

高等学校教学用书



PASCAL

吴秉钧 编

PASCAL 语言程序设计

中国矿业大学出版社

PASCAL 语言程序设计

吴秉钧 编

中国矿业大学出版社

内 容 提 要

本书系统完整地介绍了标准 PASCAL 语言的内容,并介绍了 TURBO PASCAL 语言一些常用的扩展功能。全书以结构化程序设计思想为指导,贯彻了自顶向下逐步求精结构化的程序设计方法,结合具体实例,由浅入深,使学生在熟练掌握一门高级语言的同时又学习规范的程序设计方法,培养良好的程序设计风格。

本书为计算机应用专业本科或专科的教材,也适用于非计算机专业的研究生和本科生教学和应用能力等级考试需要,并且便于自学。

责任编辑 褚建萍 孙树朴

JS16/12

PASCAL 语言程序设计

吴秉钧 编

中国矿业大学出版社出版
新华书店经销 江苏省赣中印刷厂印刷
开本 787×1092 毫米 1/16 印张 15.75 字数 373 千字
1996 年 8 月第一版 1996 年 8 月第一次印刷
印数:1—4100 册

ISBN 7 - 81040 - 529 - 2

TP·30

定价:15.00 元

前 言

PASCAL 语言是一种结构化的程序设计语言。它具有语句简明、程序结构清晰、运算功能强等特点,深受广大计算机用户的喜爱。此外,PASCAL 语言还特别适合于教学。

对于计算机专业的学生,本课程是学习“数据结构”、“操作系统”等课程的必修先导课;对于其它各类专业学生,它也是学习规范程序设计的一门重要课程。

本书在系统完整地介绍标准 PASCAL 语言内容的同时,介绍了 TURBO PASCAL 语言一些常用的扩展功能。书中有丰富的实例和习题。全书贯彻结构化程序设计思想,采用逐步求精的设计方法,着意培养读者良好的程序设计风格。

从全书内容看,“控制语句”是基础,“过程和函数”是实现结构化程序设计的重要手段,而“构造数据类型”及“指针”是 PASCAL 语言的精华所在。

本课程具有极强的实践性,除掌握必要的基本概念及语法规则外,重要的是实践。一定要自己动手编写程序、上机操作,这是学好本课程最为有效的途径。

本书是作者在多年从事 PASCAL 语言教学实践的基础上编写而成的。力求内容简明扼要、由浅入深。所有实例都已经过上机检验。期望广大读者能从中有所收获。

目录中标记 * 号的章节可作为选学内容。

由于作者水平有限,书中错误及不恰当的提法在所难免。恳请读者及同行批评指正。

编 者

1996 年 6 月

目 录

第一章 概述	(1)
第一节 PASCAL 语言与结构化程序设计	(1)
第二节 PASCAL 程序的结构	(2)
第三节 PASCAL 语言的基本元素	(4)
第四节 PASCAL 语法图	(7)
习题	(8)
第二章 数据和表达式	(9)
第一节 常数	(9)
第二节 变量及其类型	(11)
第三节 运算符及表达式	(13)
习题	(15)
第三章 简单程序设计	(17)
第一节 赋值语句	(17)
第二节 数据的输入	(18)
第三节 数据的输出	(20)
第四节 程序举例	(23)
习题	(25)
第四章 控制语句程序设计	(27)
第一节 条件语句	(27)
第二节 GOTO 语句和标号说明	(34)
第三节 循环语句	(35)
第四节 程序举例	(46)
习题	(48)
第五章 函数和过程程序设计	(51)
第一节 函数	(51)
第二节 过程	(56)
第三节 全局变量与局部变量	(58)
第四节 数值参数和变量参数	(60)

