

职业学校计算机应用专业系列教材

PASCAL语言与数据结构

马永开
桂昌宁 主编
刘永良



企业管理出版社

职业学校计算机应用专业系列教材

PASCAL 语言与数据结构

主 编 马永开

桂昌宁

刘永良

副主编 孙鸿鸣

卞志俊

企业管理出版社

图书在版编目(CIP)数据

PASCAL 语言与数据结构/马永开等主编. —北京:企业管理

出版社,1998.12

ISBN 7-80147-094-X

I. P… I. 马… II. PASCAL 语言-数据结构 IV. TP312

中国版本图书馆 CIP 数据核字(98)第 28430 号

责任编辑:齐 叶

特约编辑:晓 月

装帧设计:龙 湖

责任校对:马永开

企业管理出版社出版发行

(社址:北京市海淀区紫竹院南路 17 号 100044)

全国新华书店经销

安徽省蚌埠市方达印刷厂印刷

787×1092 毫米 1/16 印张:10.5 字数:262 千字

1999 年 5 月第 1 版 1999 年 5 月第 1 次印刷

印数:00001—15000 册

ISBN 7-80147-094-X/F·092

定价:14.80 元

总 序

从1946年第一台计算机诞生至今已历经半个多世纪。计算机的出现和广泛应用,标志着人类社会的一次大飞跃、信息时代的一次大转折、生存方式的一次大变革、现代文化的一次大融汇。随着计算机技术的飞速发展和广泛普及,其应用已遍及社会生活的各个领域。由于计算机技术已进入到我们生活中的方方面面,人类社会的生活方式、思维方式以及时空观念等各个方面都已经发生了深刻的历史性的巨变。

随着信息化社会的发展,人们对信息交流的要求越来越高。世人已普遍公认:哪个国家的信息化程度高,其经济竞争力就越强,科技发展就越快,办事效率就越高,对下一代的教育条件就越好。信息化的进一步深入需要我们培养一大批高素质人才。当今社会,熟练掌握计算机应用技术已成为高素质人才的必备条件。因此,一个十分艰巨的任务,就是要使受教育者具有使用计算机的能力和与之相适应的计算机文化素质。如果我们的知识结构和文化修养准备不足,就不能适应时代和社会发展的需要。

一本好书,是人生旅途的一掬甘泉;一套好教材,是教学成功的必要条件。广大学子和读者殷殷所望,无非是博得一艺,学有所用。本着对读者负责的精神,我们组织北京电子科技学院、集美大学、合肥工业大学、安徽大学、安徽财贸学院、江苏广播电视大学等高等院校和部分大中专学校具有丰富教学经验的教师,以及一些具有较高理论基础和软件开发经验的计算机技术人员,共同合作编写了这套计算机及应用专业教材。为保证教材的质量,我们还聘请了一批学术造诣较深的专家、教授作为本套教材的主审和编委。本系列教材具有以下几个方面的特点:

首先,作为一套计算机专业教材,必须保证整个计算机知识体系的完整性。本套教材包括必修课19种,选修课和配套教材3种,基本上涵盖了目前大中专院校计算机及应用专业所必修或选修的内容。各种教材在编排上,既注意到内容上的连贯性,又保证了教学上的相对独立性。

其次,在内容组织上,本套教材注重介绍和汲取当今计算机领域最新技术,大胆摒弃传统教材中一些过时的内容,这些变化在各本教材中都得到程度不同的体现。全套教材编写既参照了有关部委计算机及应用专业教学大纲,又参考了“程序员考试大纲”和“全国计算机水平等级考试大纲”的内容。因此本套教材既适合作为各级各类学校计算机及应用专业教材,亦可作为计算机水平等级考试学习用书。

再次,考虑到各校教学的特点,本着学以致用原则,本套教材编写始终贯彻“由浅入深,理论与实践相结合”的原则,以阐明要义为主,并辅以大量的例题、习题和上机实习,以便使读者尽快领悟和掌握。

