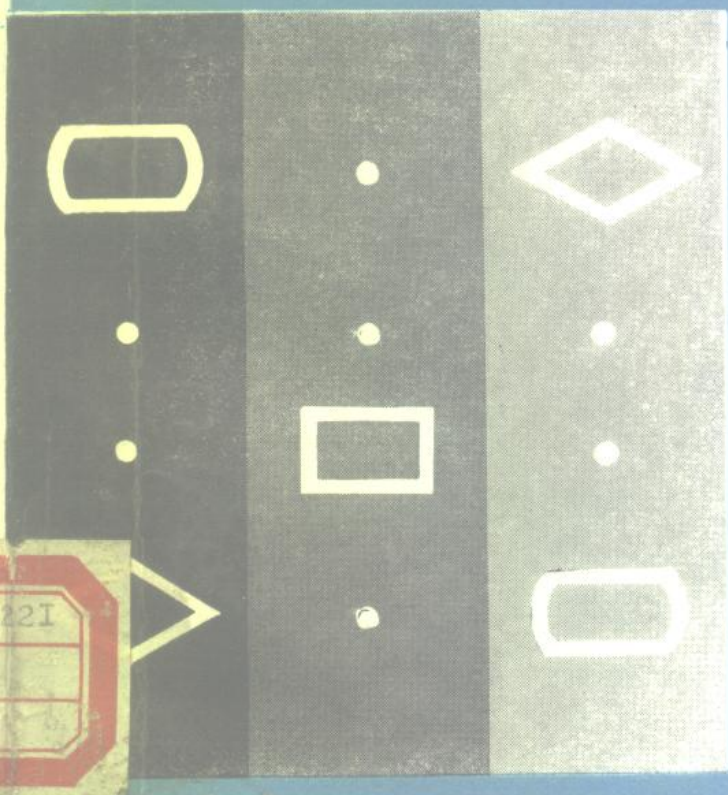


计算机基础自学丛书

高级语言程序设计

王 诚 苏云清 编著



机械工业出版社

计算机基础自学丛书

计算机基础自学丛书

高级语言程序设计

王 诚 苏云清 编著



机械工业出版社

ISBN 7-111-01168-1/S
内 容 简 介

本书是《计算机基础自学丛书》第二册，由清华大学计算机科学与技术系的老师在多年教学和科研工作的实践基础上编写的。

高级语言程序设计是当代科技、教育工作者的必修课。而PASCAL语言和C语言是当前比较优秀的高级语言，PASCAL语言正迅速推广，而C语言则前程似锦。

本书共分六章。各用两章介绍PASCAL语言和C语言的全部数据类型、多种通用语句，程序设计的基本方法和实际应用，并提供各种类型的程序70多个。最后用两章综合叙述程序的运行和程序设计的技巧。

全书层次清楚，通俗易懂，深入浅出，内容丰富，是一本较好的掌握高级语言程序设计的自学教材或培训教材。

高级语言程序设计

王 诚 苏云清 编著

责任编辑：邱锦来

封面设计：田淑文

机械工业出版社出版（北京阜成门内大街25号）

（北京市书刊出版业营业许可证出字第117号）

中国农业机械出版社印刷厂印刷

新华书店北京发行所发行·新华书店经售

开本 850×1168 1/32·印张 8 3/4·字数 216 千字

1987年9月北京第一版·1987年9月北京第一次印刷

印数 0,001—6,200·定价：2.70 元

统一书号：15033·6904

出版说明

计算机的广泛应用极大地提高了人类认识和改造客观世界的能力，特别是微电子及大规模集成电路技术的发展，微型计算机的广泛应用为各国工业技术的改造提供了重要而有效的途径。一场以解放人类智力为标志的新技术革命开始了。其规模和深度都将是空前的，不仅涉及到科学，工业，经济，教育等各个领域，同时也将会引起社会结构和生活方式的改变。