第五节	嵌套、并列与递归·····	(62)
第六节	过程参数和函数参数·····	(71)
* 第七节	无类型参数·····	(74)
习题	·····	(75)
第六章	枚举与子界类型·····	(80)
第一节	枚举类型·····	(80)
第二节	子界类型·····	(85)
习题	·····	(88)
第七章	构造数据类型·····	(90)
第一节	数组类型·····	(90)
第二节	字符串类型·····	(101)
第三节	集合类型·····	(106)
第四节	记录类型·····	(111)
第五节	程序举例·····	(121)
习题	·····	(129)
第八章	文件类型·····	(132)
第一节	类型文件定义及文件变量的说明·····	(132)
第二节	类型文件操作·····	(133)
第三节	正文文件·····	(142)
第四节	无类型文件·····	(146)
第五节	程序举例·····	(147)
习题	·····	(163)
第九章	动态数据·····	(164)
第一节	指针类型、指针变量及标准过程 NEW 和 DISPOSE·····	(164)
第二节	链表·····	(167)
第三节	二叉树·····	(185)
第四节	程序举例·····	(190)
习题	·····	(196)
* 第十章	TURBO PASCAL 的嵌入、覆盖及绘图技术·····	(197)
第一节	嵌入技术·····	(197)
第二节	覆盖技术·····	(198)
第三节	图形技术·····	(203)
习题	·····	(216)

* 第十一章 单元.....	(217)
第一节 标准单元简介.....	(217)
第二节 单元与模块化程序设计.....	(218)
习题.....	(226)
附录一 ASCII 码表.....	(227)
附录二 集成环境下 PASCAL 程序的编辑、编译和运行.....	(228)
附录三 错误信息和代码.....	(230)
参考文献.....	(241)

第一章 概 述

PASCAL 语言是国内外日趋广泛应用的一种结构化的程序设计语言。本章简要论述 PASCAL 的发展及其优点、结构化程序的概念、PASCAL 程序的结构以及 PASCAL 语言的语言元素,使读者了解 PASCAL 语言的概貌。

第一节 PASCAL 语言与结构化程序设计

一、PASCAL 的特点及发展简介

PASCAL 程序设计语言由 Niklaus Wirth 教授于 1969 年在瑞士苏黎世联邦工学院研究成功。1971 年发表了 PASCAL 语言用户手册,它标志着 PASCAL 语言的诞生。1974 年发表了 PASCAL 语言的修订报告,1980 年、1983 年国际标准组织(ISO)又先后两次发表了关于 PASCAL 语言的建议草案及 PASCAL 语言国际标准。

本语言取名为 PASCAL 是为了纪念法国 17 世纪的一个数学家 Blaise Pascal(1623—1662)。

PASCAL 语言具有丰富的数据类型、简明的通用语句、清晰的程序结构、自由的书写格式,此外还具有运行效率高、查错能力强、移植性好等优点。

PASCAL 语言既适用于系统程序的设计,又适用于应用程序的设计;既可用作数值计算,又可用于非数值问题的处理。它已成为当今世界最广泛使用的程序设计语言之一。

PASCAL 是一种结构化的程序设计语言。它特别适合于教学,有利于培养学生结构化程序设计的良好习惯。数据结构、操作系统等一些课程又大多以此语言作为描述工具。因此,PASCAL 语言已成为大中专院校计算机专业教学的第一语言,而且也越来越广泛地成为各类理、工科专业学生学习程序设计的首选语言。

自从 1985 年美国 Borland 公司推出 TURBO PASCAL 3.0 以来,TURBO PASCAL 很快风行全球,深受广大计算机用户的喜爱。其版本不断更新,性能不断提高。概括地说,TURBO PASCAL 具有以下优点:

- (1) 除实现标准 PASCAL 外,功能上有很大的扩充。
- (2) 编译快、运行速度快。
- (3) 查错方便、快速。
- (4) 集编辑、编译、运行于一体。用户界面好,操作十分方便。