在本套教材编写过程中,各位作者付出了艰辛的劳动,教材编委会的各位专家和教授对各本教材的内容进行了认真的审定和悉心的指导。在教材出版过程中我们自始至终得到中国物

资出版社领导和责任编辑以及印制单位的大力支持和帮助。本套教材承蒙中国科学院计算技术研究所、国家智能计算机研究开发中心王川宝、高文、中国机械科学研究院江波等同志进行了较为细致的终审终校工作。正是由于各方面的通力配合,才使得本套教材得以顺利出版和发行。书中参考、借鉴了国内外同类教材和专著,在此一并表示感谢。

计算机技术发展一日千里,许多新的概念和内容都在不断扩展之中,囿于编者的学识和水平,书中疏漏、错误之处还望广大读者不吝批评指正,以便对本套教材不断修订完善。

计算机及应用专业教材编委会

一九九八年元月

教材编委会顾问

(以姓氏笔划为序)

- | | |
|-----|------------------------|
| 王仲文 | 北京电子科技学院院长、教授 |
| 韦 穗 | 安徽大学副校长、教授 |
| 张全寿 | 铁道部电子计算中心主任、北方交通大学教授 |
| 李文忠 | 全国计算机基础教育学会副理事长、东南大学教授 |
| 杨善林 | 合肥工业大学副校长、教授、博士生导师 |
| 辜建德 | 集美大学校长、教授 |
| 魏余芳 | 西南交通大学教授 |

教材编委会编委

- | | |
|-----|----------------|
| 鄂大伟 | 集美大学副教授 |
| 李树德 | 北京电子科技学院副教授 |
| 刘 锋 | 安徽大学副教授 |
| 屈道良 | 上海铁路局蚌埠分局高级工程师 |
| 蒋翠清 | 上海铁路局蚌埠分局高级工程师 |

前 言

数据结构是研究非数值计算程序设计问题中计算机操作对象及其相互关系和操作的一门学科。作为计算机应用专业的一门核心课程,学好“数据结构”是学习“操作系统”、“数据库”、“软件工程”、“编译原理”等课程的基础。

“数据结构”这门课程的学习难点是内容抽象,为此,编者在本书内容编排上充分考虑到课程特点和中专技校职高学生的接受能力,将内容分为上、下两编,上编是 PASCAL 语言程序设计基础,下编是数据结构初步。“数据结构”这门学科是一种以结构化程序设计语言作为算法描述语言,用这种程序设计语言设计的程序具有结构严谨、清晰等特点。PASCAL 语言不但是一种理想的算法描述语言,而且简单易学。读者通过本书上编的学习,可以基本上掌握 PASCAL 语言的基础知识和结构化程序设计方法,为学习下编中的算法设计思想打好基础;此外,PASCAL 语言丰富的数据类型还有助于读者对数据结构概念和原理有更深入的理解。本书下编主要介绍数据结构的基本概念、原理和算法,重点介绍线性结构的原理、实现和建立在线性结构上的各种操作;对树型结构除介绍一些基础知识外,主要介绍二叉树及其应用;对较为复杂的图型结构仅介绍图的基本概念、简单实现和较为简单的应用,对涉及到图的算法只介绍算法思想,不具体给出算法的程序。

本书之所以采用这种体系结构,还考虑到以下两点:一是将下编中的算法描述作为上编 PASCAL 语言程序设计的延续,使读者既能比较容易地掌握数据结构中的算法设计思想,又能提高自己的程序设计能力;二是 PASCAL 语言和数据结构的直接结合使“数据结构”这门课程的上机实习成为现实。希望通过我们的这一尝试,能给读者学习“数据结构”提供一条捷径。

本书在编写过程中得到企业管理出版社和安徽财贸学院、南京经济学院领导的大力支持,书中参考、借鉴和引用了部分专家、学者的著作,编者在此表示衷心的感谢!

