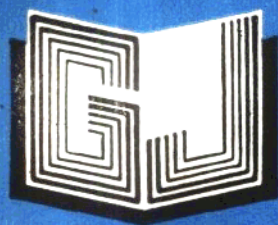


990342



高等学校教材

专科适用



PASCAL语言和程序设计

南京电力高等专科学校 史玉良 主编

2PA



前 言

本教材系按电力工业部高校 1993~1995 年专科教材编审出版计划,由中国电力企业联合会教育培训中心及电力高等专科学校教育委员会征稿并推荐出版的。

为适应国家教委对普通高等专科学校教育的要求及专科教学改革的新形势,即:加强理论知识的应用和实践动手能力的培养,基础理论以必须够用为度,以掌握概念、强化应用为教学的重点,本课程将原先的两门课《PASCAL 语言》与《数据结构》,进行调整,降低难度,精减内容,注重实用,合并成现在的一门课《PASCAL 语言和程序设计》,并选择 Turbo Pascal 5.5 系统为软件环境。因此本教材的内容除了 PASCAL 语言之外,还包括数据结构的四章内容,即线性表、二叉树、查找与排序,以及 Turbo Pascal 5.5 系统的集成开发环境、扩充功能、System、Crt、Dos、Graph 单元(附录 A、B、C)。

本课程是实践性较强的课程,为了加强实践性教学环节,书中给出大量 PASCAL 语言的例题,对所有算法均给出相应的 PASCAL 语言程序。所有程序均在 Turbo Pascal 5.5 系统中调试通过。

本教材的编写大纲由该书主编南京电力高等专科学校史玉良副教授提出。该书主审由沈阳电力高等专科学校高天元先生担任。由于史玉良老师于 1992 年退休,所以委托原计划的参编南京电力高等专科学校的叶核亚老师完成全部书稿的编写工作。对史玉良老师和高天元先生在工作中给予的指导和帮助,在此表示深深的谢意。同时,对所有在编写过程中给予帮助的同志,表示衷心的感谢。

由于编者水平有限,书中难免有错误和缺点,恳请读者批评指正。

叶核亚

1995 年 4 月

377 24/07

目 录

前 言

第一章 计算机系统与程序设计概述	1
第一节 计算机的硬件组成与工作原理	1
第二节 软件	3
第三节 程序与程序设计语言	3
第四节 二进制基础	5
习题一	9
第二章 PASCAL 语言基础	10
第一节 PASCAL 语言特点	10
第二节 PASCAL 语言程序的组成	10
第三节 语法描述	12
第四节 标准数据类型及标准函数	15
第五节 常量定义与变量说明	21
习题二	23
第三章 简单程序设计	24
第一节 语句	24
第二节 表达式	25
第三节 赋值语句	29
第四节 读语句	29
第五节 写语句	31
习题三	35
第四章 程序控制结构	37
第一节 复合语句	38
第二节 分支结构与条件语句	39
第三节 循环结构与重复语句	44
第四节 转向语句	57
习题四	58
第五章 自定义数据类型	60
第一节 类型定义	60
第二节 枚举类型与子界类型	61
第三节 数组类型	66
第四节 字符串类型	77
第五节 集合类型	82
第六节 记录类型	89

习题五	97
第六章 过程与函数	100
第一节 过程	100
第二节 函数	104
第三节 全程量与局部量	110
第四节 参数传递	113
第五节 嵌套与递归	116
习题六	122
第七章 文件类型	124
第一节 类型文件	124
第二节 类型文件的随机操作	131
第三节 文本文件	134
习题七	137
第八章 指针类型	138
第一节 静态变量与动态变量	138
第二节 指针变量	138
习题八	143
第九章 线性表	145
第一节 线性表概念	146
第二节 线性链表	150
第三节 栈	166
第四节 队列	175
习题九	180
第十章 二叉树	182
第一节 树	182
第二节 二叉树的定义及性质	183
第三节 二叉树的遍历	188
第四节 二叉排序树	200
第五节 线索二叉树	205
习题十	210
第十一章 查找	212
第一节 顺序查找	213
第二节 折半查找	214
第三节 分块查找	216
第四节 哈希查找	220
习题十一	227
第十二章 排序	228
第一节 插入排序	228
第二节 交换排序	236
第三节 选择排序	240

