

江苏省精品课程配套教材

高等学校计算机系列教材

FORTRAN 95 程序设计

白云 李学哲 陈国新 贾波 编著

清华大学出版社

江苏省精品课程配套教材

高等学校计算机系列教材

FORTRAN 95

程序设计

清华大学出版社

北京

内 容 简 介

本书基于 Compaq Visual FORTRAN 6.6 编译软件,详细介绍了 FORTRAN 95 的基本知识、内部数据类型、数据有格式输入输出、三种控制结构程序设计(顺序、选择、循环)、数组(静态数组、动态数组)、子程序(标准子程序、语句函数、内部子程序、外部子程序、递归子程序、纯子程序、逐元子程序)、FORALL 并行计算、派生类型与结构体、指针、文件、接口、模块、公私属性、操作重载、公用区、存储关联、图形处理、多语言混合编程等,以及程序设计基本概念和 Microsoft Developer Studio 可视化集成开发环境。每章后面附有大量习题,供读者平时练习和上机实验之用,书后提供附录,便于读者速查有关信息。

本书可作为高等院校程序设计课教材及各类计算机培训教材和工程技术人员的参考用书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

FORTRAN 95 程序设计 / 白云等编著. —北京:清华大学出版社, 2011.4

(高等学校计算机系列教材)

ISBN 978-7-302-24173-7

I. ①F… II. ①白… III. ①FORTRAN 语言—程序设计—高等学校—教材
IV. ①TP312

中国版本图书馆 CIP 数据核字(2010)第 243029 号

责任编辑:白立军 薛 阳

责任校对:梁 毅

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954, jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260

印 张:28.25

字 数:670千字

版 次:2011年4月第1版

印 次:2011年4月第1次印刷

印 数:1~4000

定 价:45.00元

产品编号:037411-01

前 言

FORTRAN 语言对程序设计语言的研究和发展产生了极其深刻而广泛的影响,它被广泛应用于数值分析、工程设计和科学计算等领域,为社会进步和经济发展做出了巨大贡献,并以严谨、规范、高效的特有功用受到人们的普遍青睐。

FORTRAN 语言具有顽强的生命力。在 50 多年的发展过程中,功能不断增强,版本不断更新,影响不断扩大,历经岁月,经久不衰。在面向对象和可视化程序设计技术风靡全球,以及 Visual C++、Visual Basic、Visual FoxPro 等可视化编程语言盛行天下的今天, FORTRAN 语言不甘落后,积极引入先进思想和理念,紧跟程序设计语言前进的步伐,于 1991 年发布和推出了功能强大的 FORTRAN 90,使传统 FORTRAN 语言焕发出现代气息和新的活力。之后不久,功能更加强大的 FORTRAN 95 面世,引入全新的并行化计算概念,新增了诸如纯过程、逐元过程、FORALL 结构等新功能,有效地摆脱了线性存储模式的制约,使 FORTRAN 95 成为描述并行计算的标准语言,并在并行计算领域中大展身手,独领风骚。

数据设备公司(DEC)与美国微软公司(MS)强强联合,优势互补,合作研究、开发和推出了功能强大的 Digital Visual FORTRAN 5.0,使 Microsoft Developer Studio 可视化集成开发环境与优质 FORTRAN 编译器得到完美结合。Compaq 公司收购 DEC 公司后,推出了最新的支持 FORTRAN 95 的 Compaq Visual FORTRAN 6.6。Compaq Visual FORTRAN 6.6 的推出,使 FORTRAN 95 锦上添花,更胜一筹,在社会上产生了广泛影响,受到人们的普遍欢迎。

本书通俗易懂,循序渐进,由浅入深,突出概念,注重方法,强化编程,并配以简洁说明、丰富实例和详细题解,为读者提供全面、详尽、实用的 FORTRAN 95 程序设计内容。每章后面附有大量习题,供读者平时练习和上机实验之用,书后提供附录,便于读者速查有关信息。