由于编者水平有限,书中错误和不足之处在所难免,敬请有关专家和广大读者不吝批评指正,以便不断修订完善。

编 者

1999 年 5 月

目 录

上编 PASCAL 语言程序设计基础

第1章 PASCAL 语言基础知识	(2)
§ 1.1 PASCAL 语言简介	(2)
§ 1.2 标识符和保留字	(3)
§ 1.3 直接量、常量和变量	(5)
§ 1.4 标准数据类型	(5)
§ 1.5 运算符和表达式	(7)
§ 1.6 标准函数及其引用	(9)
习题一	(10)
第2章 PASCAL 语句	(12)
§ 2.1 简单语句	(12)
§ 2.2 分支结构语句	(14)
§ 2.3 循环结构语句	(18)
习题二	(21)
第3章 过程与函数	(24)
§ 3.1 过 程	(24)
§ 3.2 函 数	(27)
§ 3.3 变量作用域和参数	(29)
§ 3.4 子程序的嵌套与递归调用	(32)
习题三	(33)
第4章 PASCAL 中自定义数据类型	(37)
§ 4.1 自定义简单类型	(37)
§ 4.2 自定义构造类型	(40)
§ 4.3 指针类型	(53)
习题四	(57)
第5章 Turbo PASCAL 简介	(59)
§ 5.1 Turbo PASCAL 集成化环境	(59)
§ 5.2 Turbo PASCAL 的特点	(62)
§ 5.3 Turbo PASCAL 程序的编辑和调试	(63)

下编 数据结构

第6章 线性表	(69)
§ 6.1 线性表及其基本运算	(69)
§ 6.2 栈	(77)

§ 6.3	队列	(80)
§ 6.4	串	(85)
习题六		(88)
第7章	线性表的排序	(89)
§ 7.1	插入排序	(89)
§ 7.2	选择排序	(92)
§ 7.3	交换排序	(93)
§ 7.4	合并排序	(97)
习题七		(99)
第8章	线性表的查找	(100)
§ 8.1	顺序查找	(100)
§ 8.2	二分查找	(102)
§ 8.3	索引查找	(104)
§ 8.4	散列查找	(106)
习题八		(110)
第9章	树	(111)
§ 9.1	树结构概述	(111)
§ 9.2	二叉树	(116)
§ 9.3	二叉树的遍历	(121)
§ 9.4	哈夫曼树及其应用	(123)
§ 9.5	树型查找	(127)
习题九		(138)
第10章	图	(140)
§ 10.1	图的基本概念	(140)
§ 10.2	图的存贮结构	(143)
§ 10.3	图的遍历	(146)
§ 10.4	最小生成树	(147)
§ 10.5	单源最短路径问题	(152)
§ 10.6	拓扑排序	(156)
习题十		(158)
主要参考文献		(160)

上 编

PASCAL 语言程序设计基础

PASCAL 语言是沃斯(N. Wirth)在 60 年代末提出来的。它较好地体现了结构程序设计的原理,是定义结构程序设计概念的第一个语言,在世界上,它是程序设计语言发展的一个里程碑。PASCAL 语言通用性强,简明易懂,容易学习、使用,是高级语言程序设计知识的基础,近年来得到了较广泛的应用。

PASCAL 语言常常被用作算法描述语言。

第1章 PASCAL 语言基础知识

§ 1.1 PASCAL 语言简介

1.1.1 PASCAL 语言的发展历史和特点

PASCAL 语言是由瑞士的 N. Wirth(N. 沃斯)教授于 1971 年在苏黎士联邦大学提出来的,这是一种结构化程序设计语言和算法描述语言。为了纪念著名哲学家、数学家 P. Pascal (P. 帕斯卡),将其命名为 PASCAL 语言。二十多年来它发展很快,1974 年颁发了标准 PASCAL 用户手册,随后在各种类型的计算机上都配备了 PASCAL 语言,IBM PC 机出现后,立即开发了 PC PASCAL 的各种版本。后来,Borland(博兰德)公司又开发了 Turbo PASCAL 版本,它的运行速度快、功能强、使用更加方便。