预计九十年代将出现以知识处理为基础的智能技术与智能系统，并得到广泛的应用。例如，智能机器人，智能信息处理系统，智能计算机辅助设计系统，智能办公室自动化系统等。它将开创新的计算机应用领域。传统的数据处理，计算机控制等应用更加向前推进。

为了适应新技术革命的需要，帮助各个领域的科技人员进一步普及和提高计算机基础科学知识，使计算机得到广泛的更有效的应用。我们特为全国机电专业及计算机应用专业中专以上文化水平的读者编写了这一套丛书。对广大读者只要具有基本的数字电路及逻辑设计基础知识，均可自学这套丛书。“丛书”着眼于计算机的应用，搜集归纳了国内外大量的资料，并在长期的教学和科研工作基础上进行编写的。丛书力求做到选材上的先进性、系统性和完整性。由浅入深全面系统地阐述了计算机科学的基础知识，同时对计算机的主要应用领域的先进技术也作了必要的介绍。

“丛书”编写者是清华大学计算机科学与技术系长期在第一线从事教学、科研工作的副教授、高级工程师、讲师。丛书由清华大学周远清主编，叶乃华同志为全套丛书的编写、定稿和出版作了大量的工作。

这套丛书共分九册。

第一册计算机概论。阐述计算机的体系结构及各部分工作原理。

第二册高级语言程序设计。介绍PASCAL语言及C语言的基本结构以及语言的使用。

第三册PDP-11机器语言与汇编语言程序设计。介绍PDP-11机的基本组成及编程技术。

第四册数据结构。阐述非数值程序中有关结构以及排序和查找等问题。

第五册操作系统。阐述操作系统的有关基本概念以及管理系统资源，提供用户应用环境。

第六册汉字微型计算机与汉字识别。介绍汉字微型计算机系统的组成、输入编码方案、汉字处理和识别技术以及汉字微型计算机系统的应用。

第七册计算机控制系统及应用。主要分析计算机控制系统的原理、设计方法（连续、离散、状态空间），通道接口及应用实例。

第八册计算机辅助图形设计。介绍计算机辅助设计基础——二维三维图形设计的基本算法以及交互式图形软件的设计。

第九册办公室自动化基础。阐述数据库的基本概念、办公室自动化基础知识及技术。

在编写这套丛书时，我们既注意到它的系统性和完整性，同时又使各册之间有相对的独立性。便于读者能够根据自己从事工作的领域以及自己的兴趣，阅读全套丛书或选读其中几册。

对于需要使用语言进行程序设计的初学者，可以首先阅读第一、二、三册。在有关领域中从事计算机应用的科技人员，可以分别选读第六、七、八、九册。如果需要系统深入了解和掌握计算机科学知识的读者，建议进一步阅读第四、五册。

书中所选实例经过上机验证，有的还附有习题，希望读者能完成练习，并上机进行实践。

这套丛书既可作广大读者的普及读物和科技参考书，又可作为大学教科书。我们希望，这套丛书能为我国计算机的普及与应用起到一定的作用。

由于编者水平有限，书中定有不当之处和错误的地方，欢迎读者批评指正。

清华大学计算机科学与技术系
计算机基础自学丛书编写组

前 言

计算机的迅速发展及其广泛应用，推动了计算机语言的程序设计。广大科技工作者迫切需要学习程序设计，特别是学习高级语言程序设计，让计算机为科研生产服务。满足他们的需要就是本书的主要目的。

目前国际上流行的高级语言种类很多，各有长短。实践表明，对于大多数科技工作者来说，掌握一至二种高级语言就能适应工作的需要。多年来的教学和科研实践使我们认识到，在当前，重点掌握PASCAL语言和C语言是较好的办法。这就是本书选择典型语言的出发点。

PASCAL语言具有丰富完备的数据类型，简明灵活的通用语句，清晰明了的模块结构以及编译紧凑方便，书写格式自由，运行效率高和程序设计风度优美等特点。教学实践还表明，由于PASCAL语言的语法严格，经过PASCAL语言学习的读者编写的程序质量比较高，错误比较少。