第四节 归并排序	245
第五节 排序算法性能评价	249
习题十二	249
附录 A Turbo Pascal 5.5 系统集成开发环境	251
附录 B Turbo Pascal 5.5 系统扩充功能介绍	265
附录 C Turbo Pascal 5.5 单元	273
附录 D ASCII 码表	291
附录 E 编译与运行出错信息	292
附录 F Turbo Pascal 5.5 系统库函数一览表	297
参考书目	300

第一章 计算机系统与程序设计概述

电子计算机是一种由程序控制操作过程，具有内部存储能力，能够自动地、高速地进行大量数字与逻辑运算的装置。它的发明与不断发展是 20 世纪科学技术的卓越成就之一。今日的计算机已不再单纯用于科学计算，它已广泛应用于数据处理、自动控制、辅助设计、系统管理、数字通信、图像处理、人工智能等多个领域，甚至已步入了人们的家庭，有力地推动着生产、科学技术与人类文明的发展。因此，具备计算机知识和应用计算机的能力是现代知识结构中不可缺少的一个重要部分。

第一节 计算机的硬件组成与工作原理

计算机系统是计算机硬件 (hardware) 和软件 (software) 的总称。硬件是由电子线路和机电装置组成的硬设备。软件则是指人们为解决各种问题而预先设计好的通过硬件执行的各种指令组成的程序。硬件是完成软件功能的物质基础。

计算机虽然是一部非常复杂的机器，但其硬件构成却和人用算盘计算具有十分相似之处。用算盘计算，参与的有算盘、纸、笔和人。算盘是运算部件，纸笔是用来保存和记录数据及结果的存储部件，而人是控制部件，负责协调各部件有秩序有节奏地工作。计算机与此相似，也是由运算器、控制器和存储器三个主要部件组成。这是计算机的主机部分，其中运算器和控制器合称为中央处理机 (Centre Process Unit, 简称 CPU)。除了主机部分，计算机还必须具有输入输出设备及外部存储器，它们是计算机的外围设备。上述部件的结构关系可用图 1-1 表示。

一、中央处理机

中央处理机是计算机的核心部件，由运算器和控制器两部分组成，它包含指令的解释和执行线路，以及为执行指令所必需的运算、逻辑和控制线路。

运算器又称算术和逻辑部件，它能完成算术运算或逻辑运算。运算过程中，运算器在控制器的控制下，从内部存储器 (简称内存) 中取得数据，经过运算又把结果送回到内存中保存。这一切都是通过执行存放在内存中的指令序列来完成的。CPU 的运算速度是衡量计算机性能的最重要指标。

控制器用于控制计算机各个功能部件协调有序地工作。主要功能有：

(1) 控制运算器和内存间数据传送过程，自动协调地执行指令系统规定的运行动作，包括从内存取指令、翻译指令、启动程序执行以及回送数据到内存等等。

(2) 控制计算机外围设备与主机之间的信息传送过程。例如，将程序和数据由输入设备传送到内存，或将程序和数据传送到输出设备输出，以及控制内存存储器与外存储器之间的数据信息交换。