PASCAL 语言的第一个显著特点是其简明性和结构化。语言本身具有构成多种控制结构的功能,但表达方面又十分简明严谨。因此,用它编写的程序其结构很清晰,便于验证其正确性,这特别对一些大型程序更有好处。

PASCAL 语言的另一个特点是提供了处理十分广泛的各类数据的强大功能,它不仅可直接处理如实数、整数、字符和布尔量等一些简单类型的数据,而且给出一些由这些简单类型数据构造出复合类型数据(如数组、记录、集合等)的方法。这就大大增强了处理各类应用问题的能力,不单可用于一些传统的数值计算问题,也可方便地应用于如检索、查找、判断、决策等非数值处理问题。

1.1.2 PASCAL 程序结构

程序是对给定数据处理过程的描述。不同的语言系统有不同的成分和结构形式,使用 PASCAL 语言描述的程序称为 PASCAL 源程序。在学习 PASCAL 语言及其程序设计之前,我们先看一个简单的程序举例,通过此例了解 PASCAL 语言的程序结构。

(一)程序举例

【例 1.1】 计算圆的面积和周长的 PASCAL 源程序。

程序如下:

PROGRAM circle(input,output);	{ 程序首部}
CONST pi=3.14159;	{ 常量说明}
VAR r,s,l:real;	{变量说明}
BEGIN	{程序开始}
read(r);	{输入圆的半径}
s:=pi * sqr(r);	{计算圆的面积}

```
l:=2*pi*r;
```

```
{计算圆的周长}
```

```
write(r,s,l)
```

```
{输出圆的半径、面积、周长}
```

```
END.
```

```
{ 程序结束}
```

上述程序虽然简单,但已基本体现了一个完整 PASCAL 程序的结构。

(二) 程序结构

PASCAL 程序一般由程序首部、说明部分和执行部分三个部分组成,其中程序首部和执行部分缺一不可。

第一行称为程序首部,它固定以保留字 PROGRAM 开头(PASCAL 程序中固定使用的单词称为保留字),后面跟着用户自定义的程序名 circle,小括号内是程序与外界联系的参数,本程序中 input 表示有输入操作,output 表示有输出操作,参数之间用逗号隔开。程序首部最后的分号“;”表示程序首部结束。

从第二行至 BEGIN 之前的若干行是程序的说明部分,PASCAL 程序说明部分包括:标号说明、常量说明、类型说明、变量说明、过程和函数定义说明,但每个程序的说明部分并不都需要全部五个种类的说明,需要时才引用,甚至可以没有说明部分。本程序只出现常量说明(以保留字 CONST 开头)和变量说明(以保留字 VAR 开头),它定义了一个常量 pi,代表常数 3.14159,又说明了 r,s,l 是三个实数类型的变量。

程序执行部分就是程序的可执行部分,它规定完成此程序功能的一系列操作。执行部分必须以 BEGIN 开始,最后以 END 结束,我们称它们为一对语句括号。因此,程序的执行部分可以说就是用语句括号 BEGIN 和 END 括起来的一系列用分号隔开的语句。其中语句的数量没有限制,不过出现的次序却是不能随意颠倒的,分号是语句之间的分隔符,不是语句的组成成分,因此,最后一个语句与 END 之间不要加分号。本例中的程序共有四个语句,第一个语句 read(r)是读入圆半径 r 的值,后两个语句分别计算圆的面积 s 和周长 l,最后一个语句 write(r,s,l)是输出 r,s,l 的值。

