

宋士华 钟美清 编著

FORTRAN77 和 结构化程序设计

浙江大学出版社

FORTRAN77
和结构化程序设计

宋士华 钟美清 编著

浙江大学出版社

FORTRAN77和结构化程序设计

宋士华 钟美清 编著

责任编辑 应伯根

*

浙江大学出版社出版

浙江大学印刷厂印刷

浙江省新华书店发行

*

开本: 850×1168 1/32 印张: 10.75 字数: 278千字

1990年6月第1版 1990年6月第1次印刷

印数: 0001—3000

ISBN 7-308-00398-1

TP·031 定价: 2.75元

前 言

多年来，我们本着使学生通过FORTRAN 语言的学习，打下今后使用计算机的扎实基础。本着 FORTRAN教材应有先进性、实用性和趣味性的要求，多次改革了教材。书中把语言教学和程序设计方法教学结合起来，使读者不仅能学到一些语法规则，而且还能学到先进的程序设计方法，使他们从一开始就养成良好的编程风格。

经过多次改革，並吸收国内外好的经验，使本教材逐步形成了自己的特色。

强调了程序设计方法：结构化程序设计是一种先进的程序设计方法，是软件工程的重要基础。为了在FORTRAN 程序设计中推广这种方法，本教材特用第五、第十章篇幅详细介绍了结构化程序设计的基本概念和实现方法。並用这种程序设计思想贯穿和指导其它各章，以便于读者更好地领会和掌握这一设计思想。

突出重点：随着中学计算机的普及，越来越多的高中毕业生已经掌握了BASIC 语言的基础知识。因此，他们入大学后学习FORTRAN 语言时，应该有较高的起点。所以，在本教材中大大精简了那些和BASIC 基本类似的常见的数据类型和语句；大大加强了FORTRAN 语言的重点，即过程和数据文件的内容。这些也正是读者学习中的难点。例如，我们用第六、第七、第八三章的篇幅介绍了过程，特别是像过程之间的数据传递等，作了较为深入的介绍。使读者不仅知其然，而且知其所以然。这对解除读者学习中的难点是会有益的。

适应面宽：本教材不仅传统地介绍了FORTRAN 语言在科学计算，即数值计算方面的应用；而且，结合 FORTRAN77 的特

点，大大加强了非数值计算方面的应用。

在语言标准方面，本教材坚持按FORTRAN77全集的标准来编写。但考虑到使用众多的微机（如IBM-PC）的读者，他们只能使用FORTRAN77子集（或子集的扩充集）。为适应这部分读者的需要，我们介绍了子集与全集的区别和限制。并且还适当地介绍了如何在FORTRAN77子集的范围内实现某些属于全集的功能，这对使用子集的读者来说，是会有启发的。

注重实践：为了有利于提高读者的程序设计能力，本教材从下列几方面强调了实践。1.在内容安排上作了改革，使读者学完第一章之后就可自己编写简单程序上机实习。2.大多数例题都是先说明设计思想，然后写出程序，以帮助读者理解为什么要这样编。3.每章都有一定数量的有一定难度的趣味性习题，目的是提高读者分析解决问题的兴趣和能力。4.增加了程序调试的内容，以便启发读者怎样去分析和查找错误，解决问题。

本教材在校内使用过程中，也引起校外不少单位的注意和重视。例如，在由合肥工业大学、吉林大学等7所高校中的10位教师编写，机械工业出版社出版的《FORTRAN77结构化程序设计》一书中，就引用了本教材85年版讲义上的内容。

本教材最后一次改写时，钟美清改写了第一、二、八章并编写了附录四；宋士华改写了其余各章和附录。在编写、使用和改写过程中，人工智能专家何志均教授曾为本书的编写作过重要的指导。

杭州大学的张森教授，杭州电子工学院的汤子瀛副教授，浙江大学的余瑞剑教授都严格评审过本书。此外，使用过本教材的我校多位教师也提过宝贵意见。在此，我们向一切关心和帮助过本书的同志，包括我校教材科和出版社的有关同志的大力支持表示衷心感谢。