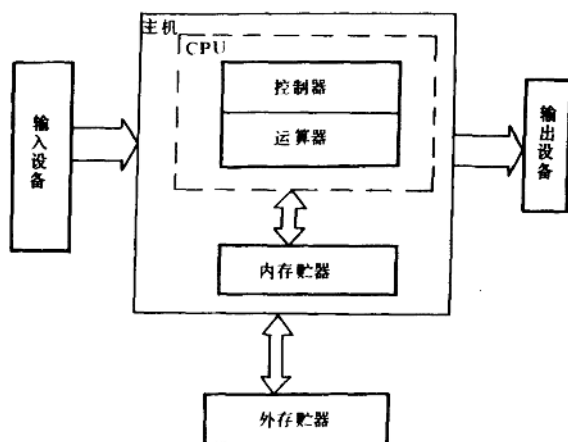


图 1-1 计算机硬件结构图

二、存储器

存储器是计算机系统中用于保存数据和程序的必不可少的功能部件。指令和数据都是以二进制代码形式保存在存储器中。

存储器具有接收、保存和发送代码的功能。存储容量越大，能容纳的代码量就越多。各种类型的信息，如指令、数值、文字、表格、图形等均可用二进制代码形式存放于存储单元中。代码一旦存入，就一直保存着，它可以多次读出，直到再次向同一单元存入新的代码时，原有的内容才会被冲掉。这就是计算机的记忆功能。

存储器分为主（内）存储器与辅助（外）存储器。内存与中央处理机（CPU）直接相连，构成计算机系统的主机部分，因而 CPU 可以对内存进行快速、直接地访问。

内存的基本单位称为存储单元，与每个存储单元相对应有一个地址，地址是有序的二进制编码，便于查找和访问。

CPU 按存储单元的地址可以对单元的内容进行直接访问，因而 CPU 从内存存取数据十分快捷，内存容量通常因机型不同多少不等，它亦是衡量计算机性能的重要指标之一。

内存通常只在通电情况下才能存储数据，一旦掉电，数据将丢失，再启动后，内存将被刷新。而外存在掉电后仍能保存数据。目前常用外存是磁盘，包括硬盘、软盘。磁盘上的数据可以长期保存。

三、输入输出设备

输入输出设备（Input/Output device，简称 I/O 设备）是人和计算机进行通信的纽带。输入设备用来输入计算机程序和数据，常用的输入设备包括键盘、鼠标器、数字化仪等。输出设备则用来输出计算结果。常用的输出设备有显示器（CRT）、打印机、绘图仪等。

另外，各种计算机的通信设备也属于 I/O 设备范畴。

四、个人计算机的硬件配置

个人计算机（Personal Computer），也称 PC 机，是应用最广泛的一种微型计算机。一

台 PC 机通常有以下硬件配置：

- 主板——PC 机的核心必备部件，包括 CPU、内存等。
 - I/O 板——主要有显示卡（必备，用来控制显示器），I/O 控制卡（如磁盘卡、串并卡等）。
 - 磁盘驱动器——包括硬盘驱动器和软盘驱动器。两者必备其一。
 - 电源——必备件，为上述三个部件供电。
- 上述四种通常组装在一个机箱内，构成 PC 机的主机。
- 显示器——显示设备，必备。
 - 键盘——输入设备，必备。
 - 打印机——输出设备。

此外，用户还可根据需配备鼠标器、数字化仪、绘图仪、光盘机等各种 I/O 设备，也可与其他 PC 机连接成网络等。

第二节 软 件

计算机系统中仅有硬件设备是无法工作的。为了运用计算机来解决各种问题，必须编制由指令组成的各种程序，用以控制计算机硬件各部件的协调工作。这些程序通称为软件。计算机软件系统的内容非常丰富，具有层次结构，如图 1-2 所示。

软件通常分为系统软件与应用软件两大类。系统软件是为了保证计算机系统运行的必备程序集合，最底层最重要的是操作系统（Operation System，简称 OS），OS 直接对计算机的硬设备资源进行管理和控制，并能协调各部件之间的并行操作，使计算机能够准确、实时、高效地完成各项任务。用于微机上的操作系统通常称为磁盘操作系统（Disk Operation System，简称 DOS），广泛应用的有 MS-DOS。

系统软件还包括一些在 OS 基础上运行的公用程序、编译程序等，如各种编辑服务程序、函数库、软件包等。

应用软件是用户根据计算机的应用领域和具体工程项目而开发研制的实用软件，如工业实时控制系统，数据采集分析系统，图书档案的数据库管理系统，电力系统的计算机监控调度系统，等等。

