序设计思想,是程序设计发展史上的一个里程碑。他们主张在程序设计中去掉 GOTO 语句,所有程序都可以由三种基本结构(顺序结构、选择结构、循环结构)组成。后来有人把函数和过程作为第四种基本结构。这四种基本结构对外来看都只有一个入口,一个出口,结构清晰,避免了由 GOTO 语句所引起的混乱。另外在程序设计方法上,他们主张采取自顶向下,逐步求精的方法。即将一个大的复杂的问题,划分成若干小的易解决的问题。每个小问题,又可划分成一些更小的更易解决的问题。这样,每个小问题解决了,整个大问题也就解决了。这种方法还为多个人同时编程序提供了方便。

而 PASCAL 语言正是基于结构化程序设计思想建立的。它所提供的语句可以充分满足实现四种基本结构的需要。它的函数和过程又为进行自顶向下,逐步求精提供了方便。

由于 PASCAL 语言具有良好的结构化程序设计特性,所以它特别适合于教学,适合于培养学生掌握自顶向下逐步求精的结构化程序设计思想和方法,并养成良好的程序设计风格和习惯。因此,国内外许多大学都将 PASCAL 作为第一门程序设计教学语言。

(2) 功能强、应用广

PASCAL 语言提供了丰富的数据类型和语句,功能强、应用广。它不仅适合于教学,也可广泛用于编写各种系统软件和应用软件。

(3) 编译和运行效率高

在 PASCAL 语言中提供了必要的说明,并去掉了一些影响效率的因素(例如去掉了乘幂运算,去掉了字符串运算,去掉了动态数组等),使得 PASCAL 语言编译和运行效率都较高。

(4) 可移植、易推广

PASCAL 语言标准化程度高,不依赖于具体的机器,用 PASCAL 语言写的源程序可以在各种具有 PASCAL 编译系统的机器上运行。如果某机型没有 PASCAL 编译系统,也可以通过用 PASCAL 语言写编译系统的自编译方法,为该机型产生 PASCAL 编译系统。

现在世界上几乎所有大、中、小型计算机都已配置了 PASCAL 编译系统,为 PASCAL 语言的广泛使用打下了基础。

1.2 基本符号、保留字、标识符

1.2.1 基本符号

PASCAL 语言只能使用以下几类基本符号:

(1) 大小写英文字母

A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z

a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z

(2) 数字

0,1,2,3,4,5,6,7,8,9

(3) 其它符号

+, -, *, /, =, <>, <=, >=, <, >, (,), [,], {, }, :=, ,, ;, ·, : , · · , ' , ↑

注意,PASCAL 语言除了能使用以上规定的基本符号外,不得使用任何其它符号。例如:

$\alpha, \beta, \gamma, \pi, \epsilon, \Sigma, \int, \$$

等都不得在 PASCAL 语言中使用。

1.2.2 保留字

在 PASCAL 语言中,有些词具有特定的含义。用户必须了解其含义,以便正确地使用,否则会造成错误。这些具有特定含义的词被称为保留字。保留字一共有 35 个,它们是:

AND, ARRAY, BEGIN, CASE, CONST, DIV, DO, DOWNTO, ELSE, END, FILE, FOR, FUNCTION, GOTO, IF, IN, LABEL, MOD, NIL, NOT, OF, OR, PACKED, PROCEDURE, PROGRAM, RECORD, REPEAT, SET, THEN, TO, TYPE, UNTIL, VAR, WHILE, WITH

保留字除按规定的意义使用外,不得另作它用。在书写时,保留字可以用大写,也可以用小写。在本书中,为了醒目和提高可读性,将保留字用大写,其它符号用小写。

按标准 PASCAL 语言规定,除了引号中的字母外是不区分大、小写的。但是在上机时,必须了解你所使用机器的 PASCAL 编译系统,看它对大、小写字母是如何规定的,以便正确输入和编译。

1.2.3 标识符

标识符是以字母开头的字母、数字组合。例如:

x,y,max,min,sum,a15,a3b7

都是合法的标识符。而如:

5x, x-y, α , π , ϵ , $e \times 10.5$

则不是标识符,或称它们为非法的标识符。

为了描述合法的标识符,可以借助语法图来描述。语法图是描述算法语言形式语法的工具。在语法图中,用矩形框表示需要进一步定义的语法单位。用圆框或两头是圆的长方形框表示终结符(基本符号或保留字)。用箭头表示它们之间的连接关系。