由于我们水平有限，不当和错误之处在所难免，望能批评指正。

编者 1990年3月于浙江大学

目 录

第一章 简单程序	1
§1.1 信息存贮	1
§1.2 源程序的书写格式	4
§1.3 常数和变量	7
§1.4 常用函数	11
§1.5 算术表达式和算术赋值语句	13
§1.6 数组和数组元素	16
§1.7 表控输入/输出语句	20
§1.8 STOP 和END 语句	26
§1.9 可执行语句和非执行语句	27
习题一	28
第二章 控制执行	30
§2.1 逻辑条件语句(逻辑IF)	30
§2.2 多重条件	34
§2.3 继续语句 (CONTINUE)	41
§2.4 DO 循环	41
§2.5 IF THEN ELSE语句	48
习题二	56
第三章 有格式输入/输出	59
§3.1 格式语句 (FORMAT)	59
§3.2 有格式输出	61
§3.3 有格式输入	70

§3.4	记录	76
§3.5	格式说明和输入/输出表的相互作用	79
§3.6	数组的输入/输出——隐循环表	81
	习题三	86
第四章	字符型数据	89
§4.1	字符表达式和字符赋值语句	89
§4.2	字符数据的输入/输出	92
§4.3	字符子串	96
§4.4	字符串的比较	99
§4.5	字符函数	102
§4.6	字符格式说明	109
	习题四	111
第五章	编程风格	113
§5.1	基本的程序结构	113
§5.2	提高执行效率——程序的优化	121
§5.3	节省存储单元	131
§5.4	提高程序质量的其它措施	135
§5.5	程序的调试	136
	习题五	145
第六章	子程序	149
§6.1	子程序的概念	149
§6.2	哑实结合	154
§6.3	调用和被调用程序	168
	习题六	171
第七章	函数	175
§7.1	内部函数	175
§7.2	外部函数	176
§7.3	函数的副作用	188

§7.4	语句函数	190
	习题七	195
第八章	辅程序的进一步特性	197
§8.1	可调数组	197
§8.2	过程名作参数——EXTERNAL 语句 和INTRINSIC语句	209
§8.3	公用语句 (COMMON)	218
§8.4	SAVE 语句	230
§8.5	数据块子程序	233
	习题八	234
第九章	数据文件	237
§9.1	数据文件的概念	237
§9.2	数据文件的打开和关闭	242
§9.3	顺序存取	250
§9.4	直接存取	258
§9.5	内部文件	266
§9.6	询问语句 (INQUIRE)	268
	习题九	271
第十章	程序设计方法	273
§10.1	模块化	273
§10.2	自顶向下和逐步求精	276
§10.3	程序设计举例	281
	习题十	295
第十一章	其它	296
§11.1	双精度和复型数据	296
§11.2	算术类型之间的转换	301
§11.3	输入/输出补充	304
§11.4	具有多重出口的控制转移语句	311

§11.5 辅程序的多重入口和可变返回点	316
附录	323
附录一 ASCII 码字符表	323
附录二 内部函数表	324
附录三 FORTRAN 语句在程序单位中的排列顺序	330
附录四 FORTRAN77 语句汇总表	332

第一章 简单程序

这一章我们介绍常用的数据类型，算术表达式，赋值语句以及表控输入和输出语句。学习这一章之后，就能编写一些简单程序。有条件的读者就可以练习上机操作，运行你自己编制的程序。

在这一章的开始，还应简要地介绍计算机中信息存贮的基本概念和FORTRAN 源程序的书写格式。

§1.1 信息存贮

我们知道，一台电子计算机是由运算器、控制器、存贮器和输入/输出设备等四个基本部分组成。常用的输入/输出设备有显示终端，打印机，磁盘机（软磁盘，硬磁盘）等。需要运行的程序和数据都应存放在存贮器中。存贮器又称为内存。

一、存贮器的结构

存贮器是由一个一个的单元组成。每个单元都有一个编号，称为“地址”。存贮器所有单元的总和称为这个存贮器的容量。每一个单元又分成若干位(bit)，每一位可存放一个二进制数（0或1）。一个单元包含位的总数称为字长。存贮器的容量和字长是计算机的重要指标之一。在存贮器中，存放信息的基本单位是字节(byte)。一个字节是由8位组成。因此，一个单元可包含一个或多个字节。例如，字长有32位的存贮单元，包含了四个字节。

