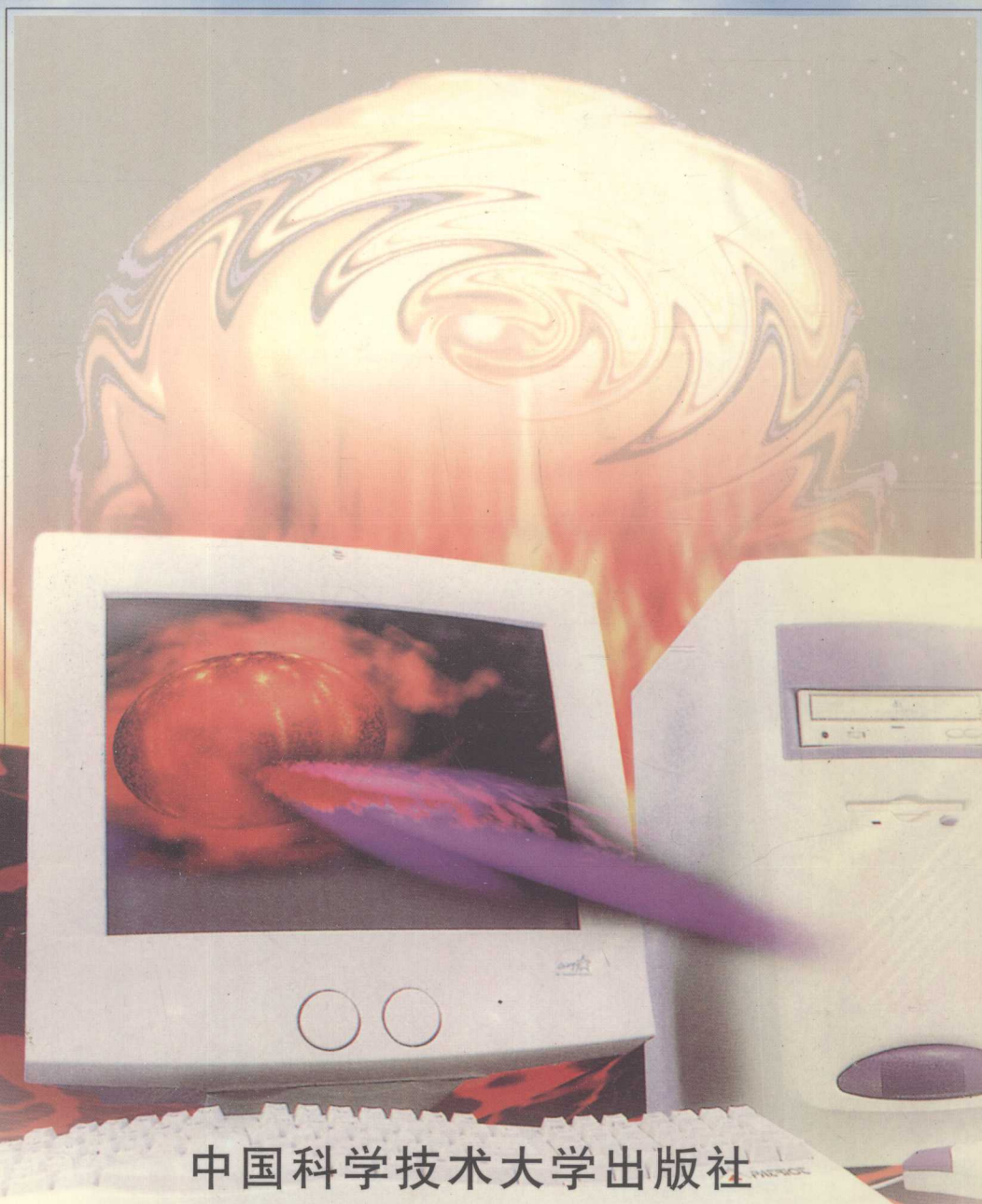


Turbo Pascal

程序设计教程

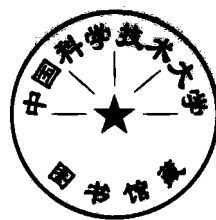
苏仕华 刘振安 张蕊 编著



中国科学技术大学出版社

Turbo Pascal 程序设计教程

苏仕华 刘振安 张 蕊 编著



中国科学技术大学出版社
1999·合肥

内 容 简 介

本书介绍了标准 Pascal 语言的基本语法、语义,通过大量精选的题例阐述了利用 Pascal 语言进行程序设计的基本方法与技巧,并着重介绍了 Turbo Pascal 的使用方法及其应用。