C语言作为UNIX系统的主力语言，也可以移植到其它系统上使用。它是以PASCAL语言相类似的思想为基础而面向结构程序设计的语言。它以英文小写字母为基础，由一系列按值调用的函数组成，模块结构明显；它具有丰富而灵活的运算符，编写程序方便；它通过预处理进行宏调用，程序表示简洁；它将指针作为数据处理，程序功能增强，特别适合于系统程序的设计。

本书首先介绍计算机程序设计的基础知识，然后根据语言发展情况及国内的实际情况由浅入深地分别讨论了PASCAL语言和C语言的程序设计及应用。最后综合叙述了如何编辑、运行和调试一个高级语言源程序以及程序设计的技巧问题。

在写作方法上，本书将PASCAL语言和C语言都拆成两章，

前一章叙述基本的内容，如过程和函数；后一章讨论一些更重要的问题，主要是该语言的特点和优点，如各种构造型数据类型。在两种语言的关系上，同样各有侧重。

参加本书编写工作的是清华大学计算机科学与技术系的教师王诚和苏云清。本书的前两章由王诚编写，后四章由苏云清编写。杨德元和史美林分别认真审阅了本书中PASCAL语言和C语言部分，最后由周远清审阅全书内容，他们都提出了宝贵的意见。还有其他许多同志对本书的编写做了各种有益的工作。在此，我们一并表示衷心的感谢！

由于我们水平有限，本书一定有不少缺点和错误，敬请广大读者批评指正。

编者

1986年于清华大学

目 录

绪 论	1
第一章 PASCAL 语言的语句和程序结构	3
§ 1-1 字汇表和程序结构	5
§ 1-2 标准数据类型和标准函数	8
§ 1-3 顺序结构程序	15
§ 1-4 分支结构程序	23
§ 1-5 循环结构程序	31
§ 1-6 模块和层次结构程序——过程和函数	40
第二章 PASCAL 语言的非标准数据类型和应用举例	68
§ 2-1 枚举类型和子界类型	68
§ 2-2 集合类型	73
§ 2-3 数组类型	78
§ 2-4 记录类型	86
§ 2-5 文件类型	93
§ 2-6 指针类型	110
§ 2-7 应用举例	125
第三章 C 语言的基本结构和程序设计	142
§ 3-1 UNIX 和 C 语言	142
§ 3-2 基本结构和符号	143
§ 3-3 简单的程序设计	146
§ 3-4 流程的控制	156
§ 3-5 函数	165
第四章 C 语言的数据类型和应用举例	178
§ 4-1 数组类型	178
§ 4-2 指针类型	180
§ 4-3 结构类型	184

§ 4-4	联合类型	187
§ 4-5	变量初始化	189
§ 4-6	UNIX系统编程	192
§ 4-7	预处理和重新定向	201
§ 4-8	应用举例	212
第五章	程序的运行	227
§ 5-1	常用键盘命令	227
§ 5-2	文本编辑	229
§ 5-3	编译和汇编	231
§ 5-4	链接和执行	232
§ 5-5	程序的调试	234
第六章	程序设计的技巧	235
§ 6-1	计算机语言的选择	235
§ 6-2	优质程序的编制	238
§ 6-3	编译错误的查找	239
§ 6-4	实践中的再检验	240
附 录		241
附录一	PASCAL语言语法图	241
附录二	保留关键字	245
附录三	标准标识符	245
附录四	ASCII字符集	246
附录五	C语言语法图	247
参考文献		257

结 论

语言是交换信息的工具。人与人之间用各种民族的语言来交流思想，交换信息。人与计算机之间也必须交换信息——人向计算机发出命令，计算机向人提供信息，这就要通过计算机语言。

计算机语言随着计算机事业的发展而发展，随着计算机应用的推广而推广。最初使用的是机器语言，它完全依赖于具体的计算机，它要求用户非常熟悉计算机的硬件系统，编写程序由专家承担。这种苛刻的要求必然阻碍计算机的发展和应用。