标识符的语法图如图 1.1 所示。

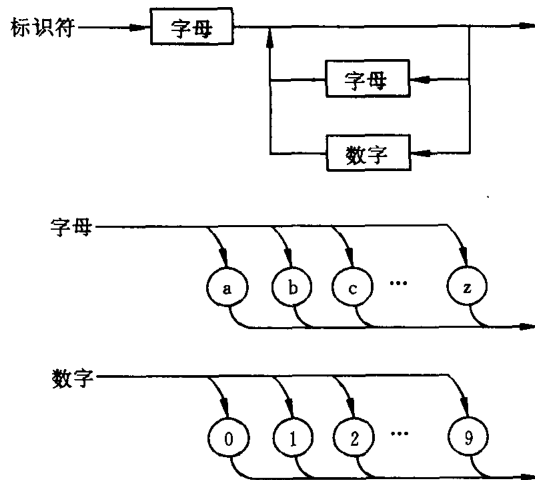


图 1.1 标识符的语法图

标识符可用来表示常量、变量、类型、文件、函数、过程或程序的名字。

标识符的长度(字符个数)是没有限制的。但标准 PASCAL 规定能区分的有效标识符长度是 8。超过 8 个字符的标识符也能使用,但是以前 8 个字符作为有效字符。因此,当两个标识符的前 8 个字符相同时,则认为是相同的标识符,多余的字符将不起作用。例如:

students1 和 students2

将被认为是相同的标识符。因为它们的前 8 个字符相同。

不同的 PASCAL 编译系统,对于可区分的有效标识符长度的规定可能是不同的,在使用时要注意。

标识符的选取最好有一定的含义,这样便于记忆,也增加了程序的可读性。标识符的书写可以用大写、小写或大、小写字母混合使用。在本书中采用小写字母。

在 PASCAL 语言中,有些标识符具有特殊的含义,称它们为标准标识符。标准标识符一共有 39 个,它们是:

标准常量 (3 个):

false, true, maxint

标准类型 (5 个):

integer, real, char, boolean, text

标准文件 (2 个)

input, output

标准函数 (17 个)

abs, arctan, chr, cos, eof, eoln, exp, ln, odd, ord, pred, round, sin, sqr, sqrt, succ, trunc

标准过程 (12 个):

get, new, pack, page, put, read, readln, reset, rewrite, unpack, write, writeln

这些标准标识符的含义和用法,将在以后各章中陆续介绍。读者在定义自己的标识符时要注意不要与保留字和标准标识符重名,以免发生错误或引起混淆。

1.3 程序结构

对于 PASCAL 程序的结构是有严格规定的。为了说明这个规定,我们先来看一个例子。

【例 1.1】 已知圆的半径,求圆的周长和面积。

设圆的半径为 r , 周长为 l , 面积为 s 。根据数学公式:

$$l = 2\pi r$$

$$s = \pi r^2$$

其中 r, l, s 可以作为合法的标识符在程序中使用。 π 是一个常量,但它不是合法的标识符,可选用 pi 来代替。

PASCAL 语言规定,程序中用到的常量和变量都必须在程序中进行说明。即说明常量的值和变量的类型。对于计算的初始数据,可以通过读语句从终端键盘读入。计算可以利用赋值语句来实现。计算结果可以通过写语句输出到屏幕或打印机。

该例完整的 PASCAL 程序如下:

程序 1.1 已知半径,求圆周长和面积的程序

```
PROGRAM circle (input,output);
  {已知半径求圆周长和面积}
  CONST
    pi=3.14159;
  VAR
    r,l,s:real;
  BEGIN
    read(r);
    l:=2*pi*r;
    s:=pi*r*r;
    write(r,l,s)
  END.
```

1. 程序首部

程序 1.1 的第一行称为程序首部。PROGRAM 是保留字。每个 PASCAL 程序都必须以它开头。circle 是该程序的名字。每个程序的名字可以不同,但必须是合法的标识符。圆括号里的内容称为程序的参数。程序参数指明程序与外部联系的文件名。input 是标准输入文件,例如键盘打字机。output 是标准输出文件,指屏幕显示器或打印机。为了读入数据和输出结果,必须写上文件参数 input 和 output。