(三) 程序的书写格式

我们有必要对程序的书写格式作些说明。虽然程序的书写格式一般不影响程序的功能,但为了使程序的结构清晰,易于阅读,注意适当的书写格式是必要的。我们在本书中都采用紧缩对齐格式书写;而且,适当加一些空格;并用大写字体表示保留字,以便与其它标识符有所区别。上面的例子就是用按这种格式书写的。除第一行程序首部外,其余各行都适当往右缩进,语句括号 BEGIN 和 END 应成对对齐,里面的语句又需再往右缩进,使得层次分明。此外,适当地在一些语句后面用花括号加一些注释,以提高程序的可读性。希望读者都能按照这种格式书写自己的程序。

§ 1.2 标识符和保留字

1.2.1 标识符

标识符是用来定义程序、过程、函数、数据类型、常量、变量等名称的符号。例如,在例 1.1 的程序中,circle 表示程序的名称,pi 表示常数 π ,r,s,l 表示实型变量的名称,它们都是标识符。

标识符由一个字母开头,后跟任意个字母或数字组成,在 Turbo PASCAL 中还可以后跟任意个下划线,举例如下:

sum	合法标识符
person __ coun	合法标识符
hit16a	合法标识符
first time	非法,不能包含空格
next, word	非法,不能包含逗号
3may	非法,不能以数字开头

标识符的长度不能超过 127 个字符,标准 PASCAL 规定只有前八个字符有效, Turbo PASCAL 规定标识符中的所有字符均有效。标识符可分为两类:标准标识符和用户自定义标识符。

标准标识符是 PASCAL 语言预先定义了其含义的,用户可以不需要再说明就可直接引用的标识符。它用于命名某些常数和标准类型、标准过程、标准函数和标准文件。例 1.1 中出现的 input、output、read、sqr、write 等都是标准标识符。所有标准标识符分列如下:

(一) 常量标识符

false true maxint

(二) 标准类型标识符

integer real boolean char text

(三) 标准函数标识符

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

(四) 标准过程标识符

get new pack put read readln
reset rewrite unpack write writeln

(五) 标准文件标识符

input output

以上的标准标识符的含义和用法将在以后适当的地方给予说明。应该指出的是,标准标识符虽已有预定的含义,但与保留字不同的是,它允许程序员在程序的说明部分重新定义而赋予其它含义。当然,一般情况下,我们都不会这样做,因为这会带来不必要的混乱。按我们的书写格式,标准标识符不用大写表示。

用户定义的标识符是用户根据定义标识符的方法按需要所定义的新标识符,一般用来给常量、变量、非标准类型、过程、函数及程序命名。例如例 1.1 中出现的标识符 circle、pi、r、s、l 等都是用户定义的标识符,分别作为程序名、常量名和变量名。用户定义的标识符必须遵循先定义后使用的原则,即应先在程序说明部分说明之后,才能在执行部分出现,这是有别于标准标识符的地方。

1.2.2 保留字

保留字是 PASCAL 语句的组成部分,有特定的意义,用户绝对不能重新定义为自己使用的标识符,它们只能用于语言中定义的用途。例 1.1 中的 PROGRAM、CONST、VAR、

BEGIN,END 都是保留字,在 § 1.1 中我们已经介绍了它们在程序中的意义。

§ 1.3 直接量、常量和变量

1.3.1 直接量和常量

在程序中一个具体的量称为一个直接量,例如:数值 1、100、3.14159265,字符 '\$'、'H',都是直接量;用户可以给直接量命名,命名了的直接量称为常量,在 PASCAL 中,常量说明部分就是给直接量命名的,它是以保留字 CONST 开头的。我们可以对上面的直接量进行常量说明如下:

```
CONST
    i1:=1;
    i2:=100;
    pi:=3.14159265;
    ch1:='$';
    ch2:='H';
```

此时, i1,i2,pi,ch1,ch2 称为常量标识符,它们是存放相应值的实体。

顾名思义,常量是一种在程序执行过程中其值不能改变的量。一个好的程序,应除在常量说明之外,很少使用直接量,这给修改程序带来方便,欲修改一个直接量,只需修改常量说明部分,勿需改动程序的执行部分。