后来人们发明了汇编语言，用符号来表示程序语言，而不是直接用机器代码来表示程序语言。虽然它仍然依赖于具体的计算机，但比机器语言方便得多，比较熟悉计算机的用户就可以用它编写程序了。

无论是机器语言还是汇编语言，它都程度不同的依赖于具体的计算机，程序的通用性受到一定的影响。为了解决这个问题，算法语言诞生了，如BASIC、FORTRAN、PASCAL、C和COBOL等语言。它们通常称为高级语言，基本上独立于具体的计算机，对用户的要求也比较低，这就使计算机走出了计算机世界，为各行各业服务，开辟了计算机应用的广阔天地，反过来又推动了计算机的高速发展。

最近数据库的普及和推广方兴未艾。数据库给用户提供了更多的便利，用户可以很容易地建库、查库和改库，可以进行统计和分析，为计算机用于管理工作做出了很大的贡献。

在计算机语言中，不同的语言使用不同的操作名称，通常机器语言使用代码，汇编语言使用指令，高级语言使用语句，数据库使用命令。无论指令、语句还是命令，它们之所以能执行，主要是系统提供了相应的语言处理程序——汇编程序、编译程序和数据库程序。

本书讨论的PASCAL语言和C语言是高级语言，都靠相应的编译程序来编译执行。

程序设计中经常要考虑两个问题：一个是语法，即所使用的语言的一系列基本规定；另一个是算法，即使用什么方法来进行程序设计。在程序设计的学习阶段，应该特别注意语法关系，掌握该语言的数据类型、变量说明以及语句的规则。而在程序设计的实践阶段，则必须加强算法的训练，只有选择正确的和较好的算法，才能节省存贮空间，节省运行时间，得到正确的结果。只重视语法、不重视算法，则不可能成为一个好的程序设计者。本书中主要介绍两种高级语言的语法规则，同时也介绍一些常用的算法，以便于读者进行较好的程序设计。

程序设计的目的是为了帮助人们解决社会实践中的各种问题，一般要经历以下几个步骤：

1. 从社会实践中提出问题；
2. 根据问题确定方法——构造数学模型，选择计算方法，确定所用语言；
3. 程序员编制程序——画出程序框图，编写程序并上机运行和调试程序；
4. 在社会实践中检验程序，如果正确无误，达到要求，则设计基本完成。否则，要进行反复修改，反复调试，直到满足要求为止。

程序员的职责，就是编制高质量的程序。所谓高质量的程序，其基本要求是正确可靠和简单清晰，同时在满足结构程序设计模块化要求的前提下，尽量做到程序短，占据内存单元少，程序运行速度快。此外还应该做到程序便于修改，调试，通用性强，便于移植。

第一章 PASCAL语言的语句和程序结构

PASCAL程序设计语言，是由瑞士的沃斯教授提出的，最早发表在1971年的《EHT》杂志上。把这种语言取名为PASCAL，是为了纪念法国十七世纪的一位杰出的数学家PASCAL先生。

PASCAL语言是按照结构程序设计的原则设计出来的一种描写算法的语言，它是从ALGOL语言发展过来的，并做了一些重要改进。它特别适用于某些课程的教学，编写计算机的系统软件，也完全适用于科学计算和事务管理。

PASCAL语言有简明灵活的通用语句，丰富完备的数据类型，清晰明了的模块结构，而且书写格式自由，编译紧凑，程序运行效率高，算得上计算机语言中的后起之秀，在广泛的应用领域中发挥着重要的作用。

为了概括PASCAL语言的具体特征，我们还是把设计者沃斯教授本人的一段话抄录在此，供大家参考。

“我们把PASCAL语言的特征列举如下：

1. 变量说明是强制性的。
2. 某些字（例如BEGIN，END，REPEAT等）是“保留”的，不能作标识符。
3. 分号（；）被看作为语句分隔符，而不是语句终止符（如同PL/1语言一样）。
4. 标准数据类型是整数、实数、字符和布尔类型。基本的数据构造工具有数组、记录（对应于COBOL和PL/1的‘STRUCTURE’）、集合和文件。这些结构还可以组合和嵌套，以形成集合的数组，记录的文件等等。数据还可以动态的产生和分配并由指针访问。
5. 集合SET数据结构提供了类似于PL/1语言的“位串”的能力。