本书语言通俗,叙述简晰,习题丰富,突出了结构化程序设计和算法设计的一般方法。它不仅可以作为大专院校“Pascal 语言”或“程序设计”等课程的教学用书,还可以作为计算机各类用户的自学教材和参考书。

图书在版编目(CIP)数据

Turbo Pascal 程序设计教程/苏仕华等编. —合肥:中国科学技术大学出版社,
1999. 2

ISBN 7-312-01073-3

I. T… II. 苏… III. Pascal 语言-程序设计-教材 IV. TP312

中国版本图书馆 CIP 数据核字(98)第 38368 号

中国科学技术大学出版社出版发行

(安徽省合肥市金寨路 96 号,230026)

中国科学技术大学印刷厂印刷

全国新华书店经销

开本: 787×1092/16 印张: 14 字数: 349 千

1999 年 2 月第 1 版 1999 年 2 月第 1 次印刷

印数: 0—5000 册

ISBN 7-312-01073-3/TP·227 定价: 16.00 元

前 言

当今世界科学技术的发展日新月异,在发达的国家和地区,电子计算机已经渗透到社会生产、生活的各个领域。在我国,电子计算机的应用亦越来越广泛,需要掌握和使用计算机的人越来越多,对程序设计语言的学习及应用要求也就越来越高。

PASCAL 语言是在 60 年代末 70 年代初,由瑞士的沃斯(N. With)教授提出的。PASCAL 语言是在 ALGOL—60 的基础上发展起来的一种结构化程序设计语言,不但具有功能强、数据类型丰富、写出的程序简洁易懂易读等特点,而且编译、运行效率高。既能适用于应用程序设计,也能应用于系统程序设计。因此,PASCAL 语言一问世就得到广大计算机用户及学习爱好者的青睐。

Turbo Pascal 是目前微机上最流行的 Pascal 语言系统软件之一。由于它具有理想的开发环境和友好的用户界面,高效的数值运算功能及多种系统功能调用的接口,因此可用来编写各种类型的计算机应用程序。Turbo Pascal 自 85 年 3.0 版本问世以来,很快就风靡全球。Turbo Pascal 还可以像 Basic 语言那样绘图,也可以像 C 语言那样用于编写和调试程序。有许多计算机专业的基础课都使用伪 Pascal 语言进行描述,例如“数据结构”等课程。使用 C 或 C++ 编制 Windows 程序时,也使用 Pascal 调用约定。DelPhi 和 C++ Builder 与 Pascal 关系更为密切。由此可见,学习 Pascal 语言,对学习“数据结构”、“计算机绘图”、“计算机程序设计方法”、“软件工程”及其它高级语言等课程,都有极大的帮助。

本书详细地介绍了 Pascal 语言及其程序设计,通过介绍 Pascal 语言讲述了结构化程序设计的基本思想和方法。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,提供了丰富而有意义的例题、习题。在 Turbo Pascal 的使用方面,本书以面向对象的 Turbo Pascal 6.0 版本为主,详尽地介绍了 Turbo Pascal 的基本功能和应用。

全书共分十五章。第一章概述了 Pascal 语言的基础知识;第二章介绍 Pascal 语言的数据类型;第三章介绍顺序结构的程序设计方法;第四章介绍选择结构的程序设计方法;第五章介绍循环结构程序设计技术;第六章介绍枚举和子界类型;第七章介绍数组及其应用;第八章介绍子程序结构的程序设计;第九章介绍集合类型和记录类型;第十章介绍文件类型;第十一章介绍了指针的概念;第十二章介

