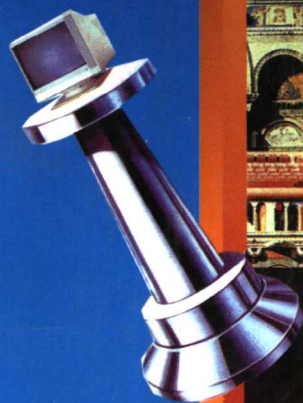


JISUANJI PEIXUN CONGSHU

PASCAL

语言程序设计入门

殷新春 编著
于向军



人民邮电出版社

计算机培训丛书

PASCAL 语言程序设计入门

殷新春 于向军 编著

人民邮电出版社

内 容 简 介

本书介绍了标准 Pascal 语言, 并通过大量精选的例题阐述了利用 Pascal 语言进行程序设计的基本方法与技巧, 最后附有 Turbo Pascal 的使用方法。本书语言通俗, 叙述清晰, 习题丰富, 突出了结构化程序设计、算法设计的一般方法。

计算机培训丛书 PASCAL 语言程序设计入门

殷新春 于向军 编著

责任编辑 王亚明

*

人民邮电出版社出版发行
北京朝阳门内南竹杆胡同 111 号
北京顺义振华印刷厂印刷
新华书店总店科技发行所经销

*

开本: 850×1168 1/32 1995 年 6 月第 一 版
印张: 7.75 1996 年 2 月北京第 2 次印刷
字数: 185 千字 印数: 8 001—16 000 册

ISBN7-115-05681-1/TP·201

定价: 11.00 元

讓更多人學習計算機
讓更多人使用計算機

谷超豪

计算机培训丛书

编 委 会

高级顾问
主 任
副 主 任
委 员

谷超豪 陈国良

牛田佳

李树岭 刘振安

(以姓氏笔画为序)

马鸿初 王德泽 白方舟

孙德敏 林达全 季晓东

徐修存 奚福云 殷新春

黄 德 程慧霞

丛 书 前 言

近年来, 计算机应用已渗透到了社会的各个领域, 计算机的操作已成为人们的一项基本技能。如何正确地使用计算机、充分发挥计算机的作用日益受到人们的重视。

学习计算机, 每个人的起点不同, 但是基本概念、基本操作和常用软件的使用是每个人都必须掌握的, 为了帮助读者打下一个坚实的基础, 人民邮电出版社组织编写、出版了这套《计算机培训丛书》。

培训包括有人指导的培训和自我培训, 因此本套丛书在写作风格上充分照顾到自学和教学的特点。一方面注重易读性, 使读者易于接受、易于记忆、易于理解, 另一方面重实用、重操作, 通过操作计算机和使用常用软件来提高读者的技能。这套丛书包括基本知识、基本操作和常用软件操作三大部分, 首批出版 18 种。主要介绍计算机的选型、升级、维护、计算机打字、计算机语言和新型软件的使用等内容。

目前, 国内微机的主流机型已趋向 386、486, 一些常用软件不断推出新版本。本套丛书充分考虑了这些情况, 在整套图书内容的安排上, 既介绍基础知识, 又介绍新软件, 让读者在掌握基础知识后能较快地掌握新软件的使用。

本套丛书主要面向计算机初学者和各种软件的初级使用者。它可以作为各类计算机培训班的教材, 也可供中学、中专教学使用。对计算机自学人员来说, 它也是一套不可多得的参考书。

随着计算机技术的不断发展, 本套丛书还将不断充实与更新, 欢迎各界朋友对丛书的编写、出版提出宝贵的意见和建议, 让我们共同把《计算机培训丛书》变成各界朋友学用计算机的良师益友, 与您一同遨游奇妙的计算机世界。

前 言

在我国,电子计算机的应用越来越广泛、越来越深入。在近年来的计算机程序设计课程的教学实践中,我们感到现有的以语言为主要内容的教材已不能适应“三个面向”的需要,而必须从数据结构、算法设计的高度来训练读者的程序设计技能。为此,我们编写了本书。