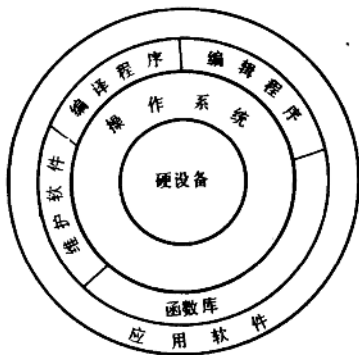


图 1-2 计算机软件层次图

第三节 程序与程序设计语言

程序是为解决问题而设计的指令序列。是借助于某种程序设计语言来描述对数据的操

作和控制。

计算机语言是人与计算机之间进行“对话”的工具。从其发展过程看，可分为三类：机器语言、汇编语言与高级语言。

一、机器语言

每一台计算机都有它自己的一套指令系统，这是在设计时所确定的，是计算机所固有的。指令系统中的每条指令称为机器指令。一般来说，机器指令告诉 CPU 进行操作的操作码和操作数的地址码。一条机器指令对应着一种基本操作数的地址码。一条机器指令对应着一种基本操作。机器指令的形式是由 0 和 1 组成的二进制代码。

机器语言是机器指令的集合，它是面向机器的，因而机器语言程序是针对该计算机的，由其特定的机器指令序列组成。机器语言的缺点是代码不易记住，书写不方便，易读性差，通用性差。

二、汇编语言

将机器指令的代码用助记符表示，则构成汇编语言。汇编程序 (Assembler) 在运行前首先进行“汇编”的过程，即将汇编程序“翻译”成机器指令，再交由 CPU 执行相应操作。

汇编语言虽与机器紧密联系，移植性差，但在某些场合仍非常适用，如对 DOS 的功能调用等。

三、高级语言

50 年代末，发明了“程序设计语言”，即高级算法语言。它是由各种类似自然语言的、具有不同意义的符号和表达式，按照一定的语法规则组成的。高级语言的语句功能强，一个语句相当于许多条机器指令的功能，因而用高级语言编写程序更方便容易。一般来说，高级语言是面向过程的，独立于机器，移植性强，易读写。

高级语言程序的执行过程要先经过相应的编译程序 (Compiler) “翻译”成机器指令 (目标代码)，然后由 CPU 运行。图 1-3 描述了程序的运行过程。

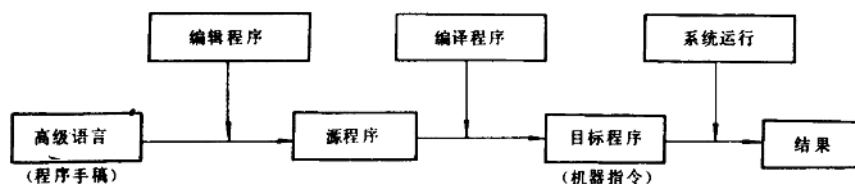


图 1-3 高级语言程序的运行过程

目前较通用的高级语言有：

- BASIC 会话式语言；
- FORTRAN 面向科学计算语言；
- COBOL 面向商业数据处理的语言；
- PASCAL 结构化的系统程序设计语言；
- C 具有高级数据结构与控制结构，亦可调用底层功能的通用语言；
- LISP 符号处理语言；

- PROLOG 用于人工智能的专家系统的逻辑型语言;
- ADA 分布式系统控制语言。

上述高级语言中,大多采用编译方式运行。只有早期的 BASIC 采用解释方式。即在执行 BASIC 语言源程序时,当一条语句由 BASIC 解释程序“翻译”成机器指令并执行完之后,再进行下一条语句的解释执行。这种方法,程序运行的速度较编译方式要慢些。

第四节 二进制基础

计算机是以二进制运算为基础的,硬件实现是以二进制为基础的数字逻辑电路。但二进制数不直观,书写不方便。为了书写方便,简化表达式,引进了八进制、十六进制等。同时数据的输入输出形式必须与人们日常使用的十进制相一致,这样在计算机中就存在多种进位制。