存储器的每一个单元都具有如下性质：

取数（读出单元中的信息）时，不破坏单元中原有的信息。

存数（向一个单元存入一个新的信息）时，若该单元中原来存有信息，就把原有信息删除，存入新信息。

二、二进制数

我们习惯使用的是十进制数，逢10进1。而在计算机中经常使用的是二进制数。它只有0和1二个数，逢2进1。1~10这10个十进制数对应的10个二进制数是：

十进制： 1 2 3 4 5 6 7 8 9 10

二进制： 1 10 11 100 101 110 111 1000 1001 1010

一个十进制数 $3124 = 3 \times 10^3 + 1 \times 10^2 + 2 \times 10^1 + 4 \times 10^0$

一个二进制数 $10111 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$
 $= 16 + 4 + 2 + 1 = 23$

如果一个二进制数比较大，看起来就很不习惯，如110100101。为了克服这个缺点，计算机中还经常使用八进制数，逢8进1。

二进制数转换为八进制数，只要把二进制数从个位数开始，每三位一组分开计算，就得对应的八进制数（即用三位二进制数表示一位八进制数）。如：

$110, 100, 101 = (645)_8$ 。

（括号外面的8表示八进制数。）

$1, 010, 101, 111, 011 = (12573)_8$ 。

当然，八进制数化为二进制数，只要把每一位数化成对应的三位二进制数就是：

$(3704)_8 = 11111000100$

$(1777)_8 = 1111111111$

计算机上还使用一种数制，即十六进制，逢16进1。它的16个数及其对应的二进制表示（用四位二进制数表示一位十六进制数）

如表1-1所示。

表1-1 16进制数

16进制	2 进制	10进制
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

三、信息存贮

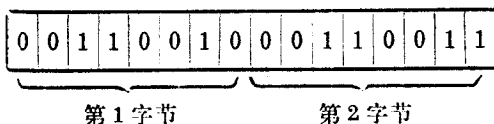
数据在存储器中的存放形式有二种。一种是二进制形式。如整数23在内存的一个字节中的存放形式是：

0	0	0	1	0	1	1	1
---	---	---	---	---	---	---	---

整数 $32767 = 2^{15} - 1$ ，需要二个字节来存放：

0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

另一种是字符形式，即用ASCII码表示，见附录一。从附录一知道，不论是一个数字，字母，还是其它符号都有一个唯一对应的ASCII码。用ASCII码表示的字符称ASCII码字符。每一个ASCII码用八位二进制数表示。即一个内存字节存放一个ASCII码字符。如整数23用字符形式存放，需要二个字节。存放形式如下：



字母和其它特殊字符一般都以ASCII码形式存放。每一个ASCII码的前4位和后4位可分别用一个16进制数来表示。如附录一所示，2的ASCII码16进制值是32，A是41，K是4B等等。

§1.2 源程序的书写格式

FORTRAN源程序的书写格式比pascal或BASIC语言有更严格的规定。一个FORTRAN源程序是由若干行组成的。一行分80列，这80列又分成四个区：1~5列为标号区，第6列为继续行标志区，7~72列为语句区，73~80列为注解区。图1-1列出了一个源程序的书写格式。

这个源程序是依据下列公式

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

求方程 $ax^2 + bx + c = 0$ 的二个实根。程序中的IF语句说明：若通过READ语句读取的系数 $A \neq 0$ ，则计算二个根ROOT1, ROOT2，并通过PRINT语句把结果显示或打印出来。否则($A = 0$)，就执行GOTO语句，使控制立即转移到标号为100的STOP语句，从而什

么也不计算就终止程序的执行。

1	2	5	6	7	72	73	80
C				The roots of quadratic			1
C				equation.			2
				READ *, A, B, C			3
				IF (A, EQ, 0.0) GOTO 100			4
				D = B * B - 4.0 * A * C			5
				ROOT1 = (- B + SQRT(D)) /			6
		1		(2.0 * A)			7
				ROOT2 = (- B - SQRT(D)) /			8
		*		(2.0 * A)			9
				PRINT *, ROOT1, ROOT2			10
	100			STOP			11
				END			12

图1-1 源程序的书写格式

一、源程序的书写规则

1. 注释行 若在一行的第1列上写字母 C 或星号*, 则该行就是注释行。注解的内容可写在2~72列之间。如果注解的内容比较多, 可写成二行或多行。但每一行的第1列都必须有一个字母 C 或星号*。

注解并不是一个源程序的必要成份。计算机并不执行它们, 只是在列源程序清单时, 原样印出注解的内容。

全部是空格的行, 也是注释行。

注释行可写在程序的任何地方。

2. 语句行 所有语句应写在语句区(7~72列之间)。一行只能写一个语句。若语句比较长, 则可以写继续行。当一个语句有继续行时, 则第1行称为起始行, 它的续行称为继续行。