6. 数组可以具有任意的维数与任意的界，数组的界（下标）是常量，即没有动态数组。

7. 象 FORTRAN, ALGOL 和 PL/1 语言一样，存在转移语句。标号是无符号整数，必须加以说明。

8. 复合语句如同 ALGOL 语言，相应于 PL/1 语言里的 DO 群。

9. ALGOL 语言中的开关语句和 FORTRAN 语言中的计算转语句的功能，在 PASCAL 中由分情形语句表示。

10. 循环语句相应于 FORTRAN 中的 DO 循环，它的步长只能是 1 或 -1（分别对应 FOR 语句中用 TO 和 DOWNTO 的情况），并仅当控制变量的值落在界内时执行。因此，被控制的语句可能完全没有执行。

11. 没有条件表达式和多重赋值。

12. 过程和函数可以递归调用。

13. 对变量不存在（象 ALGOL 语言里那样）“OWN”属性。

14. 参数有值引用与变量引用，没有名字引用。

15. 就不存在匿名的分程序来说，这里的“分程序结构”不同于 ALGOL 和 PL/1 语言的分程序结构，即每个分程序都给一个名字，并做成一个过程。

16. 常量、变量等所有对象必须在它们被引用之前加以说明。但允许以下两个例外：

(1) 指针类型定义里的类型标识符；

(2) 当存在向前引用时的过程与函数调用。

许多初次接触 PASCAL 语言的人，往往感叹它缺少某些‘中意的特色’，例如方幂运算符、字符串的并列、动态数组、布尔值的算术运算、自动的类型转换和省缺说明等，这都不是疏忽遗漏，而是有意删除的。在某些场合下，有它们反而会招致程序设计解法的效率不高；在另外一些场合，又会感到它们不利于达到程序设计清晰可靠和‘风度优美’的目标。最后，还得考虑对繁多的有效程序设计功能进行严格挑选，以保证编译程序比较紧凑和高效。”

我们将分两章介绍PASCAL语言程序设计的有关内容。第一章，着重PASCAL语言的语句和程序结构，同时介绍PASCAL语言的词汇表、标准的数据类型和标准函数的用法。第二章，着重PASCAL语言的非标准的数据类型，在最后一节，给出几个综合应用的程序例子。

§1-1 词汇表和程序结构

某种高级语言使用的符号的总和通常称为词汇表。本节首先介绍PASCAL语言使用的词汇表，然后讨论程序的结构。

一、PASCAL语言使用的词汇表

1. 符号分类 我们习惯上把PASCAL语言使用的符号分为三类。

第一类：26个大小写的英文字母。

不同的计算机系统对大小写英文字母的处理办法不尽相同。除个别实现的PASCAL语言编译程序外，一般对出现在保留字与用户标识符中的小写英文字母，都作为大写字母处理。

第二类：10个阿拉伯数字字符，0~9。

第三类：专用符号，包括24个运算符与35个关键保留字。

24个运算符如下：

+ - * / := ↑ = ⟨ ⟩ < =
 > = < > () [] { }
 ; ' .. , ; .