一、进位计数制

将数码按先后位置排列成数位,并按由低到高的进位方式进行计数称为进位计数制。

每种进位计数制都有两个基本要素,即基数和位权。基数是指计数制中所用到的数码个数。如十进制数 (Decimal), 数码是 0, 1, ..., 9 共 10 个, 它的基数是 10。在一个十进制数的每一位只能取 0, 1, ..., 9 的其中之一。再有同一数码在不同数位时, 其值不等。例如十进制的 121.5 百位的 1 值为 $1 \times 10^2 = 100$, 个位的 1 值为 $1 \times 10^0 = 1$ 。每个数码处在某个数位时, 它代表的数值是该数码本身的值乘以与所处数位相关的一个固定常数, 这个固定常数称为“位权”, 简称权, 它是以 10 为底的指数, 幂是数位的序数减 1。因此十进制的 121.5 可以展开成:

$$121.5 = 1 \times 10^2 + 2 \times 10^1 + 1 \times 10^0 + 5 \times 10^{-1}$$

一般形式为

$$S_{10} = \sum_{i=-m}^{n-1} k_i \cdot 10^i$$

其中 k_i 为第 i 位的数码, n 为整数的位数, m 为小数的位数。

上式可推广到任何进位计数制。

二、常用数制

1. 二进制数 (Binary)

二进制只有 0 和 1 两个数码。它的基数为 2, 可表示为:

$$S_2 = \sum_{i=-m}^{n-1} k_i \cdot 2^i$$

其中 k_i 可取 0 或 1。例如 1011.01 按权展开为:

$$(1011.01)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$$

在二进制的加减运算中,“逢二进一,借一为二”。二进制数不直观,书写不方便。

一个二进制数字称为一位 (bit), 8 个 bit 称为一个字节 (byte), 2 个或 4 个 byte 称为一个字 (word)。

2. 八进制数 (Octal)

数码是 0, 1, ..., 7 共八个, 基数是 8, 可表示为

$$S_8 = \sum_{i=-m}^{n-1} k_i \cdot 8^i$$

其中 k_i 可取 0, 1, ..., 7 这八个数字之一。在加减运算中, “逢八进一, 借一当八”。

3. 十六进制数 (Hexadecimal)

数码是 0, 1, ..., 9, A, B, C, D, E, F 共十六个, 十六进制的基数是 16。其中 A ~ F 相应的十进制数为 10 ~ 15。可表示为

$$S_{16} = \sum_{i=-m}^{n-1} k_i \cdot 16^i$$

其中 k_i 可取 0, 1, ..., 9, A, ..., F 之一。加减运算中, “逢十六进一, 借一当十六”。

表 1-1 列出四种进位制的 0 ~ 16 的数码对照表。

表 1-1 四种进制的数码对照

十进制 D	八进制 O	十六进制 H	二进制 B	十进制 D	八进制 O	十六进制 H	二进制 B
0	0	0	0	9	11	9	1001
1	1	1	1	10	12	A	1010
2	2	2	10	11	13	B	1011
3	3	3	11	12	14	C	1100
4	4	4	100	13	15	D	1101
5	5	5	101	14	16	E	1110
6	6	6	110	15	17	F	1111
7	7	7	111	16	20	10	10000
8	10	8	1000				

三、常用数制之间的转换

1. 十六进制数与二进制数之间的转换

将四位二进制数位对应一位十六进制数, 进行相互转换。不足时补 0。如 $(0001101000101111)_2 = (1A2F)_{16}$

$$\begin{array}{cccc} \underbrace{0001}_1 & \underbrace{1010}_A & \underbrace{0010}_2 & \underbrace{1111}_F \end{array}$$

2. 二进制数 (或十六进制) 转换成十进制数

采用“按权相加法”, 即将二进制数 (或十六进制数) 按权展开相加, 可得到相应的十进制数。如:

$$\begin{aligned} & (10110101)_2 \\ &= (1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0)_{10} \\ &= (128 + 32 + 16 + 4 + 1)_{10} \\ &= (181)_{10} \end{aligned}$$