1.3.2 变量

在程序中,每一项数据不是常量就是变量,区别在于变量的值在程序执行过程中可以改变。程序中每一变量均必须而且只能与一个数据类型相关,一个变量的类型定义了该变量可接受值的集合。

直接量和常量的类型由 PASCAL 自己决定,用户不必去说明,但是,变量的类型必须在使用前由用户说明。变量说明的例子如下:

```
VAR
    count,inde,i:integer;
    fris,last:real;
    enod:boolean;
    charac,endchare:char;
```

其中,VAR 是定义变量类型的保留字,通过以上变量说明,变量 count,inde,i 定义为整数类型变量;fris,last 定义为实数类型变量;enod 定义为布尔型变量;charac,endchare 定义为字符型变量。

§ 1.4 标准数据类型

PASCAL 语言的优点之一是具有丰富的数据类型。这些数据类型可以分为三大类。第一

类是简单数据类型,这一数据类型构造简单,又称“非构造型数据类型”,它包括四种常用的数据类型和两种用户自定义的数据类型;第二类是构造型数据类型,又称复杂型数据类型,其构造复杂,一般由其它数据类型按一定的规则构造而成;第三类是指针类型,它与前两类数据类型不同,主要用于解决动态数据的建立、删除和修改等问题。

PASCAL 数据类型,有标准类型和用户自定义类型共 11 种,如图 1-1 所示。

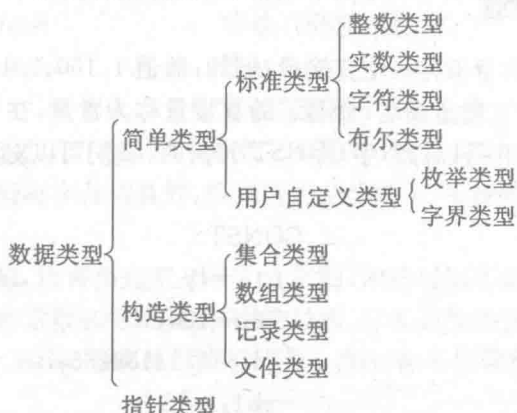


图 1-1 PASCAL 语言的数据类型

本节首先介绍标准数据类型。

1.4.1 整数类型

整数类型在变量类型定义中的标准标识符是 integer,整数的取值范围与特定的计算机的字长有关,但对每台特定的计算机都有一个表示该机所能允许的最大整数的最大标准标识符 maxint,任何一个整数 N 必须满足:

$$-\text{maxint} \leq N \leq \text{maxint}$$

超出这个范围将会产生溢出错误。

1.4.2 实数类型

实数类型在变量类型定义中的标准标识符是 real,实型数的取值范围与特定的机器有关,例如 IBM PC Turbo PASCAL 中的实型数的绝对值一般在 10^{-38} 到 10^{38} 之间,当实型数的绝对值超过 10^{38} 时,发生溢出错误,当实型数的绝对值小于 10^{-38} 时,则机器将它当 0 处理或发生下溢错误。

PASCAL 有两种形式的实型数。一种是带小数点的形式,例如: -3.5; 124.56; +147.532。

另一种是指数形式,在这种形式中用字母 ‘E’ 表示以 10 为底,如: 10^2 可以表示为 1.0E+2 2.165×10^8 、 -15×10^3 、 3.45×10^{-2} 可以分别表示成 2.165E+08、-15E3、3.45E-02。

关于实型数的表示法需要说明的是:一个实数如果有小数点,则在小数点的前后都必须有数字;如果包含有符号 E,则在 E 的前后必须有数字,而且 E 的后面只能是整数。另外,一个实型数中除了正负号、数字、小数点及字母 E 之外不能有其它符号。