程序中由花括号括起来的内容称为注释。该程序的第二行就是一个注释,它说明了该程序的目的。注释除了给人看,以增加程序的可读性外,对编译和运行都不起作用。一个程序可以包含多个出现在不同地方的注释,也可以没有注释。

2. 说明部分

在程序 1.1 中,从第三行到最后一行为程序的分程序。分程序一般由说明部分和语句部分构成。说明部分可以包括多种类型的说明。此例包括以 CONST 开始的常量说明和以 VAR 开始的变量说明。语句部分必须以 BEGIN 开始,以 END. 结束。中间是一些用分号分开的语句。

在该例的常量说明中,说明了常量 pi 的值为 3.14159。在以后的程序语句中,出现的 pi 都被当作是值 3.14159。

注意,在程序中只能使用已被说明的常量的值,而不得改变它的值。

常量说明(或称常量定义)的一般形式是:

```
CONST  
    <常量标识符> = <常量>;  
    :  
    <常量标识符> = <常量>;
```

解释:将指定的常量(值)赋给常量标识符。常量标识符的值可由后面的程序使用,但不得改变它的值。

常量标识符的类型与定义它的常量(值)的类型相同。因此,如果值是包括小数点的数,常量就是实型,如果值是不包括小数点的整数,常量就是整型。

常量说明的语法图如图 1.2 所示。

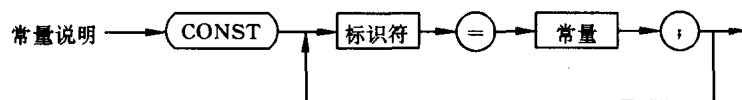


图 1.2 常量说明语法图

在程序中使用常量标识符而不使用数值本身的好处有二:其一是常量标识符的意义明确,使用它可以增加程序的可读性。其二是如果在程序的许多地方用到某个数,当要修改这个数时,必须到各个地方找出这个数,并逐一改成新数,还必须注意不要改动那些与其数值相同而意义不同的数,这是一件较困难的事。如果使用常量标识符,只需改动常量

说明中的数值即可。

在该例的变量说明中,说明 r, l, s 都是实型(real)。

注意,在程序中出现的所有变量都必须说明它们的类型。变量类型有实型、整型、字符型、布尔型等。

变量说明的一般形式如下:

VAR

〈变量表〉:〈类型〉;

⋮

〈变量表〉:〈类型〉;

解释:变量说明的作用是说明变量的类型。变量表可以是单个变量或是用逗号分开的多个变量。此时表明这些变量的类型相同,可以一起说明。编译程序将为变量说明中的每个变量分配相应的存储单元。

变量说明的语法图如图 1.3 所示。

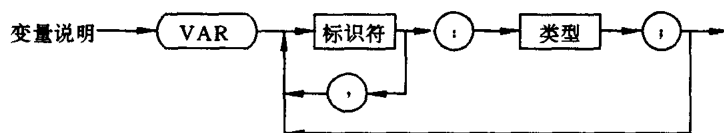


图 1.3 变量说明的语法图

3. 语句部分

从 BEGIN 开始,到 END. 结束是语句部分。程序 1.1 包括 4 个语句,每个语句之间用分号(;)分开。

第一个语句是读语句(或叫输入语句)。它的作用是读入半径 r 的值。当程序运行到此处时,将等待用户从键盘输入数据。这时如果用户从键盘输入:

12.5↵

则 r 取值为 12.5。符号↵代表回车键,输入数据结束应按此键。一个读语句也可读入多个变量的值,各变量以逗号分开。输入时各数值以空格分开。

第二个语句和第三个语句是赋值语句。它们的作用是按右端表达式计算其值,并将这个值赋给左端的变量。符号:=代表赋值号。在赋值语句中赋值只能用赋值号(:=)而不能等号(=)。符号*代表乘号。通过这两个语句,将计算出周长 l 和面积 s 的值。

表达式中最常见的运算是加(+)、减(-)、乘(*)、除(/)。在不加括号的情况下,按先乘除,后加减运算。括号只有一种,即圆括号“(”与“)”。遇括号时,先执行括号内的运算,再执行括号外的运算。

表达式必须以线型形式写出。因此,分子、分母、指数、下标等都必须写在同一行上。乘号必须明确写出,不能省略。下面是几个简单的数学表达式所对应的 PASCAL 表达式。