本书可用作高等院校程序设计课教材、计算机培训教材和工程技术人员参考用书。本书有配套的电子教学课件和实验教材,需要者可与作者(by59@163.com)或出版社联系。

苏州大学钱培德教授负责本书主审工作,详细审阅了全部书稿,提出了许多建设性的修改意见,保证了本书的水平和质量,在此表示诚挚的谢意。

本书由白云、李学哲、陈国新和贾波编著,刘怡、刘敏、高洁羽和邱劲等同志参加了本书的部分编写工作,在此表示感谢。白云教授负责全书策划、审阅和定稿。

由于编者水平有限,书中难免出现错误和不妥之处,恳请专家和读者批评指正。

编 者
2010 年 6 月

目 录

第 1 章 程序设计概述	1
1.1 程序设计语言	1
1.1.1 程序设计语言分类.....	1
1.1.2 程序设计语言概念.....	2
1.1.3 程序设计语言形式化描述.....	3
1.2 程序和程序设计	4
1.2.1 程序.....	4
1.2.2 程序设计.....	5
1.2.3 程序设计步骤.....	5
1.3 算法的基本概念和特性	5
1.3.1 算法含义.....	6
1.3.2 算法特性.....	6
1.3.3 算法评价标准.....	6
1.3.4 算法描述工具.....	7
1.4 程序设计方法.....	11
1.4.1 模块化程序设计方法	11
1.4.2 结构化程序设计方法	12
1.4.3 面向对象程序设计方法	12
习题 1	12
第 2 章 FORTRAN 95 软件开发环境	15
2.1 FORTRAN 语言发展概述	15
2.2 FORTRAN 95 主要特征	16
2.3 Compaq Visual FORTRAN 6.6 主要特性.....	17
2.4 Compaq Visual FORTRAN 6.6 硬件配置和安装.....	19
2.5 Microsoft Developer Studio 软件开发环境概述	19
2.5.1 Microsoft Developer Studio 简介	19
2.5.2 工作空间(Workspace)和项目(Project)	20
2.5.3 启动 Microsoft Developer Studio	21
2.5.4 Microsoft Developer Studio 主窗口说明	22
2.6 Microsoft Developer Studio 解题过程	27
2.6.1 前期准备工作	27
2.6.2 创建工作空间	27

2.6.3	创建项目	28
2.6.4	创建源程序文件	29
2.6.5	创建辅助文档文件	30
2.6.6	编译源程序文件	32
2.6.7	构建可执行程序文件	33
2.6.8	运行程序	34
2.6.9	调试程序	35
2.6.10	程序多区域显示	38
习题 2	39
第 3 章	FORTRAN 95 基础知识	42
3.1	字符集	42
3.2	名称	42
3.3	关键字	43
3.4	程序结构	44
3.5	语句	46
3.6	程序书写格式	47
3.7	数据类型	48
习题 3	50
第 4 章	内部数据类型与顺序结构程序设计	51
4.1	常量	52
4.1.1	整型常量	52
4.1.2	实型常量	53
4.1.3	复型常量	57
4.1.4	字符型常量	57
4.1.5	逻辑型常量	59
4.2	变量	59
4.2.1	变量名称	59
4.2.2	变量类型	60
4.2.3	整型变量	61
4.2.4	实型变量	62
4.2.5	复型变量	63
4.2.6	字符型变量	64
4.2.7	逻辑型变量	65
4.3	表达式	66
4.3.1	算术表达式	67
4.3.2	字符表达式	69

4.3.3	关系表达式	70
4.3.4	逻辑表达式	71
4.3.5	表达式运算的误差问题及其解决办法	73
4.4	赋值语句	74
4.4.1	算术赋值语句	75
4.4.2	字符赋值语句	76
4.4.3	逻辑赋值语句	77
4.5	表控输入输出语句	77
4.5.1	表控输入语句	77
4.5.2	表控输出语句	80
4.6	参数语句(PARAMETER 语句)	83
4.7	函数	85
4.8	END 语句、STOP 语句和 PAUSE 语句	87
4.9	简单顺序程序设计举例	87
习题 4		91

