



PASCAL 程序设计

张德荣 编



陕西师范大学出版社

814343

73.57.824
29-C90

PASCAL 程序设计

阅览 8 清

张德荣 编

陕西师范大学出版社



10359031



10312

内 容 提 要

本书介绍标准 PASCAL 语言的用法和编写结构化程序的基本方法。编写了一些常见的应用程序。严格按照 PASCAL 标准讲解，所编程序可用于多种类型的机器。全书由浅入深，循序渐近，程序结构清晰，层次分明，体现了结构程序的优点。有选学内容，能适应不同读者的需要。可作为高等学校教材，也可供科技工作者参考。

PASAL 程序设计

张德荣 编

*

陕西师范大学出版社出版

(西安市陕西师大120信箱)

陕西省新华书店经销

西安电子科技大学印刷厂印刷

*

开本787×1092 1/32 印张6.25 插页2 字数126千字

1988年6月第1版 1988年6月第1次印刷

印数：1—3000册

ISBN7-5613-0073-5

G·85 定价：1.45元

说 明

本书是为使用 PASCAL 语言编写程序的初学者编写的。介绍标准 PASCAL 语言的内容，叙述编写程序的基本方法。在我校多次使用，既曾作为第一种算法语言课的教材，也曾作为第二种算法语言课的教材。PASCAL 语言本身，小巧清晰，用途广泛，适宜作为推广计算机应用的启蒙语言，易于受到初学者的欢迎。使用本教材的实践过程，也充分证实了这一点。

书中所选例题，一般比较短小，许多例子中的程序，也有它自己的特点。全部程序，都在 MV-6000 计算机上严格地使用标准 PASCAL 编译通过，还在 IBM-PC 机编译通过。备有 IBM-PC 计算机用的软盘一张，包括书中全部源程序，部分习题答案，还有编辑和编译程序，使用方便，欢迎复制(免费)。

本书把 GOTO 语句放在最后介绍，以利于养成编制结构程序的良好习惯。为了适应不同情况的需要，用“举例说明”和“应用举例”等标题对书中程序加以区别。前者阐明语言用法，内容较易，后者结合应用介绍编写程序的基本方法，内容较深；既讲算法，又讲程序，难点往往在于算法。请读者根据需要决定取舍。对个别较难的内容，还用星号“*”标明，初学者可以不学。

最后，对试用过本书的冯德民，唐家益，王四万等同志表示诚挚的谢意。

编 者

一九八七年二月

于陕西师范大学

目 录

第一章 引言	1
§ 1 从用户观点看计算机	1
§ 2 算法语言	3
§ 3 字符集, 语法范式和语法图	4
§ 4 程序结构	9
§ 5 举例说明	10
第二章 标准类型和表达式	12
§ 1 标准类型和标准函数	12
§ 2 表达式	18
第三章 常用语句	23
§ 1 输入和输出	23
§ 2 空语句	27
§ 3 赋值语句	28
§ 4 复合语句	30
§ 5 如果语句	30
§ 6 重复语句	41
§ 7 应用举例	52
第四章 标量类型和子界类型	57
§ 1 标量类型	57
§ 2 子界类型	60
§ 3 分情形语句	63
§ 4 应用举例	64

第五章 集合类型	69
§ 1 集合类型.....	69
§ 2 应用举例.....	73
第六章 数组类型	77
§ 1 一维数组.....	77
§ 2 多维数组.....	84
* § 3 紧缩数组.....	89
§ 4 布尔数组.....	91
第七章 过程与函数	94
§ 1 过程.....	94
§ 2 函数.....	105
§ 3 递归.....	108
* § 4 应用举例.....	111
§ 5 数组排序.....	119
第八章 记录	133
§ 1 不带变体的记录.....	133
§ 2 带变体的记录.....	137
§ 3 举例说明.....	141
第九章 文件	144
§ 1 文件的基本概念.....	144
§ 2 文件成分的读写.....	146
§ 3 行文文件.....	149
* § 4 应用举例.....	152
第十章 指针类型	156
§ 1 指针类型.....	156
§ 2 链表.....	159

* § 3 树结构	166
第十一章 转向语句	172
§ 1 标号和转向语句	172
习题	174
附录 A 字汇表	185
附录 B 语法范式	187
附录 C 程序索引	193
参考书	194

第一章 引 言

本书介绍算法语言 PASCAL，同时介绍程序设计的基本方法，以便编写常见问题的 PASCAL 语言程序。

本章先介绍一些基本概念。

§ 1 从用户观点看计算机

电子计算机由运算器，控制器，存储器和输入输出设备等部件组成。前三者是计算机的主体，构成所谓主机。运算器与控制器是计算机的核心部分，称为中央处理机，简称 CPU (Central processing Unit 的缩写)。