绍程序、单元及应用;第十三章介绍 Turbo Pascal 程序设计环境的使用方法;第十四章介绍 Turbo Pascal 程序调试的基本方法;第十五章介绍 Turbo Pascal 绘图的知识。每章均精选了若干例题,其中第一章~第十一章还配备了适量的习题,可供读者自己解题和上机调试,逐步提高程序设计的技巧。为了方便读者学习,本书还把一些常用知识,以附录的形式给出。它们是错误信息、ASCII 码表、扩展码表和键盘扫描码表。

为保证教材质量和风格的统一,经我们广泛征求多方面的意见,结合我校多年、多层次教学的实际情况,充分讨论制定编写大纲,通过分工负责,层层把关,以求圆满完成写作任务。全书由苏仕华执笔主编,刘振安主审,张蕊复核并仔细验证程序的正确性。本书从写作到最后定稿,花了近两年时间,在写作过程中得到出版社社长张泰永编审的大力支持和帮助,尽量使我们的拙作得以精益求精。参加本书编写的还有:经纶、许文铮和贾伯琪等老师。

本书承蒙我校计算机系主任、博士生导师陈国良教授对全书进行了仔细审阅,并提出许多宝贵意见。另外,在本书编写过程中,还参阅了许多书籍和资料,特此一并表示感谢。

本书是在长期教学和应用开发经验的基础上,参考有关教材和资料编写的。全书内容通俗易懂,循序渐进。它不仅可以作为大专院校“Pascal 语言”或“程序设计”等课程的教学用书,还可以作为计算机各类用户的自学教材和参考书。

虽然经过多次仔细校对,但仍难免有不妥之处,敬请读者给予批评指正。

作者

1999 年 1 月

目 录

第一章 Pascal 语言概述	1
1.1 Pascal 语言的发展	1
1.2 Pascal 语言的特点	2
1.3 Pascal 语言的程序结构	3
1.4 Pascal 语言的符号	6
习题 1	8
第二章 Pascal 语言的数据类型	9
2.1 常数与常量	9
2.2 变量与变量说明	12
2.3 标准函数	16
2.4 表达式	16
习题 2	19
第三章 顺序结构的程序设计	20
3.1 语句分类	20
3.2 赋值语句	20
3.3 输入与输出语句	21
3.4 程序设计实例	26
习题 3	27
第四章 选择结构的程序设计	29
4.1 复合语句	29
4.2 IF 语句	30
4.3 CASE(情况)语句	34
习题 4	37
第五章 循环结构程序设计	39
5.1 当循环语句(WHILE 语句)	39
5.2 直到循环语句(REPEAT—UNTIL 语句)	41
5.3 FOR 循环语句	43
5.4 GOTO 语句和标号说明	47
习题 5	49
第六章 枚举型和子界型	51
6.1 枚举类型	51
6.2 子界类型	55

习题 6	57
第七章 数组及其应用	58
7.1 数组的定义.....	58
7.2 字符数组及字符串.....	65
7.3 布尔型数组.....	67
习题 7	69
第八章 子程序结构的程序设计	71
8.1 函数.....	71
8.2 过程.....	74
8.3 全程量和局部量.....	79
8.4 子程序结构程序设计示例.....	82
8.5 递归子程序.....	84
习题 8	88
第九章 集合和记录类型	92
9.1 集合类型.....	92
9.2 记录类型.....	97
习题 9	107
第十章 文件类型.....	108
10.1 顺序文件类型的定义.....	109
10.2 文件的建立和读入.....	109
10.3 文件的复制、更新和合并	114
10.4 文本文件.....	118
习题 10	121
第十一章 指针类型.....	122
11.1 指针类型的定义.....	122
11.2 指针变量和指针对象的运算.....	123
11.3 链表.....	127
11.4 栈和队列.....	131
习题 11	136
第十二章 程序、单元及应用	137
12.1 什么是单元.....	137
12.2 uses 子句	138
12.3 单元的结构.....	138
12.4 接口部分.....	139
12.5 实现部分.....	139
12.6 初始化部分.....	140
12.7 如何使用单元.....	140
12.8 引用单元说明.....	141
12.9 标准单元简介.....	144