1.4.3 布尔类型

布尔类型在变量类型定义中的标准标识符是 boolean,布尔型的直接量只有两个: true

(真)和 false (假)。因此布尔型变量的取值为这两个值之一,布尔型变量在程序中十分有用,他主要用于控制程序中语句的执行次序。

1.4.4 字符类型

字符类型在变量类型定义中的标准标识符是 char,一个字符是写在两个单引号之间的一个字符,例如:‘A’、‘a’、‘B’等。

字符型变量的值是一个字符,而不是一串字符,关于字符串处理将在结构数据类型中讨论。字符型变量的取值范围是 ASCII 字符集,所有字符均按其 ASCII 码值排序,例如,‘A’<‘B’,因为字符 A 的 ASCII 码为 65,而 B 的 ASCII 码值为 66。

§ 1.5 运算符和表达式

在程序设计中,常常需要描述各种运算式,这在程序设计语言中就是表达式。由变量、常量、函数、运算符和圆括号组成的有意义的式子就称为表达式。本节介绍如何用标准数据类型的量构成表达式,首先讨论 PASCAL 语言所允许的运算符。

1.5.1 运算符

按运算符的运算优先级,可以将运算符分为如下五类:

(一)一元减运算符

它表示取操作数的负值,操作数只有一个,或是整型数或是实型数。例如: -5, -0.5, -36 就是由一元减运算符构成的三个表达式。

(二)逻辑非运算符 NOT

它表示取布尔型操作数的逻辑非值。例如,NOT false 取 true,NOT true 取 false。

(三)乘法类运算符

此类运算符包括: *、/、DIV、MOD、AND,它们的功能、操作数类型、结果类型及其应用举例如表 1-1 所示。

表 1-1 乘法类运算符

运算符	功 能	操作数类型	结果类型	举 例
*	乘	实型和实型	实型	$1.1 * 1.7 = 1.87$
		整型和实型	实型	$3 * 1.1 = 3.3$
		整型和整型	整型	$11 * 12 = 132$
/	除	实型和实型	实型	$1.87 / 1.7 = 1.1$
		整型和实型	实型	$567.6 / 6 = 94.6$
		整型和整型	实型	$67 / 10 = 6.7$
MOD	取模	整型和整型	整型	$23 \text{ MOD } 5 = 3$
DIV	整除	整型和整型	整型	$23 \text{ DIV } 5 = 4$
AND	逻辑与	布尔型和布尔型	布尔型	$\text{true AND false} = \text{false}$

(四)加法类运算符

此类运算符包括: +、-、OR,它们的功能、操作数类型、结果类型及其应用举例如表 1-2 所示。

表 1-2

加法类运算符

运算符	功 能	操作数类型	结果类型	举 例
+	加	实型和实型	实型	$1.1+1.7=2.8$
		整型和实型	实型	$3+1.1=4.1$
		整型和整型	整型	$11+12=23$
-	减	实型和实型	实型	$1.87-1.7=0.17$
		整型和实型	实型	$567.6-60=507.6$
		整型和整型	整型	$67-10=57$
OR	逻辑或	布尔型和布尔型	布尔型	$\text{true OR false}=\text{true}$

(五)关系运算符。

此类运算符包括： $=$ (等于)、 $<>$ (不等于)、 $>$ (大于)、 $>=$ (大于等于)、 $<$ (小于)、 $<=$ (小于等于)。关系运算符可用于两个同种类型的量的比较(整型数和实型数可以放在一起比较)，其结果是逻辑值 true(真)或 false(假)。例如：

$5=5$	结果为 true
$5=10$	结果为 false
$\text{false} < \text{true}$	结果为 true
$\text{false} > \text{true}$	结果为 false
$'A' < 'C'$	结果为 true
$12.5 >= 3$	结果为 true
$18.5 <= 10.3$	结果为 false