各组成部分的主要功能如下：

运算器 能完成加减乘除等算术运算，简单的逻辑运算和其他一些基本操作。

存储器 存储各类信息，如原始数据，中间结果以及指令等等。

控制器 根据设计制造计算机时安排好的方法和步骤，按照使用时所进行的操作，指挥各部件自动地，协调地完成规定的任务，如让输入输出设备传递信息，或按已存储的指令内容指挥各部件执行指令所包含的各项运算或操作。控制器是计算机的指挥中心。

输入输出设备 负责存储器和输入输出设备之间信息

的传递。

上述计算机的组成部件，包括机械的，磁性的，以及各类电子的，统称机器系统，或硬件。

计算机不断发展，应用日益广泛；与此同时，对计算机的要求也不断增多，不断提高。为了适应各类用户的需求，符合人们多年形成的习惯，既使用户满意，又要充分发挥机器的效能：各种各类庞大的系统程序，应运而生。这些系统程序，总称为软件。硬件和软件构成了整个计算机系统。

时至今日，一般用户不直接使用硬件本身，仅仅通过各种系统程序使用计算机。只要了解有关系统程序的用法及其功能，就能有效地使用计算机。

上面所说的存储器能直接在控制器的控制下进行工作，特称为内存储器，简称内存。由于工业和经济方面的原因，内存容量较小，不能满足大型问题的需要，因而用容量更大的磁盘和磁带作为辅助存储器，简称外存。外存和内存的信息可以互相传递。现在，小型计算机，甚至微型计算机都配有容量相当大的磁盘。借助于软件系统的使用，用户可以像使用内存一样使用某些外存。

同时，软件还使一台计算机能供多个用户同时使用，并让每个用户感到他独自一人操纵着计算机。

因此，不必对硬件的构造，不必对软件的具体内容，有更多的了解，只要熟悉他们的用法，就可以使用计算机系统。以后，我们把计算机系统简称为计算机。在使用计算机的各个阶段，着重了解每项操作所能达到的功能，就可以在一定范围内，卓有成效地使用计算机。

必须注意，使用计算机，有极为严格的规定，任何一个字母，数字或符号，都不允许有丝毫错误，只要还存在任何错误，都必然达不到预期的结果，甚至出现难以设想的现象。使用计算机，对于培养周密而细致的思维，是很好地锻炼。

§2 算法语言

计算机硬件只能接受机器语言，即机器指令，只会执行由机器指令写成的程序。所谓机器指令程序，就是为了处理数据，精心设计而编出的机器指令序列。机器指令十分繁琐，严格依赖于机器。为了方便用户，为了让同一程序可以为不同计算机所执行，人们设计出算法语言。计算机系统有一种叫做编译程序的软件，把按照算法语言规定所写成的程序(称为源程序)翻译为机器指令程序(称为目标程序)，以便硬件执行。还有一种程序，它不把整个源程序直接翻译为完整的目标程序；它在阅读了源程序以后，根据源程序要求和系统特点，整理出简要的资料，据此和源程序文本以及原始数据和中间结果的具体情况，让计算机逐步完成源程序所列出的算法。此种程序称为解释程序。解释程序和编译程序在用户和计算机之间，起桥梁作用。

算法语言的规定十分严格。设计算法语言的时候，既要考虑易写易读，使用方便，又要发挥机器效能，还不能不照顾是否便于编译且能及时发现错误。千头万绪，因素甚多。只能从全局出发，取大利而弃小害，作出各方基本满意的裁决。这样看来，算法语言又是用户与系统之间的协议书。

在学习和使用算法语言的时候，对算法语言的作用及其设计思想均应予以充分注意。

算法语言，种类繁多。常见的有BASIC(是Beginner's All-Purpose Symbolic Instruction一词的缩写)，FORTRAN(FORMula TRANslation一词的缩写)，ALGOL(ALGORithmic Language的缩写)，PASCAL(以法国数学家PASCAL命名)等等。PASCAL语言的用户手册和修订报告公布于1974年。是一种用途广泛的高级语言，它吸取了ALGOL语言的特点，还提供了构造数据类型的方法，既便于进行数值计算，又便于描述非数值性问题，包括书写编译程序，还宜用以学习程序设计技巧，作为一种教学语言。加之语言本身小巧清晰，许多微型计算机都配有PASCAL编译程序。在我国也逐渐被更多的人所应用。有人建议把它作为大学推广计算机应用的启蒙语言。还有人认为，它已有十余年的历史，目前形式的PASCAL有可能不会再用很多年，但它的概念还会在多年内仍为设计算法语言的人继续使用。

这份讲义，介绍PASCAL语言的基本内容，讲解如何用它编写解决常见问题的源程序。源程序和编译程序都符合语言的所有规则，所编程序才能为计算机所接受，完成编译。即使编译成功，也只有当算法正确，没有丝毫错误，才能解决实际问题。