12.10 编写用户单元	145
12.11 Turbo Pascal 的 Crt 单元功能及使用	147
第十三章 Turbo Pascal 程序设计环境	157
13.1 Turbo Pascal 集成环境概述	157
13.2 Turbo Pascal 编辑器	162
13.3 集成环境菜单的使用	165
13.4 Turbo Pascal 集成环境下的程序应用	168
第十四章 Turbo Pascal 程序调试	171
14.1 程序错误类型	171
14.2 Turbo Pascal 集成调试器	172
第十五章 Turbo Pascal 绘图	180
15.1 绘图程序的基本结构	180
15.2 Turbo Pascal 图形单元中有关图形函数	182
15.3 基本图形绘制函数	187
15.4 图形文本	191
15.5 动画程序设计	193
附录 A 错误信息	195
A.1 编译错误信息	195
A.2 运行错误	205
附录 B ASCII 码表	209
附录 C 扩展码表	211
附录 D 键盘扫描码表	213
主要参考文献	215

第一章 Pascal 语言概述

Pascal 语言结构严紧、层次清楚,能系统地体现结构化程序设计及软件工程的思想。它虽然能胜任各种系统软件及应用软件的开发,但这并非它的主要目的。它之所以被公认为世界上最流行的程序设计语言之一,是因为它最适合作为程序设计教学。例如程序设计、数据结构及软件工程的教课书,都是用 Pascal 语言描述的。由此可见,作为计算机专业的学生,必须熟练地掌握 Pascal 语言程序设计技术。

本章将简要介绍 Pascal 语言的发展及特点,并讨论它的程序结构及其一般规则。

1.1 Pascal 语言的发展

1957 年,美国 IBM 公司率先推出了第一个高级语言 FORTRAN 语言。其后不久,于 1958 年由 GAMM(德意志联邦共和国应用数学与力学协会)在苏黎世会议上提出了“关于算法语言 ALGOL 的最初报告”,并于 1960 年由来自丹、英、法、德、荷、瑞、美等国的 13 位计算机科学的专家、学者进一步研究,通过了高级语言 ALGOL-60 语言。随着计算机应用向经济领域的不断渗透,于 1962 年又发表了主要用于商业数据处理的高级语言 COBOL 语言。这是三种迄今仍广泛使用的高级语言。FORTRAN 语言和 ALGOL 语言主要适用于科学技术中的数值计算,而 COBOL 语言则主要适用于数据处理。以这三种语言为代表的早期高级语言,对加快计算机的推广与应用,曾经起了重大作用。1964 年~1967 年间,美国 IBM 公司又逐步推出了可应用于任何领域的 PL/1 通用语言。不过,这种语言规模庞大,规则繁多,不便掌握,不易推广。随着结构化程序设计思想的提出和发展,瑞士苏黎世联邦工业大学的 N. 沃思(Niklaus Wirth)教授,于 1968 年研究出一种反映了结构程序设计概念的高级语言。为了纪念对计算机科学有贡献的法国数学家 Blaise Pascal,N. 沃思教授特意将自己新发明的这种高级语言命名为 Pascal 语言,并在 1971 年将它正式发表在瑞士《ETH》杂志上。1973 年又发表了一个修订报告和一个用 ISO 字符集表示的语言定义(即标准 Pascal),从而进一步完善了 Pascal。Pascal 语言的诞生,是计算机语言发展史上一个重要的里程碑。

1987 年,我国批准了 Pascal 语言国家标准(GB 7591-87)。我国制定的 Pascal 国家标准与 ISO 标准一致,有不少 Pascal 版本是参照 1974 年的标准 Pascal 实现的,如 UCSD Pascal、OMSI Pascal、PDP-11 Pascal 等等。也有不少 Pascal 版本是参照 1983 年 ISO 的标准 Pascal 实现的,如 MS Pascal、SVS Pascal、Turbo Pascal 等。本书介绍的 Pascal 语言主要是以 Turbo Pascal 6.0 为蓝本。

1.2 Pascal 语言的特点

和 BASIC 语言是从高级语言 FORTRAN 中衍生而来的一样, Pascal 语言也是从高级语言 ALGOL 中衍生而来的。不过, BASIC 语言是经过简化而得的, 而 Pascal 语言则是由扩展而成的, 因此 Pascal 语言功能更强, 更便于使用, 更易于推广。