第 5 章	数据有格式输入输出	95
5.1	概述	95
5.2	格式说明与格式编辑符	95
5.3	整数有格式输入输出	98
5.4	实数有格式输入输出	100
5.4.1	小数型实数有格式输入输出	100
5.4.2	指数型实数有格式输入输出	102
5.4.3	双精度型实数有格式输入输出	104
5.5	复数有格式输入输出	104
5.6	字符串有格式输入输出	105
5.6.1	A 编辑符	105
5.6.2	字符串编辑符	106
5.7	逻辑值有格式输入输出	107
5.8	二、八、十六进制数据有格式输入输出	108
5.8.1	数据二进制编码表示	108
5.8.2	二、八、十六进制数据有格式输入输出	110
5.9	不可重复编辑符	113
5.9.1	X 编辑符	113
5.9.2	/编辑符	114
5.9.3	\编辑符	115
5.9.4	T、TL、TR 编辑符	115
5.9.5	:编辑符	116

5.9.6	Q 编辑符	116
5.10	纵向走纸控制	117
5.11	输入输出表与格式说明表的相互关系	119
5.12	有格式输入输出应用举例	122
	习题 5	123
第 6 章	选择结构程序设计	127
6.1	概述	127
6.2	逻辑 IF 语句	128
6.3	块 IF 语句	130
6.3.1	基本块 IF 语句	130
6.3.2	多分支块 IF 语句	133
6.3.3	嵌套块 IF 语句	137
6.3.4	块 CASE 选择结构	138
6.4	选择结构程序设计举例	140
	习题 6	143
第 7 章	循环结构程序设计	145
7.1	循环结构概述	145
7.1.1	循环及循环结构	145
7.1.2	循环结构分类	147
7.2	用 DO 循环语句实现计数型循环	149
7.2.1	DO 循环语句	149
7.2.2	DO 循环语句的执行过程	151
7.3	用 DO WHILE 循环语句实现条件型循环	153
7.3.1	DO WHILE 循环语句	153
7.3.2	DO WHILE 循环语句的执行过程	154
7.4	强制性终止循环过程(EXIT、CYCLE)	155
7.5	特殊 DO 循环语句与 EXIT 语句	157
7.6	循环嵌套	158
7.7	循环结构程序设计举例	160
	习题 7	166
第 8 章	数组及其应用	168
8.1	概述	168
8.2	数组声明	170
8.2.1	用类型声明语句进行数组声明	170
8.2.2	用 DIMENSION 语句进行数组声明	172

8.2.3	用类型声明语句和 DIMENSION 属性进行数组声明	172
8.3	数组元素引用	173
8.4	数组的逻辑结构和存储结构	177
8.5	数组的输入输出	179
8.5.1	使用 DO 循环语句输入输出数组	180
8.5.2	使用隐含 DO 循环子句输入输出数组	182
8.5.3	使用数组名或数组片段输入输出数组	184
8.6	给数组赋初值	185
8.6.1	通过数组构造器给数组赋初值	185
8.6.2	通过 DATA 语句给数组赋初值	187
8.7	WHERE 语句	188
8.7.1	逻辑 WHERE 语句	188
8.7.2	块 WHERE 语句	189
8.7.3	多分支 WHERE 语句	190
8.8	FORALL 语句	191
8.8.1	逻辑 FORALL 语句	192
8.8.2	块 FORALL 语句	193
8.9	动态数组	196
8.10	数组应用举例	200
	习题 8	213
第 9 章	函数与子例行程序	216
9.1	概述	216
9.2	标准子程序	222
9.2.1	标准函数	222
9.2.2	标准子例行程序	224
9.3	语句函数	226
9.4	内部子程序	230
9.4.1	内部函数子程序	231
9.4.2	内部子例行程序	236
9.5	子程序形参与实参之间的数据传递关系	241
9.5.1	变量作形式参数	242
9.5.2	数组作形式参数	243
9.5.3	可调数组作形式参数	246
9.5.4	假定大小数组作形式参数	249
9.5.5	假定形状数组作形式参数	250
9.5.6	子程序名作形式参数	251
9.5.7	星号作形式参数	256