现在的 TURBO PASCAL 语言已可以用来编写任何类型、任何规模的程序。可以说,TURBO PASCAL 的出现是 PASCAL 发展的一个里程碑。

二、结构化程序设计

结构化程序设计的思想由著名荷兰学者 E. W. Dijkstra 于 1969 年首先提出。他针对当时所谓的“软件危机”(即软件的开发与满足需求的矛盾、软件价格的昂贵与质量低劣的矛盾以及对已开发软件的大量维护工作量)指出“GOTO 语句是有害的”。他认为 GOTO 语句是造成程序混乱不堪的祸根,是程序复杂性难以控制的主要原因。这一论断引起了计算机学术界的一场大争论。众所周知,GOTO 语句是一种控制程序执行流向的语句,是一种十分灵活的语句。但如果不加限制地使用 GOTO 语句,特别是逆向转移的 GOTO 语句,它会破坏程序静态结构与动态结构的一致性,降低程序的可读性、可靠性,增加程序调试、维护的难度。但是一概拒绝 GOTO 语句也是不可取的,比如使用 GOTO 语句实现从循环结构中提前退出,显然可以提高程序的执行效率。所以不能简单地从形式上是否使用 GOTO 语句来判定程序质量的优劣;而应以程序的可读性、可维护性、静态结构与动态结构的一致性作为判断的依据。结构化程序设计就是使程序具有合理的结构,以便保证和验证其正确性的程序设计方法,这便是结构化程序设计的基本思想。

结构化程序可以由顺序、分支、循环三种基本结构构成。它们都是单入口、单出口的。PASCAL 语言是一种良好的结构化程序设计语言,完全有能力取消 GOTO 语句,但基于上述原因,PASCAL 语言仍然保留了 GOTO 语句,只是限制了它的使用。

第二节 PASCAL 程序的结构

在系统地学习 PASCAL 语言之前,为使大家对 PASCAL 语言的程序结构有一个初步的认识,举例如下:

例 1-1 输入两个整数,计算它们之和,并输出。

```
PROGRAM EX1_1(INPUT,OUTPUT);           }程序首部
{THIS PROGRAM IS USED FOR S=X+Y}
VAR
    X,Y,S : INTEGER;                    }说明部分
BEGIN
    WRITE('ENTER X & Y : ');
    READ(X,Y);
    S := X+Y;
    WRITELN('S=',S)                     }语句部分
END.
```

}分程序

运行示例:

ENTER X & Y : 123 456✓

S=579

从上述程序可以看出一个程序是由程序首部、说明部分和语句部分三部分组成的。任何一个 PASCAL 程序,不管多么复杂或多么简单,都是这样的结构形式。程序中的字母可以大写,可以小写,或大小写混用。本书中的程序全部使用大写。

一、程序首部

程序首部是用来指出程序的名字,并列出本程序所用到的文件。它总是由保留字 PROGRAM 开头,后跟程序名(程序名由用户定义。本例程序名为 EX1-1)。在保留字 PROGRAM 和程序名之间应至少留有一个空格。紧跟程序名之后是程序参数表。参数表用一对圆括号括起来,各参数之间用逗号分隔,它指出了该程序需用到的文件,表示了该程序与外界的联系。INPUT 和 OUTPUT 是两个标准文件(它们对应的实际文件是键盘和显示器),它们表示程序中有输入和输出的操作。程序首部以分号(;)结尾。

程序首部在 TURBO PASCAL 中并非必须出现,它是可选的。通常采用如下形式,仅起文档作用。

```
PROGRAM 〈程序名〉;
```

二、说明部分

说明部分用于定义或说明程序要用到的标号、常量、类型、变量、过程和函数。它们的定义或说明分别冠以保留字 LABEL, CONST, TYPE, VAR, PROCEDURE 和 FUNCTION。一个程序的说明部分并不一定全部包括上述五项说明,要视问题的需要而定。如在程序 EX1-1 中,只有变量说明一项。

标准 PASCAL 规定,每一说明只可以出现一次或不出现,且必须严格遵守上述顺序。而在 TURBO PASCAL 中,这些说明可出现任意多次,而且顺序随意(若要说明自己定义类型的变量,这个类型定义必须在变量说明之前)。