Pascal 语言的主要优点和特点是: 语法规则简明易懂, 数据类型丰富实用, 表达方式简洁灵活, 程序结构严谨清晰, 书写格式自由方便, 程序设计风格优美, 编译紧凑、运行效率高、易维护等。Pascal 语言不仅适用于数值运算和非数值运算的数据处理等应用程序的编制, 而且也适用于诸如操作系统等系统软件的编写。

Turbo Pascal 除了继承标准 Pascal 的所有优点和特点外, 还具有自己独特的风格和特点:

1. 与国际标准基本兼容, 并作了若干扩充

尽管 Turbo Pascal 未明确指明, 但与 Pascal 的国际标准基本兼容。除此之外, 它还扩充了许多其他功能, 诸如与计算机硬件有关的绝对地址变量、机器字位及字节操作、中断处理、内存管理、外部子程序调用、程序内机器代码、内部数据格式、嵌入文件、整数逻辑操作和结构常量等等。另外, 针对广泛的 IBM 微机用户, 还提供了图形及色彩、窗口和声音等特殊功能, 通常它们都是通过需求函数和过程来引用的。这些手段都充分发挥了微机自身的特点和功能, 使得 Turbo Pascal 语言的表达能力更强、更实用。

2. 用户界面好, 编程效率高

用计算机解题, 人们要做很多工作。在程序员按照逐步精化的原则用程序设计语言编制程序后, 常规的方法是首先借助正文编辑程序来编辑用户自己的程序, 然后进行编译、连接装配、运行等阶段的工作。这个过程常常要循环重复多次, 每一阶段都可能出错。一旦有错, 又需要回到用编辑程序进行修改, 再重复上述过程, 直到得到满意的结果。这种传统的方法无统一的用户界面, 多个工具分离, 用户处理整个过程中在工具程序之间转来转去、调进退出, 很花時間。Turbo Pascal 很好地解决了这个问题, 它具有工具箱式的环境, 有统一的界面。提供了集高性能文件管理、编辑、编译、调试运行为一体的集成开发环境。自 4.0 版本以后, 这个界面是彩色多窗口的。现在的 6.0 版本, 更是为之一新, 能提高编程效益, 并有如下特点:

- ① 多重叠窗口;
- ② 鼠标、菜单、对话框;
- ③ 多文件编辑器, 可编辑的文件长度可达 1Mb;
- ④ 增强性调试工具;
- ⑤ 完整的工作面保存和恢复功能。

3. 查错功能强, 错误定位准

Turbo Pascal 的集成开发环境交互性能好, 对文件可以统一处理, 直接支持编辑、编译、连

接装配、运行和调试等各阶段工作。值得特别指出的是, Turbo Pascal 在编译时的查错功能甚强。一旦发现有错, 立即进入编辑状态进行修改, 修改后直接转编译源程序。它的错误定位及错误性质的指示相当准确。这一点对程序员而言无疑是很受欢迎的。在 Turbo Pascal 中仅当修改了当前找到的第一个错误之后, 才能继续编译下去。这一方面说明其编译算法设计得好, 发挥了微机的特点, 同时反映了 Pascal 语言的定义严谨。

4. 单元可分别编译, 有利于大型软件的开发

Turbo Pascal 中的一个单元(Unit)是常数、数据类型、变量、过程及函数的集合, 每个单元很像一个独立的 Pascal 程序, 它有必要的初始化代码和程序体, 是允许程序分块独立编译的说明库。

单元设施可用于“软件工程”课程的教学和实践。例如, 在概要设计阶段, 可以利用单元的接口部分进行模块的功能描述、模块接口说明、数据的组织等; 详细设计阶段, 在实现部分进行各模块的细化工作; 编码阶段, 对单元进行分别编译; 测试阶段, 根据具体情况进行由底向上或自顶向下的测试组装工作。

5. 带面向对象的扩充, 支持面向对象程序设计(OOP)

面向对象的程序设计(简称 OOP)是在抽象数据类型的基础上发展起来的一种颇有前途的软件设计方法, 它将数据抽象和类型继承融为一体, 使人们在软件设计中普遍遵循的模块化、信息隐藏、抽象代码共享等软件工程思想得到充分的体现。Turbo Pascal 自 5.5 版本起, 扩充了支持面向对象程序设计的设施。在 Turbo Pascal 6.0 中又得到了进一步的改进和充实, 使得 Turbo Pascal 也跻身于 OOP 的行列, 成为面向对象程序设计的有力工具。