全书共分八章。第一章概述了 Pascal 语言的基本知识;第二、三、四章分别介绍了如何利用 Pascal 语言进行顺序结构的程序设计、选择结构的程序设计和循环结构的程序设计;第五、六章分别介绍了 Pascal 的字符类型、枚举类型、子界类型、数组类型;第七章介绍了子程序结构的程序设计。前面的介绍均以标准 Pascal 为蓝本(介绍了除集合类型、记录类型、文件类型、可调数组以外的全部内容),第八章则简要介绍了 Turbo Pascal 对标准 Pascal 的部分扩充及使用方法。

本书每章都精选了若干例题,配备了适量的习题,希望读者能通过自己解题和上机调试,逐步提高程序设计的技巧,形成良好的程序设计风格。

有一点要指出的是:Pascal 在编译时对 Pascal 标准函数和标准标识符的大小写并不加以区分,本书为了醒目,对所有 Pascal 标准函数和标准标识符采用了首字母大写,在实际编程时,建议读者对它们全部采用小写。

考虑到初学者的知识背景,全书内容力求通俗易懂,循序渐进。

作 者

目 录

第一章 Pascal 语言的基本知识	(1)
1.1 Pascal 语言的发展	(1)
1.2 Pascal 语言的优缺点	(2)
1.3 Pascal 程序的组成	(3)
1.4 语法图	(5)
1.4.1 图形符号	(5)
1.4.2 语法图表示的示例	(6)
1.5 关键字和标识符	(7)
1.5.1 关键字	(7)
1.5.2 标识符	(8)
1.6 常量与变量	(10)
1.6.1 常量	(10)
1.6.2 变量	(14)
1.7 标准函数与算术表达式	(16)
1.7.1 标准函数	(16)
1.7.2 算术表达式	(20)
1.8 数据类型	(23)
1.9 语句分类	(24)
1.10 程序的输入和运行	(25)
1.10.1 编辑	(25)
1.10.2 编译	(25)
1.10.3 运行	(25)
1.10.4 分析	(26)
练习 1	(26)

第二章 顺序结构的程序设计	(31)
2.1 赋值语句	(31)
2.2 输出语句——写语句	(32)
2.3 输入语句——读语句	(36)
2.4 程序举例	(38)
练习 2	(41)
第三章 选择结构的程序设计	(45)
3.1 布尔表达式	(45)
3.1.1 布尔常量与布尔变量	(45)
3.1.2 关系表达式	(46)
3.1.3 布尔运算及布尔表达式	(47)
3.1.4 布尔型数据的输入和输出	(48)
3.2 IF 语句	(49)
3.2.1 IF 语句的概念	(49)
3.2.2 复合语句	(52)
3.2.3 IF 语句的嵌套	(55)
3.3 CASE 语句	(61)
3.4 程序举例	(63)
练习 3	(69)
第四章 循环结构的程序设计	(73)
4.1 FOR 循环	(73)
4.1.1 FOR 语句	(73)
4.1.2 FOR 循环的嵌套	(81)
4.2 WHILE 循环	(85)
4.3 REPEAT-UNTIL 循环	(91)
4.4 枚举法	(97)
练习 4	(101)
第五章 其它简单数据类型	(105)

5.1	类型概念的进一步说明	(105)
5.2	字符类型	(106)
5.2.1	字符常量	(106)
5.2.2	字符符号常量	(109)
5.2.3	字符变量	(109)
5.2.4	字符量的运算	(110)
5.2.5	字符数据的输入和输出	(111)
5.2.6	字符处理程序举例	(112)
5.3	枚举类型	(117)
5.3.1	为什么要使用枚举类型	(117)
5.3.2	枚举类型的定义	(118)
5.3.3	枚举类型的性质	(119)
5.3.4	枚举类型应用举例	(121)
5.4	子界类型	(125)
5.4.1	问题的提出	(125)
5.4.2	子界类型的定义	(126)
5.4.3	子界类型数据的运算规则	(127)
5.4.4	子界类型应用举例	(128)
练习 5		(129)
第六章	数组及其应用	(133)
6.1	数组的定义	(133)
6.1.1	一维数组	(133)
6.1.2	多维数组	(143)
6.2	字符数组及字符串	(150)
6.2.1	字符数组	(150)
6.2.2	字符串常量	(151)
6.2.3	字符串变量	(151)
6.3	数组操作示例	(155)