在不能使用⟨与⟩的系统中，可以用(•与•)来代替。

2. 保留字和标识符 关键保留字是专用符号，是PASCAL语言中有特定含义的单独符号，用户不能把它们移作它用。35个保留字如下：

AND	ARRAY	BEGIN	CASE
CONST	DIV	DO	DOWNTON
ELSE	END	FILE	FOR
FUNCTION	GOTO	IF	IN

LABEL	MOD	NIL	NOT
OF	OR	PACKED	PROCEDURE
PROGRAM	RECORD	REPEAT	SET
THEN	TO	TYPE	UNTIL
VAR	WHILE	WITH	

用户为了标记常数、类型、变量、过程和函数的名字时，就要定义标识符。有效的标识符必须以字母开头，后面跟一些字母或数字字符，也可以不跟。标识符的长度可以有长有短，但不同版本的编译程序对它们的处理方法可以不同，“标准”PASCAL语言只识别与使用其最前面的八个字符。

例如，**ABCD A B105 A12345678** 和 **A123456789 B** 都是正确的标识符，但标准的PASCAL语言认为后二者是同一个标识符，因为它们的前八个字符完全相同。

反之，**1A A-B** 和 **A, B** 等不是正确的标识符，头一个不是以字母开头，后两个第一个字母开头后，后面跟的有非数字或字母字符。

3. 分隔符 通常用来分隔标识符等；在任何一种计算机语言中，都非常重要。在PASCAL语言中，把空格、行结束符和注释都看作分隔符。并且规定，在任何两个相邻的标识符之间、数之间或单词符号（保留字）之间，至少必须有一个分隔符，连续的多个分隔符与单独一个分隔符具有相同的作用。但在一个标识符内部或一个数内部不允许出现分隔符。分号是语句分隔符。

二、PASCAL语言的程序结构

任何语言写的程序，总要描述数据和对给出的数据进行处理。

从结构上看，我们可以把PASCAL语言的一个程序分为三部分。

第一部分，是PASCAL语言程序的首部。它的作用主要是给出程序名和这个程序中所使用的文件的参数名。

第二部分，是程序的说明部分，它对数据和子程序（过程与函数）进行说明。一般应按下列顺序逐项说明：标号说明，常量

定义，数据类型定义，变量说明，过程与函数说明。但并不是每个程序中都要有全部这些说明，它们是由用户按其需要安排的。

第三部分，是程序的执行部分。它对数据进行全部处理。这是程序的实质部分。

和其它任何一种计算机语言一样，PASCAL语言也有自己的注释部分。它可以出现在程序中任何需要注释的位置。注释是指包括在〈与〉之间的全部内容。它是程序的重要组成部分，能大大提高程序的可读性，对修改与完善一个程序也会有相当的好处。要养成合理使用注释的良好习惯。

图1-1是PASCAL语言程序结构的示意图。

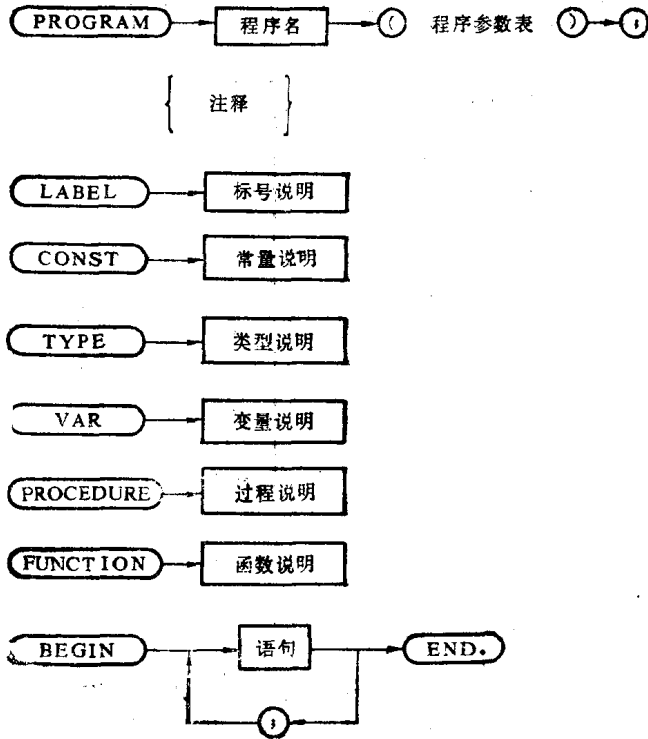


图1-1 PASCAL语言程序结构