6. 其它

- ① 它规定了无类型变量, 对于书写系统软件这是绝对必要的;
- ② 说明部分的各个说明段次序可随意, 十分灵活方便;
- ③ 自带的编辑器与 WordStar 命令几乎相同, 广大用户不必专门为了掌握编辑程序而花很多功夫。

由于以上特点, 掌握 Pascal 语言已成为学习计算机这门学科的必备知识之一, 它能给学生以严格而良好的程序设计的基本训练。

1.3 Pascal 语言的程序结构

在系统地学习 Pascal 语言之前, 我们先学习一个简单的 Pascal 语言程序实例, 以便对它的程序结构有个初步的、完整的认识。应当指出, 虽然 Pascal 语言的编译程序不区分大、小写字母, 也不管程序的书写格式如何。但是, 为了提高程序的易读性、易维护性, 则应采用如下程序书写技巧与方法: Pascal 程序均采用缩进对齐的书写格式; 大写字母(或黑体字母)专用于表示保留字(如: BEGIN, END, IF, THEN, ELSE, REPEAT, UNTIL 等), 而小写字母则用来表示一般标

标识符(即自定义标识符,如变量名、程序名、文件名等)。

【例 1.1】根据所给半径,求圆面积。

```

PROGRAM yarea(Input,Output); {程序首部}
CONST
    pi=3.1415926;           {常量说明}
VAR
    r,area;Real;           {变量说明}
BEGIN                      {执行语句开始}
    Read(r);
    area:=pi*r*r;
    Writeln('area=',area);
END.                       {执行语句结束}

```

在这个程序中,第一行为程序首部;第 2~5 行为说明语句部分;第 6~10 行为执行语句部分。这个简单的程序实例表明,一个 Pascal 程序由程序首部和程序体(包括说明部分和语句部分)两个部分组成。这种 Pascal 程序的基本构造方式,可简要而直观地表示为如图 1.1 所示的 Pascal 程序结构语法图。所谓语法图,是指用以直观地描述有关语法规则的形象化图形。在 Pascal 语言的语法图中,通常弧形框内为保留字或标识符;圆形框内为有关运算符及其它无需再定义的语法成分;矩形框内为尚需要由其它有关语法图来定义的语法实体;箭头“→”所指者为有关相邻语法成分的后继部分。



图 1.1 Pascal 程序结构语法图

1. 程序首部

程序首部是程序的开头部分,其语法如图 1.2,它包括:

- ① 程序标志:用“PROGRAM”来标识“这是一个 Pascal 程序”。标准 Pascal 规定任何一个 Pascal 程序都必须以此字开头。
- ② 程序名称:由程序设计人员自己定义,如例中的 yarea。
- ③ 程序参数:写在程序名称后面括号内,如例中的“Input”、“Output”,程序参数(逻辑设备名)表示程序与外界的联系,“Input”是系统预定义的输入文件,一般隐含指定为键盘。“Output”是系统预定义的输出文件,一般为显示器。如果程序无须输入数据,则“Input”可以省略。在写完程序首部之后,应有一个分号。



图 1.2 程序首部的语法图

2. 程序体

程序体是程序的主体,有时也称为“分程序”。程序体包括说明语句部分和执行语句部分两个部分。

① 说明语句部分:所谓说明语句部分,它是用来描述程序中所要用到的标号、常量、类型、变量、过程和函数的,由标号说明、常量说明、类型说明、变量说明、过程说明和函数说明等顺序构成。不过,这并不意味着一定要求这六部分都同时出现在说明中;相反,它只要求那些在本程序要用到的说明部分依序出现其中。Pascal 语言规定,凡是在程序要用到的所有数据量都必须在说明部分加以说明。如在【例 1.1】程序中的第 2~3 行是“常量说明”,定义“pi”为一个常量标识符,它代表 π 的近似值;第 4~5 行是“变量说明”,定义 r、area 两标识符为实型变量。说明部分的语法图如图 1.3 所示。