三、语句部分

语句部分是在一对保留字 BEGIN 和 END 之间的部分,它们由一系列的语句组成。每一语句执行一定的动作,所以语句部分也称程序的执行部分。每两个语句之间用一个分号分隔,分号不是语句的一部分,所以在标准 PASCAL 中 END 和最后一个语句之间没有分号,但是在 TURBO PASCAL 中写上分号也并不错。在示例中每行只写一个语句,其实 PASCAL 允许一行写几个语句,也允许一个语句写成几行。书写格式自由、灵活是 PASCAL 的一个突出优点。但要注意,最后一个 END 必须以圆点结尾。

四、注释

程序中用花括号 '{' 和 '}' 括起来的内容为程序的注释。注释的内容在程序编译时,不产生目标码,因此,注释部分的有无对程序的执行不产生任何影响,但不要认为注释毫无用处。在一个程序中,尤其是一个复杂的程序中,加上注释可增加程序的可读性,也有利于程序的修改、调试和交流。注释可出现在程序中任何可以出现分隔符的地方。注释除了可使用花括号之外,在 TURBO PASCAL 中尚可使用符号对 '(*' 和 '*)'。

顺便指出,在一个程序的调试过程中,若觉得某个语句或某程序段有问题,可以用一对花括号将其括起来,这样,便可使其在程序中暂时不起作用,待问题找出后再决定其取舍。

一个程序工作者,除尽可能使自己编写的程序正确之外,保证程序的清晰与可读性也是十分重要的。在 PASCAL 程序设计中特别强调按层次构造的书写格式,希望读者一开始便