凡是继续行，必须在继续行所在的第6列上写上一个非0非空格的数字或字母。图1-1中二个继续行分别用1和*来标志。在非继续行（始行或语句）的第6列上只能写0或空格。一般我们用空格。

FORTRAN77全集规定，继续行最多可以有19行。所以，一个语句包括起始行在内可以写20行。但FORTRAN77子集规定最多可以有9个继续行。

3. 语句标号 语句标号（简称标号）应写在标号区（1~5列之间）。标号是一个无符号的整数，它的数值范围是1~99999之间，否则是非法的。

在同一个程序单位中，标号区内不能有相同的标号，继续行的前面也不能有标号。

语句标号只反映语句间的引用和被引用关系并不反映语句的先后次序。

4. 注解区 73~80列为注解区（也称为顺序号区）。在这个区域内，可以为每一个语句行写上编号，或对一些重要语句作点说明。和注释行一样，计算机在编译时并不理会这些内容。

5. 一个程序（主程序或子程序）的最后一个语句必须是END语句。它表示一个程序单位的结束。这样的程序单位又称为程序块。对图1-1的例子来说，这个程序块是一个主程序块。

应该指出，上述的书写（编排）规则，不仅是输入计算机的格式要求，也应是编写程序时的格式要求。不论是做作业还是编写实际的FORTRAN程序，都应遵循上述规则，从而养成良好的习惯。

二、FORTRAN程序的上机操作过程

FORTRAN程序在计算机上的操作过程大致可分为三步：

1. 调用编辑程序输入和修改我们的源程序。

2. 调用编译程序检查源程序的语法规则；若没有错误，则把它翻译成由机器指令组成的目标程序（又称浮动模块）。

3. 调用连接程序把浮动模块连接装配成具有绝对地址的可执行程序。需要时，可命令计算机执行这个可执行程序。

上述三个系统程序——编辑程序，编译程序和连接程序是由计算机系统提供的，它们由一个更大的系统程序——操作系统管理。

由于各种计算机在操作过程中所使用的操作命令各不相同，因此，具体的操作过程就不在这里介绍了。读者上机时应查阅有关的上机手册。

§1.3 常数和变量

一、FORTRAN77的字符集

FORTRAN77的字符集由49个字符组成。它们是：

26个字母，A~Z；

10个数字，0~9；

13个专用字符，+，-，*，/，=，（，），，，.，'，:，\$ 和空格符（我们用□表示）。

这13个专用字符中，有些符号的意义是明确的，有些字符有多种用途，在用到它们时再作说明。FORTRAN77子集中没有:和\$这两个专用字符。

二、常数

在FORTRAN语言中共有六种常数：整常数，实常数，双精度常数，复常数，逻辑常数和字符常数。前四种统称为算术常数。在这一节，我们先介绍整型、实型和字符型常数。逻辑型数据在第二章介绍，而双精度和复型数据将在第十一章介绍。

1. 整、实型常数

整常数——它的表示形式和数学中的整数相类似，就是一个不带小数部分的整数。如：123，-4567，+8000（+号可以省略）。

实常数——是一个带有小数或（和）指数的实数。它有二种基本的表示形式：

a. 只带有小数部分，它称为基本实常数。如：12.34，-5.678，.123，-5.。小数点前面或后面无效的0以及数前的加号“+”都可以省略。

b. 整常数或基本实常数后跟一个实指数。所谓实指数是指：字母E后跟任选带符号的整常数。如：+0.428E+2，-369E6，1.0E-2。实指数代表10的方幂。所以这三个实常数的值是42.8，-369000000，0.01。

单纯的实指数不是一个实常数，E的前面必须有一个整常数或基本实常数，E后跟的整常数也不能省略，但可以是0。

2. 字符常数

字符常数的形式是用一对撇号（也叫单引号）括起的一串字符。在一对撇号中至少要有一个字符（可以是空格符）。撇号只起分界作用而不是常数的一部分。例如

'FORTRAN-77'

'Character constant'

'1234 + 567'

如果这串字符中要用撇号，可用二个相邻的其间不插入任何字符的撇号来表示。如

'Didn't'

即使撇号内的字符都是数字，如'12345'，也不是一个整型数而是一个字符常数。

括在撇号内的字符串就是这个字符常数的值，而其中字符的