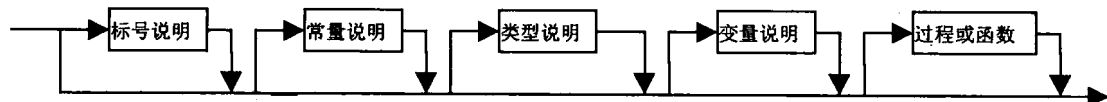


图 1.3 说明部分的语法图

② 执行语句部分:执行语句部分是整个程序的主体,是程序的可执行部分。而前面所述的程序首部与说明部分则是程序的非执行部分,执行部分是一个 Pascal 程序的核心部分。执行部分以 BEGIN 开始,以 END 结束,其间有若干个语句,语句之间以分号隔开。执行部分之后有一个句点,表示整个程序结束。其语法图如图 1.4 所示。

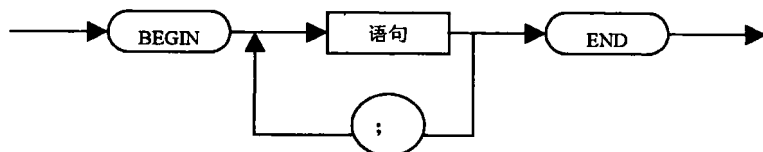


图 1.4 执行部分语法图

此外,Pascal 程序的书写方法也比较灵活。为了增强程序的易读性,可在程序中使用一对花括号,把所要注释说明的文字内容括起来。注释的内容由人们根据需要书写,可以用英语或汉语表示。注释可以放在任何所需之处。

Pascal 语言程序的书写格式非常灵活,允许一行写多条语句,也允许一条语句写成多行。但为了程序的可读性和易修改性,一般不提倡这种写法。而 Turbo Pascal 的程序结构更为灵活,甚至可以省略程序首部。

1.4 Pascal 语言的符号

1.4.1 字符集

Pascal 语言的基本字符集由字母、数字和其它符号三部分共 56 个字符组成。26 个字母不区别大、小写,即 A,B,C,...,Y,Z 或 a,b,c,...,x,y,z。数字即 10 个阿拉伯数字:0,1,2,...,9。其它符号是指如下 20 个字符:

+ - * / . ' = < > , : ; ^ () [] { } 空格

1.4.2 关键字

关键字又称保留字,是指语言中具有固定意义的一批英文单词(或英文单词的缩写),如 BEGIN、END、VAR、PROGRAM 等。每个关键字都有专门的用途,用于特定的位置,不能再作其它用途。标准 Pascal 中一共有 37 个关键字,它们是:

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

其中,除 OTHERWISE 外,其余 36 个保留字称为标准 Pascal 保留字,OTHERWISE 在 Turbo Pascal 中不能用,而是用 ELSE,各个保留字的语义极其使用方法,将在以后各有关章节分别予以讲述。

1.4.3 标识符

标识符是指由字母开头再跟若干个字母或数字的字符序列。通常又分为标准标识符与用户定义的标识符(有时简称标识符)。标识符的构成语法图如图 1.5 所示。

字母取自 26 个英文字母(大小写不分),数字来源于 10 个阿拉伯数字。尽管从语法上讲,标识符的长度不限,但在计算机上实现时,总是有长度限制的。标准 Pascal 规定最长为 8 个字符,如果标识符长度超过 8 个字符,只识别前 8 个字符。有的 Pascal 版本放宽了长度限制,如 Turbo Pascal 允许有效长度为 63,并且除字母数字之外还可以使用下划线,如 name _ of _ stu-

dent 等,这样可提高程序的可读性。

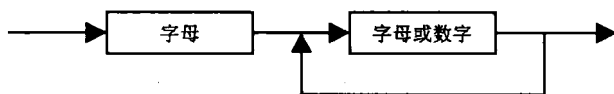


图 1.5 标识符的构成

1. 标准标识符

即 Pascal 预先定义好的标识符(也称预定义标识符),它们有特定的含义。标准 Pascal 规定了以下 40 个预定义标识符:

标识符类别	预定义标识符
标准常量名	False, True, Maxint
标准类型名	Boolean, Char, Integer, Real, Text
标准文件名	Input, Output
标准函数名	Abs, Arctan, Chr, Cos, Eof, Eoln, Exp, Ln, Odd, Ord, Pred, Round, Sin, Sqr, Sqrt, Succ, Trunc
标准过程名	Dispose, Get, New, Pack, Page, Put, Read, Readln, Reset, Rewrite, Unpack, Write, Writeln

2. 自定义标识符

由用户自己根据需要定义的标识符,是用来标识(表示)程序、过程、函数、类型、符号常量、变量等的名字的符号,或者说,标识符是可以用作以上语法实体的名字。例如一个变量名、一个常量名或一个子程序名等。

用户在定义标识符时应注意以下几点:

①禁止使用关键字作为标识符。

②尽可能避免与标准标识符同名。从语法上讲,可以使用标准标识符作为用户自定义标识符,但为保持程序的清晰易读,避免混乱或可能出现的错误,建议不要这样做。

③尽量使得用户标识符具有明显、直观、明确的意义,使所定义的标识符在形式上具有易读性,遵循“见名知义”的原则去定义自己的标识符。

1.4.4 Pascal 专用符

这是一类用以某些特殊操作或特殊内容的专用符号。它们通常由如下几类专用符构成。

①算术运算符的种类如下:

+ (加) - (减) * (乘) / (实型量相除)

②关系运算符的种类如下:

> (大于) >= (大于等于) < (小于)
 <= (小于等于) = (等于) <> (不等于)

③标点符号的种类和作用如下:

- 小数点或记录与域连接符或程序终结符
- , 有关项的分隔符
- : 变量名、类型等的说明分隔符
- ; 语句分隔符
- ' 单引号, 用作字符常数或串的定界符
- .. 子界说明符
- : = 赋值运算符
- () 括参数表达式
- [] 括下标或括集合表达式
- { } 括注释或说明

习 题 1

- 1.1 PASCAL 语言的程序结构由哪几部分组成? 试述各部分的构成形式和作用。
- 1.2 PASCAL 基本字符集由哪些成分构成? 什么是 PASCAL 保留字?
- 1.3 PASCAL 语言中有哪些常用标准文件、标准函数、标准过程以及标准常数。
- 1.4 如何定义一个 PASCAL 语言中标识符? 下列符号中哪些是正确的 PASCAL 标识符? 哪些是错误的? 为什么?

A25X, 5BH4, PEL. 1, A—31, END, ABS, U+V

A*B, \$ 2400, FG1(x), CAI, CONST, xzy, E—05

第二章 Pascal 语言的数据类型

Pascal 语言的主要优点之一,就是为用户提供了丰富的数据类型。在 Pascal 语言中,数据总是属于一定的数据类型的,而每一种数据类型则决定了一类数据的取值范围(即值域)和它所能参加的各种运算。Pascal 语言数据类型的分类,如图 2.1 所示。

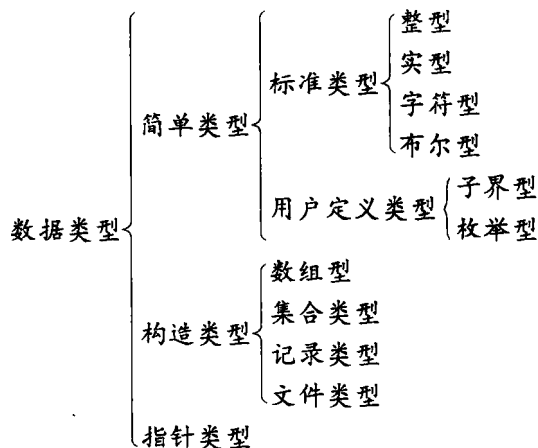


图 2.1 Pascal 语言数据类型分类

本章中仅介绍 Pascal 语言预定义的标准类型即整型、实型、字符型和布尔型,其余的数据类型将在以后的有关章节中讲述。

2.1 常数与常量

在程序中,数据常以三种最基本的形态出现,即:常数、常量与变量。

2.1.1 常数

所谓常数,是指在程序中直接出现的数字、字符或字符串(即用一对单引号括起来的有限字符序列)等。例如:123,3.14,-256.78,'ABC123','this is a string',等等。在标准 Pascal 中,常数有四种基本类型:整型、实型、字符型、布尔型(即逻辑型)。