9.5.8 形式参数与实在参数的结合关系	257
9.6 递归(RECURSIVE)子程序	258
9.6.1 递归定义	258
9.6.2 递归函数	258
9.6.3 递归子例行程序	260
9.7 外部(EXTERNAL)子程序	261
9.8 纯(PURE)子程序	267
9.9 逐元(ELEMENTAL)子程序	269
习题 9	270
第 10 章 派生类型与结构体	272
10.1 概述	272
10.2 派生类型定义	274
10.3 结构体声明	276
10.4 结构体成员引用	277
10.5 结构体初始化	277
10.6 结构体数据输入输出	279
10.6.1 结构体数据输入	279
10.6.2 结构体数据输出	280
10.7 派生类型应用举例	281
习题 10	285
第 11 章 指针与动态数据结构	287
11.1 概述	287
11.1.1 指针的存储结构和访问方式	287
11.1.2 指针变量和目标变量声明	289
11.1.3 指针状态	290
11.2 指针引用和赋值	290
11.2.1 指针引用	290
11.2.2 指针赋值	292
11.2.3 派生类型、结构体与指针	293
11.3 整型指针	294
11.4 与指针相关的函数和语句	296
11.4.1 置空语句(NULLIFY)和置空函数(NULL)	296
11.4.2 链接测试函数(ASSOCIATED)	296
11.4.3 动态存储分配语句(ALLOCATE、DEALLOCATE)	296
11.5 指针数组	297
11.5.1 指针数组	297

11.5.2	指针数组作为函数返回值	298
11.6	动态链表	299
11.6.1	链表概念	299
11.6.2	创建和搜索链表	300
11.6.3	插入结点	302
11.6.4	删除结点	304
11.6.5	环形和双向链表	306
习题 11		308
第 12 章	文件与设备	309
12.1	概述	309
12.2	物理设备和逻辑设备	311
12.3	文件	312
12.3.1	外部文件	312
12.3.2	内部文件	313
12.3.3	文件标识	314
12.4	外部文件分类	315
12.4.1	文件存储格式	315
12.4.2	文件存取方式	316
12.4.3	文件定位	318
12.5	文件基本操作	318
12.5.1	文件打开(OPEN)	318
12.5.2	文件关闭(CLOSE)	323
12.5.3	文件结束(ENDFILE)	324
12.5.4	文件输入(READ)	325
12.5.5	文件输出(WRITE)	328
12.5.6	文件查询(INQUIRE)	328
12.5.7	文件指针反绕(REWIND)	334
12.5.8	文件指针回退(BACKSPACE)	334
12.6	使用外部设备	335
12.7	文件的读写操作	335
12.7.1	有格式顺序存取文件读写操作	335
12.7.2	有格式直接存取文件读写操作	337
12.7.3	无格式顺序存取文件读写操作	338
12.7.4	无格式直接存取文件读写操作	339
12.7.5	二进制顺序存取文件读写操作	340
12.7.6	二进制直接存取文件读写操作	341
12.8	文件应用举例	342