养成良好的习惯。

图 1-1 表示 PASCAL 程序结构语法规则。

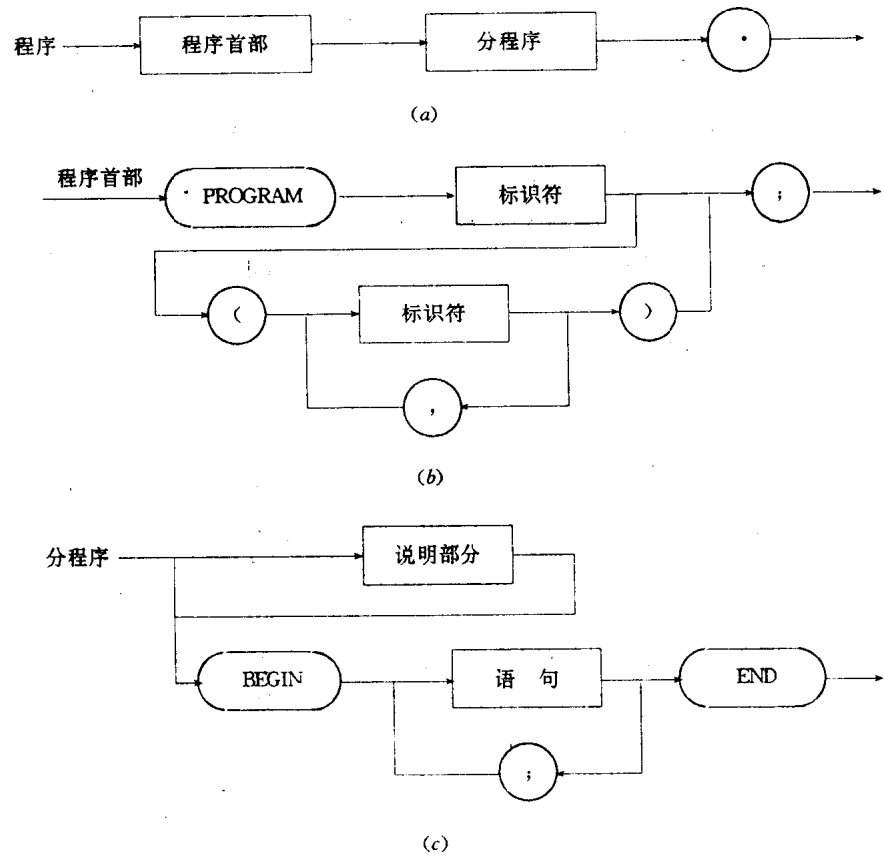


图 1-1 程序结构语法规则

第三节 PASCAL 语言的基本元素

PASCAL 有自己的一套基本元素,它们按照一定的语法规则构成 PASCAL 语句,进而构成 PASCAL 语言。

一、基本字符

基本字符是由字母、数字和特殊字符构成,它们均属于 ASCII 码字符集中的字符。

字母:大写字母 A~Z,小写字母 a~z。

数字:0~9。

特殊字符:

单符:

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

_(下划线)

双符:

: = < > <= >= ..

符号#, \$ 及下划线_只允许在 TURBO PASCAL 中使用。此外,可以用(* *)及(.)分别代替{ } 和[]。

二、保留字

保留字(又称关键字)是英文单词或它们的缩写,它们具有特殊的含义。因此整个保留字需作为一个不可分割的符号使用,不允许对它们重新定义。保留字的作用是用来命名 PASCAL 语句,表示某些运算符,描述某些数据类型,说明程序、过程及函数首部的。

在标准 PASCAL 中保留字共 36 个;但 TURBO PASCAL 的保留字有所扩充。以下所列的保留字打有 * 号的是 TURBO PASCAL 扩充的保留字。

* ABSOLUTE	END	* INLINE	PROCEDURE	TYPE
AND	* EXTERNAL	* INTERFACE	PROGRAM	* UNIT
ARRAY	FILE	* INTERRUPT	RECORD	UNTIL
BEGIN	FOR	LABEL	REPEAT	* USES
CASE	FORWARD	MOD	SET	VAR
CONST	FUNCTION	NIL	* SHL	WHILE
DIV	GOTO	NOT	* SHR	WITH
DO	IF	OF	* STRING	* XOR
DOWNTON	* IMPLEMENTATION	OR	THEN	
ELSE	IN	PACKED	TO	

三、标识符

程序中要用到许多名字,诸如标号、常量、变量、类型、过程、函数等,标识符就是用来标记名字的。标识符的形成规则是:它是以字母或下划线开头的字母、下划线、数字的序列。下划线用于标识符在标准 PASCAL 中是不允许的。标识符的长度是任意的,但不同的版本对标识符有效长度的限制是不同的。标准 PASCAL 仅前 8 个字符有效,但 TURBO PASCAL 的有效长度为 63 个字符。

标识符分为两类:标准标识符及用户定义标识符。

1. 标准标识符

标准标识符是 PASCAL 系统预定义的标识符。这些标识符都具有特定的含义,在程序中可以直接使用。标准 PASCAL 共有 40 个标准标识符,它们用以标准常量、标准类型、标准过程、标准函数及标准文件的标识。现分列如下:

标准常量:FALSE,TRUE,MAXINT。

标准类型:INTEGER,REAL,CHAR,BOOLEAN,TEXT。

标准文件:INPUT,OUTPUT。

标准函数:ABS,ARCTAN,CHR,COS,EOF,EOLN,EXP,LN,ODD,ORD,
PRED,ROUND,SIN,SQR,SQRT,SUCC,TRUNC。

标准过程:DISPOSE,GET,NEW,PACK,PAGE,PUT,READ,READLN,

RESET,REWRITE,UNPACK,WRITE,WRITELN。

表 1-1 列出了标准 PASCAL 语言定义的 17 种标准函数。用户可以直接使用。

表 1-1 标准函数及应用举例

函数名	函数值	应用举例
ABS(X)	X	ABS(-13.2)=13.2 ABS(13.2)=13.2
ARCTAN(X)	TG ⁻¹ X	ASCTAN(1.0)=0.785398 即 45°
CHR(X)	序号为 X 的 PASCAL 字符	CHR(65)='A'
COS(X)	COSX	COS(3.1416/180 * 30)=0.8661
EOF(X)	TRUE 或 FALSE	读到文件结束值为真(TRUE)否则为 FALSE
EOLN(X)	TRUE 或 FALSE	读到文件行结束为真(TRUE)否则为 FALSE
EXP(X)	e ^x	EXP(0.7419373)=2.1
LN(X)	LNx	LN(2.1)=0.7419373
ODD(X)	X 为奇数时得 TRUE X 为偶数时得 FALSE	ODD(15)=TRUE,ODD(16)=FALSE
ORD(X)	X 的序号	ORD('A')=65,ORD(TRUE)=1
PRED(X)	X 的前序值	PRED(5)=4,PRED(TRUE)=FALSE
ROUND(X)	最接近 X 的整数	ROUND(-5.7)=-6,ROUND(18.4)=18
SIN(X)	SINX	SIN(3.1416/180 * 30)=0.5000
SQR(X)	X ²	SQR(1.2)=1.44
SQRT(X)	$\sqrt{X}$	SQRT(1.44)=1.2
SUCC(X)	X 的后继值	SUCC(-3)=-2 SUCC('h')='i'
TRUNC(X)	X 的整数部分	TRUNC(3.7)=3 TRUNC(6.2)=6

表 1-2 给出了标准函数自变量与函数值的类型。需要注意的是函数 SIN 及 COS 的自变量可以是整型或实型,但是角度必须用弧度表示。

表 1-2 标准函数自变量与函数值类型表

标准函数 函数值类型	自变量类型	整 数 (INTEGER)	实 数 (REAL)	字 符 (CHAR)	布尔型 (BOOLEAN)	文 件 (FILE)
整 数 (INTEGER)		ABS SQR PRED SUCC	TRUNC ROUND	ORD	ORD	
实 数 (REAL)		SIN COS ARCTAN LN EXP SQRT	SIN COS ARCTAN LN EXP SQRT ABS SQR			
字 符 (CHAR)		CHR		PRED SUCC		
布尔型 (BOOLEAN)		ODD			PRED SUCC	EOF EOLN

2. 用户定义标识符

用户定义的标识符用来为程序设计者根据自己的需要定义的常量、变量、类型、过程、函数及文件等取名。用户在定义标识符时应注意：

(1) 禁止使用保留字作为标识符。

(2) 尽量避免使用标准标识符作为用户定义标识符。虽说标准标识符允许重新定义，但这样做会失去它们原来的意义，导致混乱，可能给自己带来预想不到的麻烦。

(3) 遵守先定义后使用的原则。即只有在说明部分定义了标识符，才可在语句部分使用。

下面是一些标识符的示例。

合法标识符：

A A1 SYMBOL SQUARE EX1-1

不合法标识符：

1A(不能以数字开头) BEGIN(禁止使用保留字)

TWO WORDS(不能有空格) NO.(不允许字符圆点)

四、分隔符

空格、逗号、分号及行结束符称为 PASCAL 的分隔符。PASCAL 规定任意两个相邻标识符、数、保留字或两个语句之间必须至少有一个分隔符，以便编译程序能识别。为便于阅读，程序也需要如同自然语言一样，恰当地使用分隔符。值得指出的是，这些分隔符不能互相代用，即该用空格的地方只能用空格，该用逗号的地方只能用逗号，…。

现有一程序段

```
PROGRAM ADD;
```

```
VAR
```

```
XY: INTEGER;
```

```
BEGIN
```

```
READ(X,Y);
```

```
Z:=X+Y;
```

```
;
```