§3 字符集，语法范式和语法图

1 字符集

算法语言都有自己规定的字符集合。PASCAL语言

使用大小写英文字母，共 52 个；使用由 0 至 9 共十个数字，还有标点符号等，详见附录 A。由字母所组成的 35 个保留字有明确而固定的含义，不能变更语法的规定而应用。标点符号也是如此，只不过具体机器规定有代用符号。方括号，圆括号，花括号，各有各的用途，应严格按照规定使用。

标识符是用户用以表示常数，类型，变量，过程，函数以及文件等的名字。规定标识符由一个字母开头，其后附以 n 个(允许 $n=0$)字母或数字。标识符由用户根据他自己的习惯以便于识别而任意选用。但大部分计算机只取标识符中前八个字符进行识别。附录 A 中列出了一些标准标识符，它们也各有各的含义。以后陆续说明。

大多数计算机，对同一英文字母的大写或小写不加区别，这种机器把 `Abc`，`aBc`，`abC` 视为同一标识符，把 `a2` 和 `A 2` 也视为同一标识符。

合法的标识符如：

```
Pi f10T0 2 x sum root3 readin
```

非法的标识符如：

```
1st array teuel.4 root-3
```

因为数字不能作为标识符的第一个字符，`array` 是保留字，字符减号和句号不能作为标识符中的字符。要想把含有空格的字符序列 `F T` 当作一个标识符而使用，也是非法的。

本书用小写黑体字母写出保留字，而用小写字母书写标准标识符和用户自己定义的标识符，以突出保留字而利于阅读源程序。

2 语法图

用“字母”表示英文字母中的某一个，用“数字”表

示 0 至 9 中的某一个，这些规定和上一小节所定义的标识符均可用图 1-1 表示。

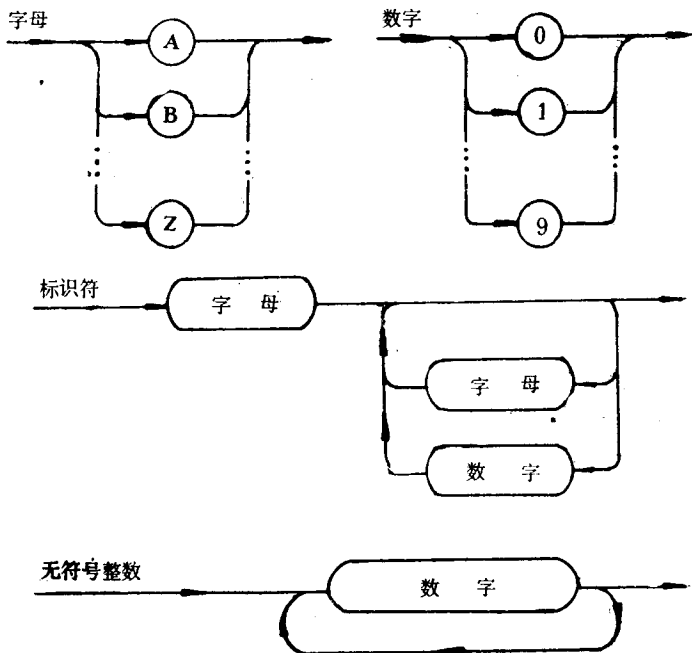


图 1-1 语法图举例

语法图的规则既直观又清晰：沿箭头顺序前进，遇岔道择一而行，直至终点为止。任何一种走法，都构成合法结果。

3 巴科斯(BACKUS)范式

巴科斯范式简记为BNF(BACKUS NORMAL

FORM或 BACKUS NAUR FORM),它不是PASCAL语言中的符号,仅用它来描述 PASCAL 语法规则。举例说明如下。

$\langle \text{字母} \rangle ::= A | B | \dots | Z | a | \dots | z$

$\langle \text{数字} \rangle ::= 0 | 1 | \dots | 8 | 9$

$\langle \text{标识符} \rangle ::= \langle \text{字母} \rangle \{ \langle \text{字母} \rangle | \langle \text{数字} \rangle \}$

$\langle \text{无符号整数} \rangle ::= \langle \text{数字} \rangle \{ \langle \text{数字} \rangle \}$

以上各式中的 ::= 表示“定义为”,竖线|表示“或”。花括号表示由它所括起的内容可以出现几次,也可能不出现。第一式中的 $\langle \text{字母} \rangle$ 在符号 ::= 的左端,表示此式就是“字母”的定义;第三式中 $\langle \text{字母} \rangle$ 出现在符号 ::= 的右端,表示“字母”的含义应由定义“字母”的范式而确定。