习题 12	346
第 13 章 接口、模块与重载	347
13.1 接口界面块	347
13.2 模块	349
13.3 重载(OVERLOAD)	354
13.3.1 子程序重载	354
13.3.2 操作符重载	356
13.3.3 赋值号重载	360
习题 13	361
第 14 章 公用区、存储关联与数据块子程序	362
14.1 公用区	362
14.1.1 COMMON 语句	363
14.1.2 公用区特性和使用规则	364
14.1.3 有名公用区的作用	366
14.2 存储关联(等价)	367
14.2.1 EQUIVALENCE 语句	368
14.2.2 EQUIVALENCE 语句使用规则	369
14.3 数据块子程序	370
习题 14	373
第 15 章 图形处理	375
15.1 图形处理概述	375
15.1.1 坐标系统	375
15.1.2 图形颜色	379
15.1.3 线型	381
15.1.4 填充样式	381
15.2 QuickWin 绘图设计基础	382
15.3 设置 QuickWin 窗口	386
15.4 多窗口操作	387
15.5 常用绘图函数	392
15.5.1 绘制像素点	392
15.5.2 设置当前绘图点	393
15.5.3 绘制直线	393
15.5.4 绘制形状	394
15.6 文字信息的显示	399
15.7 图形应用实例	402

习题 15	412
第 16 章 Visual Basic 与 FORTRAN 95 的多语言混合编程	414
16.1 概述.....	414
16.1.1 动态链接库.....	414
16.1.2 动态链接库的优点.....	415
16.1.3 动态链接库的实现方法.....	415
16.2 Visual Basic 与 FORTRAN 95 混合编程	415
16.2.1 Visual Basic 与 FORTRAN 95 混合编程基础	415
16.2.2 Visual Basic 与 FORTRAN 95 混合编程步骤	419
16.3 Visual Basic 与 FORTRAN 95 混合编程应用	420
习题 16	428
附录 A FORTRAN 95 标准函数简表	430
附录 B FORTRAN 95 标准子例行程序简表	436
参考文献.....	438

第 1 章 程序设计概述

对于计算机系统来说,如果把 CPU 比作心脏,那么程序就是其灵魂。学习和掌握程序设计是驾驭和运用计算机系统的重要一环。本章主要介绍程序设计的一些基本知识、基本概念和基本方法,为学习和掌握 FORTRAN 95 程序设计奠定基础。

1.1 程序设计语言

任何计算机系统都是在“程序”的控制下自动工作的,人们通过某种程序设计语言编写的程序来指挥和控制计算机运行。程序设计语言是人与计算机进行交流的重要工具,伴随计算机的诞生而诞生,伴随计算机的发展而发展,在计算机科学技术发展历程中,发挥了巨大作用。

1.1.1 程序设计语言分类

从计算机诞生之日起,程序设计语言从低级语言到高级语言,从传统语言到现代语言不断地向前发展,新的、功能强大的程序设计语言不断涌现。程序设计语言种类很多,在广泛使用的程序设计语言有几十种。

程序设计语言属于人工语言、形式语言、非自然语言,分为低级语言和高级语言两种。

低级语言又分为机器语言和汇编语言。机器语言是一种计算机能直接识别、理解和执行的程序设计语言,它由 0 和 1 两个二进制符号按照确定的规则进行排列和组合。例如,计算 $3+4$,并从屏幕上输出,用 MCS-48 微型计算机上配置的机器语言编写程序为:

```
0010001100000011    -----将 3 送至累加器 A 中
0000001100000100    -----将 4 与累加器 A 相加,结果 7 存储在累加器 A 中
00000010             -----将累加器 A 中的数 7 从总线输出
```

机器语言与具体的计算机硬件有关,不同种类计算机配置不同的机器语言,如不同国家、不同民族有不同的自然语言一样。对计算机来说,识别、理解和执行机器语言很容易,但对人类来说却是苦不堪言。只有极少数非常专业的计算机专家才使用机器语言编写程序,绝大多数人不用机器语言编写程序,但是了解一点机器语言知识也有益处。机器语言的最大优点是运行效率极高,在一些嵌入式控制系统中有时会用到机器语言程序。机器语言的缺点是难以理解、阅读和编写,程序的可靠性、可维护性及可移植性差。

为解决机器语言存在的突出问题,克服其不足和缺点,开发了汇编语言(或称符号语言)。汇编语言是用一些易于理解的符号来取代机器语言中难于理解的二进制编码,如用 ADD 代替加法的二进制编码 00000011。用汇编语言代替机器语言编写程序,可提高程序的可读性、可靠性和可维护性。例如,计算 $3+4$,并从屏幕上输出,用 MCS-48 微型计