或者:

$$\begin{aligned} & (10110101)_2 \\ &= (B5)_{16} \end{aligned}$$

$$= (11 \times 16^1 + 5 \times 16^0)_{10}$$

$$= (181)_{10}$$

3. 十进制转换成二进制（或十六进制数）

将十进制数转换成二进制数（或十六进制）时，整数部分采用“除 2（或 16）取余法”小数部分采用“乘 2（或 16）取整法”。

“除 2（或 16）取余法”，就是用 2（或 16）不断地去除要转换的十进制数，直到商为 0 时为止。将所得各次余数，以最后所得余数为最高数位，最先所得余数为最低数位的顺序排列，即为相应的二进制数（或十六进制数）。例如将十进制数 14 转换成二进制数。用“除 2 取余法”，过程如下：

2	14	余数=0	(最低数位) (最高数位)
2	7	余数=1	
2	3	余数=1	
2	1	余数=1	
	0		

$$\text{即 } (14)_{10} = (1110)_2$$

“乘 2（或 16）取整法”，就是用 2（或 16）不断地去乘要转换的十进制纯小数，直到满足精度或小数部分等于 0 为止。将各次所得的乘积以最先所得的为最高数位，最后所得的为最低数位的顺序排列，即为相应的二进制数（十六进制数）。例如将十进制数 0.5803 转换成二进制数，要求有效位数为 6 位，用“乘 2 取整法”，过程如下：

0.5803			(最高数位) (最低数位)
× 2			
1.1606	整数=1		
0.1606			
× 2			
0.3212	整数=0		
× 2			
0.6424	整数=0		
× 2			
1.2848	整数=1		
0.2848			
× 2			
0.5646	整数=0		
× 2			
1.1392	整数=1		

即 $(0.5803)_{10} = (0.100101)_2$

类似地，也可将十进制数转换成十六进制数。

三种常用数制之间的转换用图 1-4 表示。

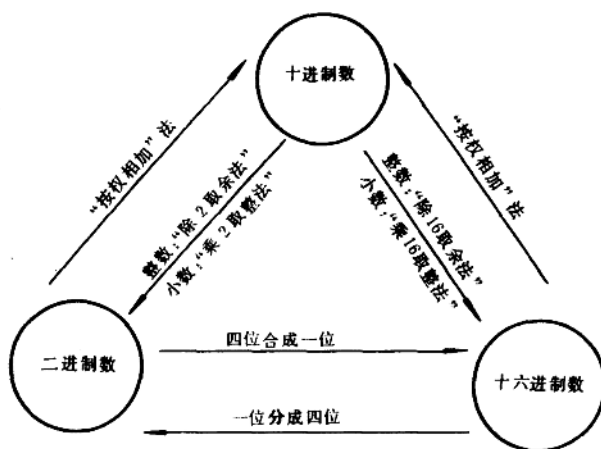


图 1-4 三种常用数制之间的转换

四、二进制的运算规则

由于二进制数只有 0 和 1 两个数码，因此它的加、减、乘、除等算术运算规则比十进制的运算规则简单得多。这也是计算机中采用二进制数主要原因之一。

1. 加法运算规则

二进制数的加法运算规则如下：

$$\begin{aligned} 0+0 &= 0 \\ 0+1 &= 1+0 = 1 \\ 1+1 &= 10 \quad (\text{有进位}) \\ 1+1+1 &= 11 \quad (\text{有进位}) \end{aligned}$$

例如： $10110101 + 00011011 = 111010000$

$$\begin{array}{r} 10110101 \\ + 00011011 \\ \hline 11010000 \end{array}$$

两个二进制数相加，每一位要考虑将两个加数和低位来的进位进行加法运算，得到本位的值和向高位的进位。相加的次序由低向高位，即与十进制运算次序相同从最右边开始，依次向左逐位进行运算。

2. 减法运算规则

二进制数的减法运算规则如下：

$$\begin{aligned} 0-0 &= 0 \\ 1-0 &= 1 \\ 1-1 &= 0 \\ 0-1 &= 1 \quad (\text{有借位}) \end{aligned}$$