运算符在表达式的计算中是有优先次序的，各种运算符的运算优先级如下：

- (1) 圆括号($\dots(\dots(\dots)\dots)\dots$)，按由内至外，逐层展开的规律进行。
- (2) 一元减： $-$ 。
- (3) 逻辑非：NOT。
- (4) 乘法类运算： $*$ 、 $/$ 、DIV、MOD、AND。
- (5) 加法类运算： $+$ 、 $-$ 、OR。
- (6) 关系运算： $=$ 、 $<>$ 、 $<$ 、 $<=$ 、 $>$ 、 $>=$ 。

优先级相同的运算符，按其在表达式中出现的次序从左向右进行。

1.5.2 表达式

从前面的举例可以看出，每个表达式都有一个值，根据此值的类型，可将表达式分为三类：算术表达式、字符表达式、布尔表达式。

(一)算术表达式

算术表达式相应于数学上的一个计算公式，它是由整型或实型的常量、变量、函数、算术运算符($*$ 、 $/$ 、DIV、MOD、 $+$ 、 $-$)及圆括号所组成的有意义的式子。单独的一个常量、变量或函数也可以看成是一个表达式。算术表达式的值只能是实型或整型的。

(二)字符表达式

因为对字符型量只能使用关系运算符，不能进行其他运算，因此，所谓字符表达式就是单独一个字符型常量、字符型变量或字符型函数。

(三)布尔表达式

在介绍布尔表达式的概念之前,先建立关系式的概念:用一个关系运算符将两个表达式连接起来的式子称为关系式。在关系运算符的举例中,我们能够发现关系式取布尔值(即取 true 或 false)。

由布尔型常量、布尔型变量、布尔型函数、关系式、逻辑运算符及圆括号组成的有意义的式子称为布尔表达式。单独一个布尔型常量、布尔型变量、布尔型函数或关系式都可看作是布尔表达式。每个布尔表达式最终都可得到一个布尔值。

§ 1.6 标准函数及其引用

在标准 PASCAL 语言中,共定义了 17 种标准函数,用以实现最通用的或其他一些特殊的函数运算。这些函数的名称已经确定,作为 PASCAL 语言中的标准标识符,建议用户不把它们用作用户自定义标识符。

用户不必对标准函数进行任何说明就可以直接引用,在引用标准函数时,要注意其自变量的类型与取值范围以及函数运算的结果类型。下面我们分四大类介绍标准函数的功能与使用要求,并用字母 x 表示自变量。

1.6.1 算术运算函数

(一)绝对值函数 $\text{abs}(x)$

其中 x 为整型数或实型数,函数值也为整型数或实型数。正数的绝对值就是自己,负数的绝对值是其相反数。例如: $\text{abs}(-5)=5$, $\text{abs}(67.89)=67.89$ 。

(二)平方值函数 $\text{sqr}(x)$

其中 x 为整型数或实型数,函数值也为整型数或实型数。例如: $\text{sqr}(1.2)=1.44$, $\text{sqr}(-9)=81$ 。

(三)平方根函数 $\text{sqrt}(x)$

其中 x 必须为正的整型数或正的实型数,函数值一律为实型数。例如: $\text{sqrt}(4)=2$ 。

(四)正弦函数 $\sin(x)$ 和余弦函数 $\cos(x)$

其中 x 为整型数或实型数,函数值一律为实型数。 x 必须用弧度制,不能直接用角度,例如: $\sin(3.14159/6)=0.5$, $\cos(3.14159/3)=0.5$ 。

(五)反正切函数 $\arctan(x)$

其中 x 为整型数或实型数,函数值一律为实型数且为弧度制。

(六)指数函数 $\exp(x)$

其中 x 为整型数或实型数,函数值一律为正的实型数。它是求以 e 为底的指数值。

(七)自然对数函数 $\ln(x)$

其中 x 必须为正的整型数或正的实型数,函数值一律为实型数。

1.6.2 逻辑判断函数

这里仅介绍奇数函数 $\text{odd}(x)$ 。其中 x 必须是整型数, x 为奇数值时,函数值为 true,否则为 false。