计算机上配置的汇编语言编写程序为：

```
MOV A, 3H          -----将 3 送至累加器 A
ADD A, 4H          -----将 4 与累加器 A 相加, 结果 7 存储在累加器 A 中
OUTL BUS, A       -----将累加器 A 中的数 7 从总线输出
```

汇编语言同机器语言一样,与计算机硬件有关,不同种类的计算机配置了不同的汇编语言。汇编语言的优点是执行速度快,占用空间少,运行效率高;其缺点是程序不易编写、理解、维护和移植。一般情况下不使用汇编语言,但是在一些嵌入式控制系统中还习惯采用其编写程序,以提高运行效率,所以汇编语言还有一定的生存空间。

汇编语言程序必须转换成机器语言程序后,才能被计算机识别、理解和执行。转换工作由一个称为“汇编程序”的软件来自动完成。转换过程称为“汇编”,被转换的程序称为“源程序”,转换后的机器语言程序称为“目标程序”。

高级语言是指其表达方式接近自然语言和数学语言的一类程序设计语言。高级语言克服了低级语言的固有缺点。它面向过程、易读易学、易于编写、可靠性高、可维护性好,基本上与计算机硬件无关,能在不同类型计算机上运行。例如,计算 $3+4$, 并从屏幕上输出,用高级语言编写程序只需用一条语句“PRINT *, 3+4”即可。对于一个复杂问题,若用高级语言编写程序来解决,其优点更加明显。下面是计算物体运动距离的 FORTRAN 95 语言程序。

```
!计算物体运动距离
PROGRAM calculate
REAL v0,a,t,s
READ *,v0,a,t      !输入初速度、加速度和运动时间
s=v0*t+0.5*a*t*t
PRINT *, 's=',s
END
```

用高级语言描述解题算法很方便,故高级语言也称为“算法语言”。高级语言的出现开创了计算机科学技术发展的新纪元,是计算机科学技术发展史上的一次革命性事件。FORTRAN 语言是最早出现的高级语言,受其影响,之后不同风格、不同用途、不同规模、不同版本的高级语言不断涌现,目前高级语言有上百种,但常用的只有十多种。不同的高级语言,其词法、语法和语义不尽相同,应用范围也存在差异。

高级语言程序不能被计算机识别、理解和执行,必须将其转换成机器语言程序。转换工作由称为“编译程序”的软件自动完成。转换过程称为“编译”,被转换的高级语言程序称为“源程序”,转换后的机器语言程序称为“目标程序”。

1.1.2 程序设计语言概念

程序设计语言同自然语言一样都有严格的词法、语法和语义规定,而不同之处在于程序设计语言不允许出现二义性和不确定性。词法、语法和语义概念是程序设计语言中的 3 个最基本的概念,学习程序设计语言的一个重要内容就是深入理解和全面掌握其词法、语法和语义规定。

1. 词法

组成合法语句的基本单位称为单词,构成合法单词的规则称为词法。词法通常采用严格的形式化描述,也可采用语法图描述。程序设计语言规定了一组描述词法的规则,称为词法规则。通过词法规则可判断一个单词是否合法。程序中单词必须符合语言的词法规则,如语句“PRINT *,ave”中的“PRINT”、“*”、“,”、“ave”都是符合 FORTRAN 95 语言词法规则的合法单词。如果将 PRINT 改写成 PRN,由于 PRN 不是 FORTRAN 95 语言的合法单词,所以修改后的语句是错误的。