上列程序段的错误之处：

(1) 保留字 PROGRAM 不恰当地插入了空格。

(2) 由于变量说明时，X 和 Y 之间没有用逗号分隔，所以编译程序将它们当作一个变量标识符。因此，在执行第一个语句时，变量 X 和 Y 便成了未定义的标识符。

(3) 变量 Z 是一个未定义的标识符。

第四节 PASCAL 语法图

PASCAL 的语法规则可以通过它的语法图来描述。PASCAL 涉及的每一个语法概念都可通过语法图给以直观、精确、唯一的解释。

语法图中的圆圈表示 PASCAL 的保留字、运算符或规定的其它特殊符号。带圆弧的长

条框表示 PASCAL 的保留字或无需进一步定义的语法实体。长方框表示由其它语法图定义的语法实体。

图 1-2 的语法图表示的是标准 PASCAL 中标识符的语法规则(标准 PASCAL 中标识符不能使用下划线)。

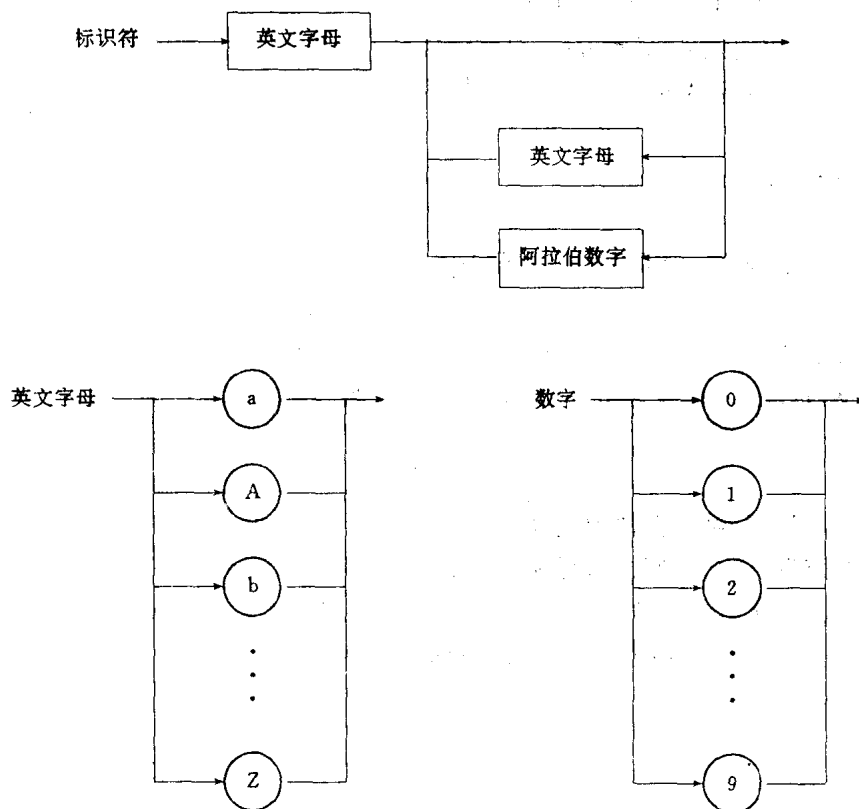


图 1-2 标识符的语法规则

习 题

1-1 PASCAL 语言主要有哪些特点?

1-2 什么是结构化程序设计? 怎样评价 GOTO 语句?

1-3 PASCAL 语言的程序结构有几部分组成?

1-4 TURBO PASCAL 中用户定义标识符的原则是什么? 指出下列字符串中哪些是标识符? 哪些不是? 为什么?

ABC; BEGIN; -AB; EX1_1; π ; 12A; A12B; N. Wirth; a&b;
TURBO PASCAL;

1-5 花括号 '{' 和 '}' 在程序中的作用是什么?

第二章 数据和表达式

在 PASCAL 中,数据的各种不同性质用数据类型表示。程序中的每一数据项必定与某一种数据类型相联系,数据类型不仅规定了该数据的取值范围,而且也规定了它们能执行的运算。