例如: $11010000 - 00011011 = 10110101$

$$\begin{array}{r} 11010000 \\ -00011011 \\ \hline 10110101 \end{array}$$

同加法运算类似,减法运算时,若不够减时,可向高位借1。那么每一位被减数和减数及低位的借位进行减法运算后,就可得到本位的差值及向高位的借位。

3. 乘法运算规则

$$0 * 0 = 0$$

$$0 * 1 = 0$$

$$1 * 0 = 0$$

$$1 * 1 = 1$$

乘法运算很简单,除了 $1 \times 1 = 1$ 外,其它情况的乘积均为0。

习 题 一

1. 计算机系统由哪两部分组成? 硬件结构由哪几部分组成? 各部件的功能是怎样的?
2. 计算机的软件有哪些? 操作系统的功能是什么?
3. 个人计算机(PC机)的基本硬件配置是怎样的? 操作系统是什么?
4. 解释下述名词: 指令, 程序, 机器语言, 汇编语言, 高级语言。列举你所知道的高级语言。
5. 下列二进制数相当于十进制的多少? 相当于十六进制的多少?
(1) 11001011 (2) 11010001
(3) 10110111 (4) 10001010
6. 二进制数的运算
(1) $10011001 + 00110101$
(2) $11001101 - 10010111$

第二章 PASCAL 语言基础

第一节 PASCAL 语言特点

PASCAL 语言是瑞士苏黎世联邦工业大学沃思(N. Wirth)教授于 1968 年研究成功的。

PASCAL 语言是一种编译型的结构化系统程序设计语言。它遵循“结构化程序设计”思想,强调“结构化,层次化,模块化”的程序设计原则。PASCAL 提供丰富的数据类型以及用于控制流程的语句,因而程序具有清晰的结构,灵活通用的语句,自由的书写格式,优美的设计风格,紧凑的编译方式,高效的运行结果等特点。

使用 PASCAL 语言可以编写系统软件及各种应用软件。各高等院校的计算机专业将 PASCAL 语言作为首选的程序设计教学语言。

PASCAL 语言从 1975 年的标准化版本开始,不断发展完善,至今已有适用于各种计算机的多种系统。目前广泛应用的有 Turbo Pascal 系统,它提供给用户编辑、编译、运行一体的集成开发环境,以及多种系统功能,包括 Crt、Dos 单元,图形,覆盖技术及汇编接口等。本书以 Turbo Pascal 5.5 系统为软件环境,书中所有例题均已在该系统上调试通过。

第二节 PASCAL 语言程序的组成

一、程序结构

首先我们来看一个简单的程序例子:

```
PROGRAM Example;  
BEGIN      (* main *)  
    write ('Hello!');
```

```
END.
```

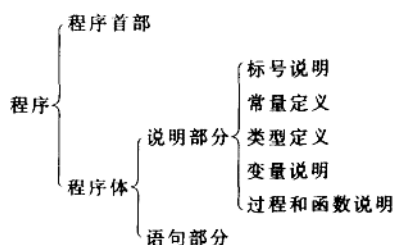
程序以 PROGRAM 为标志开头,Example 是该程序的名字,由 BEGIN 和 END 括起来的是程序的语句部分,表示程序所要执行的操作。该程序的运行结果是在屏幕上显示字符串“Hello!”。

由上述例子可以看出,PASCAL 语言程序由程序首部和程序体组成。程序首部以 PROGRAM 为标志开始,跟上用户定义的程序名,之后加分号“;”。程序名只能由字母和数字组成,并且以字母开头,不允许出现其它符号。

程序体分为说明部分和语句部分。说明部分对程序中用到的所有变量、常量、标号、过程或函数进行定义(或称说明)。语句部分的作用是通知计算机执行指定的操作。如 write 是写语句,它告诉计算机要在屏幕上显示数据或字符串。语句部分以 BEGIN 为起始标志,END 为终止标志,中间是语句序列。句号“.”作为程序结束的标志。