2. 语法

具有特定含义的符号串或句子称为语句,构成合法语句的规则称为语法。语法通常采用形式化描述,也可采用语法图描述。程序设计语言规定了一组描述语法的规则,称为语法规则。通过语法规则可判断一个语句是否合法。程序中语句必须符合语言的语法规则,如语句“PRINT *,ave”就是符合 FORTRAN 95 语言语法规则的合法语句,如果将其改写成“PRINT +,ave”就不是一个合法语句。虽然“PRINT”、“+”、“,”、“ave”都是 FORTRAN 95 语言的合法单词,但 PRINT 后直接跟一个“+”号,不符合 PRINT 语句的语法规则,所以修改后的语句是错误的。

3. 语义

合法语句的实际含义称为该语句的语义。语义通常采用自然语言描述。程序设计语言规定了每一个语句的语义。对语义的理解是否准确,直接影响程序的可靠性和正确性。语义理解有误,程序就可能蕴涵错误,准确理解语句的语义,对于程序设计来说至关重要。若将输出语句“PRINT *,ave”的语义理解为输入 ave 值,则程序运行结果就不可能正确。

1.1.3 程序设计语言形式化描述

程序设计语言的词法和语法一般采用形式化描述。形式化描述的手段有很多,BNF(巴科斯范式)是最常用的一种描述手段,也是一种特定的形式体系,可看作是描述程序设计语言的形式化语言,它定义了一组描述语言的规则。通常称 BNF 为元语言,称程序设计语言为对象语言,称规则为产生式,称规则中不属于对象语言的符号为元语言符号(连接词),称属于对象语言的符号为终结符,称定义产生式的符号为非终结符。BNF 中的元语言符号如表 1-1 所示。

表 1-1 元语言符号及含义

元语言符号	含 义	元语言符号	含 义
→	定义为	{ }	表示括号内对象可出现任意次,也可不出现
	或者	< >	表示括号内为元语言符号,非终结符

续表

元语言符号	含 义	元语言符号	含 义
()	表示括号内对象一定取其中之一	“”	表示引号内为对象语言符号, 终结符, 可省略
[]	表示括号内任取一个对象, 也可不取		

例如, 采用 BNF 描述 FORTRAN 95 语言“标识符”的一组产生式规则是:

```
<标识符> → <字母> { <字母> | <数字> | "_" }  
<字母> → "大写英文字母" | "小写英文字母"  
<数字> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
```

这组规则严格而精确地定义了 FORTRAN 95 语言中的标识符, 其含义是: 标识符由若干个字符组成, 第一个字符必须是大写英文字母或小写英文字母, 其后可以是大写英文字母、小写英文字母、数字或下划线。这组规则没有规定标识符长度, 具体长度由具体计算机系统或编译程序规定, 需查阅编译系统用户手册。由此可以看出 BNF 描述比自然语言描述更简洁、规范、精确和严格。本书采用 BNF 描述。词法和语法有时也用图形(语法图)描述。

1.2 程序和程序设计

对于程序设计语言, 不但要学习其词法、语法和语义, 更重要的是学习如何灵活使用这些规则来求解实际问题, 学习编写程序的方法来让计算机按照人们的意愿工作。

1.2.1 程序

程序设计语言是全部计算机指令(语句)的集合, 程序是符合程序设计语言词法、语法和语义规则的计算机指令(语句)序列, 语句序列规定了语句的执行顺序。用程序设计语言可写出许多不同的程序, 计算机的神奇之处就是可以执行许多奥妙无穷的程序。

下面是一个计算任意圆柱体体积的 FORTRAN 95 程序。

```
!已知圆柱体的半径 r 和高度 h, 求它的体积 v, 并输出  
PROGRAM Volum  
READ *, r, h  
V=3.1415 * r * r * h  
PRINT *, 'V= ', V  
END
```

程序有小型、中型、大型和特大型之分, 上述程序就是一个小型程序, Word 文字处理程序是一个大型程序, Windows 操作系统程序是一个特大型程序。程序加文档说明通常称为软件。由于中型、大型和特大型程序一般都配置详细的文档说明, 所以称其为软件, 如 Word 文字处理程序一般称为“Word 软件”。软件按性质可分为系统软件、支撑软件和