PASCAL 有丰富的数据类型,可以分为三大类:简单类型、构造类型和指针类型。如图 2-1 所示。本章仅介绍简单类型中的标准类型。

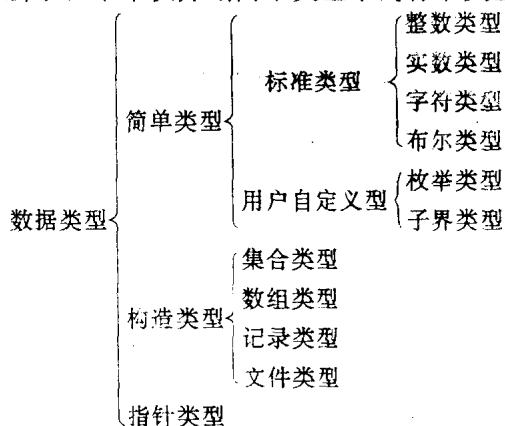


图 2-1 PASCAL 数据类型

第一节 常 数

常数是指可以直接参与运算的量。所以又称为直接量。

一、整数和实数

数有整数和实数两种。在 TURBO PASCAL 中,整数除了可用十进制(数字 0~9)表示外,还可以用十六进制(数字 0~9,大写字母 A~F,或小写字母 a~f)表示。当用十六进制表示数时,需在数前面加前缀 \$。

下列表示的整数都是合法的:

123 -345 0 +2468

\$12 (相当于十进制数 18) \$12A (相当于十进制数 298)

一个十进制整数通常等于一个机器字长。目前微型机字长多为 16 位。故十进制整数的范围是 -32768 到 32767。十六进制数的范围是 \$0000 到 \$FFFF。整数的取值范围较小,但它是一个精确表示,而且计算机处理整数的速度比实数要快,所以一个数若能表示为整数,就不应表示为实数。但若一个数超过了整数的表示范围,或确定是一个实数,则可用实数表示。对于字长为 16 位的计算机,实数的表示范围在 10^{-38} 到 10^{38} 之间。实数有两种表示形式,

即十进制表示法(又称习惯表示法)和指数表示法(又称科学表示法)。

以下是正确的习惯表示法:

-3.5 0.0 123.45 +678.9

要注意,采用习惯表示法时,小数点的两侧都必须有数字,缺一不可。因此+123.;.56都是不合法的。

以下是正确的科学表示法:

6.25E+0.8	(=6.25×10 ^{0.8})
-51E-1	(=-51×10 ⁻¹)
0E0	(=0×10 ⁰)
+0.6378E2	(=0.6378×10 ²)
-1.2345678901E+12	(=-1.2345678901×10 ¹²)
尾数 阶码	

这里字母 E 表示以 10 为底的指数。采用科学法表示时应注意,尾数必须有,但小数部分可无;阶码必须有,且必须是整数。因此,下列表示都是不正确的。

E-6	(缺尾数)
3E	(无阶码)
2.E3	(尾数出错)
2E1.2	(阶码出错)

二、字符及字符串

字符常数就是用两个单引号括起来的一个 ASCII 码字符集中的字符。例如:'A';'a';
'2';'";'┐'。

这里'A'与'a'分别表示大写字母 A(其 ASCII 码值为 65)及小写字母 a(其 ASCII 码值为 97),即作为字符数据是要区分大小写的。'2'是字符 2(其 ASCII 码值是 50),而不是整数 2。表示字符单引号本身时,必须重复书写一次。空格也是一个字。

一个字符串常数是括在两个单引号之间的字符序列。若两个单引号之间没有任何字符,则为空串。TURBO PASCAL 允许在字符串中出现控制字符。控制字符有两种形式,一种是在符号'#'后跟一个 0~255 之间的整数。它表示相应的 ASCII 码值字符。如:#13 表示字符 CR(回车),#27 表示字符 ESC。另一种是在符号'^'后跟一个字符,它表示相应的控制字符,如^G 表示字符 BEL,即响铃。

以下是字符串的一些例子:

```
'This is a string constant';  
'TURBO PASCAL';  
'You'll see';  
' '  
'Waiting for input!' ^G ^G ^G 'please wake up';
```

三、布尔常数

布尔常数只有两个:FALSE(假)及 TRUE(真)。它们是有序的,且 FALSE<TRUE。