和上一小节的语法图相对照,可以看出,它们以不同方式描述语法,异曲同工。建议读者了解其中一种即可。希望能懂得语法范式的含义,能根据语法范式判断所写程序是否符合语法。

无符号数的语法范式如下。

$\langle \text{无符号数} \rangle ::= \langle \text{无符号整数} \rangle | \langle \text{无符号实数} \rangle$

$\langle \text{无符号实数} \rangle ::= \langle \text{无符号整数} \rangle . \langle \text{无符号整数} \rangle$

$| \langle \text{无符号整数} \rangle E \langle \text{标度因子} \rangle$

$| \langle \text{无符号整数} \rangle . \langle \text{无符号整数} \rangle E \langle \text{标度因子} \rangle$

$\langle \text{标度因子} \rangle ::= \langle \text{无符号整数} \rangle | \langle \text{正负号} \rangle \langle \text{无符号整数} \rangle$

$\langle \text{正负号} \rangle ::= + | -$

由此可见,无符号整数是一个或多个数字组成的序列,至少有一个数字,如108000是无符号整数,但108,000却不是无符号整数,因为无符号整数由数字序列组成,不允许出现十个数字以外的任何字符。

至于无符号实数,有且仅有范式中所列出的三种形式.

例如

3.1416 0.31416 E 1 0.31416 E + 1
3.0 0.3 27.1828 E - 1

都是无符号实数,其中的 E 表示“乘以 10 的…幂”.还可看出,在有小数点的无符号实数中,小数点的前后都必须是无符号实数,所以

3. .3 E 3 0.314,16

都不是合法的数. 3.0, 1 E 2 虽然都是合法的数,但计算机把它们当作实数,不把它们当作整数.

使用计算机时,源程序以及输入和输出的数,都是十进制数.

4 常量

常量有五种,其范式如下.

〈字符串〉 ::= '〈字符〉{〈字符〉}'

〈常量〉 ::= 〈无符号数〉 | 〈正负号〉〈无符号数〉

 | 〈正负号〉〈常量标识符〉 | 〈常量标识符〉

 | 〈字符串〉

字符指可以打印的任何符号(对大小写字母加以区别),字符串指用一对单引号括起的字符序列,如果要把单引号本身作为字符序列中的一个字符,可以在这一对单引号内应当写此单引号的地方,写两个单引号.特别注意,四个单引号''''表示由一个单引号组成的字符串.

下列都是合法的常量

3 3.1416 +702 -pi 'Entre a number'

'The area of it''s Surace'

下列都是非法的常量

π .5 E10 .Entre a number.

§ 4 程 序 结 构

1 程序结构概况

PASCAL 程序由三部分组成。

1° 程序首部 用保留字 **program** 开头, 标志程序的起始, 其后列出程序名以及有关参数。

2° 分程序 分程序又分为说明部分和语句部分两部分。

3° 程序结束标志 语句部分一定以保留字 **end** 结尾, 此 **end** 后的句号, 标志程序结束。

2 分程序中的说明部分

分程序中的说明有: 标号说明, 常量定义, 类型定义, 变量说明, 过程与函数说明五部分。这五部分中的某些或全部都可以不出现(附录 B 各范式中的〈空〉就表示什么也没有), 如果出现, 必须按照上列顺序排列。标号说明, 过程与函数说明两部分, 分别在第十一章, 第七章介绍。

常量定义部分由保留字 **const** 带头, 以分号结尾, 中间有一个或几个常量定义; 若常量定义多于一个, 则两个常量定义之间, 用分号作为分隔符; 常量定义的形式为

〈标识符〉=〈常量〉

类型定义部分由保留字 **type** 带头, 以分号结尾, 中间

有一个或几个类型定义；若类型定义多于一个，则两个类型定义之间，用分号作为分隔符；类型定义的形式为

〈标识符〉=〈类型〉

变量说明部分由保留字 **var** 带头，以分号结尾，中间有一个或几个变量说明；若变量说明多于一个，则两个变量说明之间，用分号作为分隔符；变量说明的形式为

〈标识符〉{, 〈标识符〉}:〈类型〉

在程序中，用标识符表示变量，类型等等。也可用标识符来表示常量。除了标准常数，标准类型等标准标识符不必说明就可使用以外，对其它一切标识符都要在说明部分按上述规则，进行说明或定义。

3 分程序中的语句部分

语句部分由保留字 **begin** 开头，以保留字 **end** 结尾；中间有一个或多个语句；若有多个语句，则每两个语句之间，用分号作为分隔符。

分程序可以没有说明部分，但必须有语句部分。

编写程序，一般先确定算法；再列出实现算法的语句，同时统计所用过的标识符，根据规定，列出相应的说明。在源程序中，说明部分一定写在语句部分之前。

§5 举例说明

本节举例说明程序结构。

例1 编一程序，显示计算机系统的最大整数。

解 程序如下。