程序中的说明部分是描述数据属性的,语句部分用来描述对数据进行的操作。这两部

分是互相联系的，缺一不可。下面列出 PASCAL 语言程序的基本组成结构。



PASCAL 程序的书写方式比较灵活。允许在一行中写几个语句，也允许一个语句分几行写。在程序的语句间可以插入若干个注释行。注释必须用花括号“{”和“}”，或圆括号加星号“(*”和“*)”括起，括号一定要配对使用，而且不能混淆，注释行对程序运行不起作用。应当使自己所写的程序结构清晰，层次明显，易读性好。

程序中除了用单引号括起的字符串之外，其它字母的大写与小王形式是等价的。即 BEGIN 也可写成 begin 或 Begin。

如果程序有错误，编译器能够检查出语法错误，并显示出错信息，用户修改后可继续运行。如下面程序有错：

```

PROGRAM Welcome;
BEGIN
  Writeln ('Welcome to Pascal Programming!');
END
  
```

编译时得到如下出错信息：

Error 85: “;” expected (缺分号)

将第一行 Welcome 后的 “:” 改为 “;”。

Error 3: unknown identifier (未知标识符)

第三行 writeln 拼写错误，改正。

Error 10: unexpected end of file (不希望的文件尾)

程序应以句点 “.” 结束。在最后一行的 END 之后加 “.”。再次编译可通过。

二、词汇集

PASCAL 语言词汇集是指允许出现在程序中的数字、字母、特殊符号的集合。它由基本字符（包括字母、数字及特殊符号）、关键字与标识符、分隔符组成。

字符是构成词汇的基本元素。目前国际上常用的字符集是 ASCII 码（美国信息交换标准码，American Standard code for Information Interchange）。

ASCII 码有 128 个字符，其中可打印字符 95 个，33 个控制字符。ASCII 码表见附录 D。

可打印字符包括有 26 个英文字母的大、小写形式，数字 0 到 9，及一些特殊符号，如 +、-、*、/、=、<、>，等等。

控制字符不能在屏幕上显示，它们用于执行某种动作，常用的有回车、换行、响铃、文件结束标志等等。

1. 基本字符

字母：A~Z，a~z

数字: 0~9

特殊字符: + - * / = ^ () [] { }.

, ; ' # \$ _

赋值运算符: :=

关系运算符: <=, >=, <>, =, <, >

子界分隔符: ..

注释号: (*, *) 与 { }

2. 关键字 (keyword) 与标识符 (identifier)

关键字是在语言中具有固定含义的单独符号。不允许用户再赋予它们别的含义。如程序首部标记 PROGRAM, 语句部分的 BEGIN、END 等都是关键字。

表 2-1 列出 PASCAL 语言规定的程序中能使用的关键字。

表 2-1

关 键 字 表

AND	ARRAY	BEGIN	CASE	CONST		
DIV	DO	DOWNTO	ELSE	END		
FILE	FORWARD	FOR	FUNCTION	GOTO		
IF	IN	LABEL	MOD	NIL		
NOT	OF	OR	PROCEDURE	PROGRAM		
RECORD	REPEAT	SET	THEN	TYPE		
TO	UNTIL	VAR	WHILE			
标准常量:	FALSE	TRUE	MAXINT			
标准类型:	INTEGER	REAL	BOOLEAN	CHAR	TEXT	
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:	READ	READLN	WRITE	WRITELN	NEW	DISPOS

标识符 (identifier) 定义为由字母 (letter) 开头的字母数字串。

用户可以自定义标识符, 用来表示程序中的变量、常量、类型及过程名等。

标识符在程序中必须遵循“先说明, 后使用”的原则。定义标识符必须符合语法。如下列是合法的标识符:

i, x1, abc, sum

下列则是不合法的标识符:

1A, x+y, R [1], k*

标识符中的字母大小写同义。如 a 与 A 认为是同一标识符。

同样, 关键字的使用也不区分大小写, 如 BEGIN、Begin、begin 认为表示的是同一关键字。

第三节 语 法 描 述

高级程序设计语言的结构规则称为语法。