6.4 高精度运算	(161)
练习 6	(164)
第七章 子程序结构的程序设计	(168)
7.1 函数	(168)
7.1.1 函数的定义	(169)
7.1.2 函数的调用	(170)
7.2 过程	(174)
7.2.1 过程的定义	(175)
7.2.2 过程的调用	(176)
7.3 全程量和局部量	(178)
7.3.1 局部量及其作用域	(178)
7.3.2 全程量及其作用域	(180)
7.4 形参和实参的结合	(183)
7.4.1 值形参	(183)
7.4.2 变量形参	(183)
7.5 子程序结构程序设计示例	(186)
7.6 查找与排序	(192)
7.6.1 查找	(192)
7.6.2 排序	(195)
练习 7	(207)
第八章 Turbo Pascal 使用说明	(214)
8.1 Turbo Pascal 入门	(214)
8.1.1 安装	(214)
8.1.2 Turbo Pascal 集成环境	(215)
8.1.3 Turbo Pascal 编辑器	(217)
8.1.4 Pascal 程序的编译和运行	(222)
8.2 对标准 Pascal 的扩充	(223)
8.2.1 数据类型	(223)

8.2.2	运算符	(228)
附录 I	EBCDIC 码表	(230)
附录 II	Pascal 关键字	(231)
附录 III	Pascal 标准标识符	(232)
附录 IV	Pascal 标准函数表	(233)
附录 V	Pascal 运算符一览表	(234)

第一章 Pascal 语言的基本知识

1.1 Pascal 语言的发展

Pascal 语言是以曾发明第一台机械计算机（史称“加法机”）的法国数学家 Blaise Pascal 的名字命名的。表 1.1 示意了 Pascal 语言的产生和发展。

表 1.1 Pascal 语言的发展

时 间	内 容
1968 年	拟定了 Pascal 语言初稿
1969 年	N·沃思教授设计成功 Pascal 语言
1970 年	CDC-6000 机上首次实现了 Pascal 语言的编译程序
1971 年	发表了 Pascal 语言用户手册，它作为 Pascal 语言诞生的标志
1974 年	发表了 Pascal 语言的修订报告，即 N·沃思提出的“标准 Pascal”
1980 年	ISO* 发表了 ISO (DP/7185) 关于 Pascal 的建议草案 (即是 1979 年英国标准化协会提交的 Pascal 英国标准)
1983 年	ISO 正式发表了 Pascal 语言国际标准 (ISO 7185-1983)
1987 年	我国批准 Pascal 语言国家标准 (GB 7591-87)

* ISO: International Standard Organization, 国际标准化组织

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, PDP-11 Pascal, OMSI 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。这两个标准大体上相同, 但有两处做了较大的改动:

①将 1974 年的标准 Pascal 关于过程参数和函数参数的形式说明改为使用过程首部或函数首部;

②ISO 的标准 Pascal 引入了可调数组, 但尚未有实际系统实现之。

本书介绍的 Pascal 语言是这两种 Pascal 标准的“交集”(未介绍可调数组), 基本不涉及具体的 Pascal 版本, 只在供指导实验的第八章才介绍了 Turbo Pascal 的使用。

1.2 Pascal 语言的优缺点

N·沃思教授在他的研制报告中指出研制 Pascal 的两个目的:

第一, 该语言能清晰地展示程序设计中算法和数据的自然结构, 因而使程序设计教学成为逻辑的系统训练。

第二, 开发这种语言的实现方法, 使这种实现在现有的各种计算机上既可靠又有效。

就 Pascal 语言的实践情况来看, 已经达到并且远远超过了 N·沃思原先的设计目标。

从使用者的角度来看, Pascal 语言有以下几个主要的特点:

①它是结构化的语言。Pascal 语言提供了直接实现三种基本结构的语句以及定义子程序的功能, 可以方便地书写结构化的程序。

②有丰富的数据类型。Pascal 提供了整型、实型、字符型、枚举型、子界型以及由以上类型数据构成的数组类型、集合类

型、记录类型和文件类型。此外，还提供了其它许多语言中所没有的指针类型。

③能适用于数值运算和非数值运算领域。

④程序的书写格式比较自由。

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。

当然，Pascal 语言并非十全十美，用于教学是比较理想的，但用于实际还美中不足：文件处理功能较差，不能使用随机文件；没有提供可调数组的功能；缺乏与机器语言的接口等等。这些妨碍了 Pascal 语言在实际中的应用，沃思于 1980 年发表的 MODULA-2 语言就是 Pascal 语言的发展，弥补了 Pascal 语言的不足。另外，各种具体的 Pascal 版本不同程度地对标准 Pascal 做了扩充，如字符串处理、图形、声音、色彩等，这样更加有利于 Pascal 语言的实际应用。

1.3 Pascal 程序的组成

我们看看下列利用辗转相除法求最大公约数的程序：

```
program gcd(Input, Output);  
  var r, n, m: Integer;  
  begin  
    Readln(m, n); {input data}  
    Write('(' , m, ', ', n, ')=');  
    while n > 0 do  
      begin  
        r := m mod n;
```

```

        m:=n; n:=r
    end;
    Writeln(m)
end.

```

从这个程序可以看到:

(1) 一个 Pascal 程序分为两个部分: 程序首部和程序体。

(2) 程序首部是程序的开头部分。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。由程序设计者自己定义, 如例中的 gcd。

③ 程序参数。写在程序名称后面括号内, 如例中的“Input”、“Output”, 程序参数表示程序与外界的联系, “Input”是系统预定义的输入文件, 一般的系统隐含指定 Input 的输入设备为键盘。“Output”是系统预定义的输出文件, 一般输出设备为显示器。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。

(3) 程序体是程序的主体, 有时也称为“分程序”。程序体包括说明和执行两个部分。

① 说明部分用来描述程序中用到的数据的属性。本程序中第二行是“变量说明区”, 用来定义变量的名称、类型。

Pascal 规定, 凡程序中用到的所有变量、符号常量、数组、过程与函数等数据都必须在说明部分进行定义 (或称“说明”), 也就是说, 不允许使用未经定义的变量。

② 执行部分的作用是通知计算机执行指定的操作。如果一个程序中不写执行部分, 在程序运行时计算机什么工作也不做。因此, 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。

(4) Pascal 程序的书写方法比较灵活。当然，书写不应以节省篇幅为目的，而应以程序结构清晰、易读为目的。请读者自己编写程序时应尽量模仿本书中例题程序格式。

(5) 在程序中，一对大括号间的文字称为注释。注释的内容由人们根据需要书写，可以用英语或汉语表示。注释可以放在任何空格可以出现的位置。

1.4 语 法 图

每一种计算机语言都有自己的语法规则，语法规则可以用自然语言来描述，但往往叙述冗长而不严格，例如前面介绍的 Pascal 程序的组成。常用语法图来表示语法规则。

1.4.1 图形符号

语法图中常用的符号如图 1.1 所示。

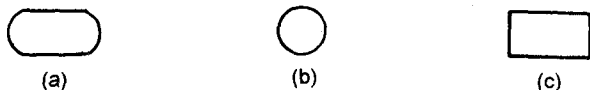


图 1.1 语法图图形符号

其中：(a) 表示 Pascal 的关键字，或无需进一步定义的语法实体（如字母或数字）；(b) 表示 Pascal 的操作符或规定的其它特殊符号；(c) 表示由其它语法图定义的语法实体。

常常可以将 (a) 与 (b) 混用，根据框内的内容多少而画成圆边框或圆型框。