

高等学校教材

FORTRAN 语言程序设计导引

李续武 编

西北工业大学出版社

24
12

高等学校教材

FORTRAN 语言 程序设计导引

李续武 编

西北工业大学出版社

1990年5月 西安

内 容 简 介

本书是编者根据多年来讲授 FORTRAN 语言课程的教学实践编写而成的。全书共分七章,前六章系统地介绍了 FORTRAN 语言及其程序设计的基本内容。为了适应计算机语言发展的新水平,第七章扼要地介绍了有关 FORTRAN 77 的内容,并着重指明了 FORTRAN 77 和 FORTRAN IV 的主要差异点。每章还附有习题,可用来巩固和深化所学内容。

本书内容简明扼要,通俗易懂,深入浅出,层次分明,便于自学,可作为大专院校非计算机专业本科生、大专生及各类成人大学学生学习 FORTRAN 语言的教材,也可供各类中等专业学校师生及生产、科研部门为工程技术人员举办培训班使用。

高等学校教材

FORTRAN 语言程序设计导引

编 者 李续武

责任编辑 柴文强

责任校对 钱伟峰

*

西北工业大学出版社出版

(西安市友谊西路127号)

陕西省新华书店发行

西北工业大学出版社印刷厂印装

ISBN 7-5612-0263-6/TP·43(课)

*

开本 850×1168 毫米 1/32 9 印张 227 千字
1990 年 5 月第 1 版 1990 年 5 月第 1 次印刷

印数 1—1400 册 定价: 2.32 元

前 言

目前国际上使用的程序设计语言的种类很多，各有它们的适用范围。

FORTRAN 语言是 FORMula TRANslation 的缩写，意为“公式翻译”语言，是科学技术方面的重要语言，也是当前广泛应用的语言之一。它首先开始于美国，由美国国际商业通用机器公司（IBM）设计并提出的。随后，发展成为 FORTRAN I 和 FORTRAN V。1972 年，国际标准化协会（ISO）又公布了完全、中间和基本的三级 FORTRAN 标准文本，作为国际标准。完全的 FORTRAN 相当于 FORTRAN V，基本的 FORTRAN 相当于 FORTRAN I。在 FORTRAN 语言本身发展的过程中，它逐步改进了一些不足之处，并吸收了其它程序设计语言的一些优点，从而使它日臻完善。尤其是 1978 年，美国国家标准化协会（ANSI）颁布了标准 FORTRAN 新文本 ANSI-X3.9-1978（通常称之为 FORTRAN 77），它与 FORTRAN V 是向上兼容的，增加了适于结构程序设计的语句，扩充了字符处理的功能，引进了表控输入/输出语句，增强了文件处理的能力。一经宣布后，很快被国际标准化组织（ISO）采用作为 FORTRAN 文本，并为包括我国在内的许多国家所接受。我国国家标准总局 1982 年颁发 GB 3057-82（即 FORTRAN 77），1983 年正式实行。尽管如此，目前在国际上最广泛流行的仍然是以 FORTRAN V 为主，在具体的计算机上实现时，又总是在 FORTRAN V 的基础上向 FORTRAN 77 的方向作某些增加和扩充。本书为了减少篇幅以介绍 FORTRAN V 为主，同时为了适应计算机软件发展的新水平，在最后一章扼要地介绍一下 FORTRAN 77 的内容，目的在

59-1705

于力求简单明了的讲清有关 FORTRAN 语言的全貌。

本书是编者根据多年来讲授“FORTRAN 语言”课程的实践编写而成的，共分七章，系统地介绍了 FORTRAN 语言及其程序设计的基本内容。叙述上力求简捷严谨，层次分明，有机地结合例题，加深对内容的理解，便于自学。本书可作为非计算机专业的大专院校学生的教材，也可以作为电大、夜大、函大等各类成人大学的教材，又可以作为各类中等专业学校师生及生产、科研部门为工程技术人员举办培训班使用。讲授以 40~52 学时为宜。

本书由西北工业大学计算机科学与工程系主任康继昌教授审阅。在编写和出版的过程中，得到了西北工业大学电子工程系、出版社以及印刷部门许多同志的关心和帮助。在此，向他们表示衷心的感谢。

由于编者的水平所限，书中如有错误和不足之处，恳请读者批评指正。

编 者

1989 年 7 月

目 录

第一章 计算机及程序的概述..... 1

§ 1.1 计算机的硬件和软件..... 1

§ 1.2 对程序的初步认识..... 3

习题一..... 8

第二章 FORTRAN 语言的基本成分..... 9

§ 2.1 FORTRAN 语言的基本字符..... 9

§ 2.2 常数.....12

§ 2.3 语句标号和标识符.....15

§ 2.4 变量及类型说明.....17

§ 2.5 数组和数组元素.....20

§ 2.6 函数.....22

§ 2.7 表达式.....26

习题二.....31

第三章 FORTRAN 语言的基本语句（上）.....34

§ 3.1 赋值语句.....34

§ 3.2 输入语句.....40

§ 3.3 输出语句.....44

§ 3.4 格式语句.....49

§ 3.5 输入 / 输出表.....63

§ 3.6 停语句和暂停语句.....72

习题三.....74

第四章 FORTRAN 语言的基本语句 (下)78

§ 4.1 转向语句.....	78
§ 4.2 条件语句.....	82
§ 4.3 循环语句.....	89
§ 4.4 输入/输出部分的补充.....	107
§ 4.5 数据初值语句.....	123
习题四.....	127

第五章 FORTRAN 语言的过程 130

§ 5.1 FORTRAN 的过程.....	130
§ 5.2 语句函数.....	131
§ 5.3 函数子程序.....	136
§ 5.4 子例程子程序.....	141
§ 5.5 外部语句.....	157
习题五.....	160

第六章 数据联系语句与数据块子程序 164

§ 6.1 等价语句.....	164
§ 6.2 公用语句.....	168
§ 6.3 等价语句和公用语句的联用.....	173
§ 6.4 数据块子程序.....	175
§ 6.5 FORTRAN 语言的小结.....	176
习题六.....	186

第七章 FORTRAN 77 内容的简介 188

§ 7.1 FORTRAN 77 的基本概念与重要的 扩充.....	188
---------------------------------------	-----

§ 7.2	有格式的输入输出语句·····	195
§ 7.3	控制语句·····	200
§ 7.4	数组、数据类型和数据联系·····	208
§ 7.5	函数与子程序·····	218
§ 7.6	文件·····	223
§ 7.7	格式控制编辑符·····	233
习题七	·····	236
附 录	·····	239
附录 1	FORTRAN 77全集的内部函数表·····	239
附录 2	FORTRAN 77全 集、子 集与 FORTRAN W 语 句 对照表·····	243
附录 3	FORTRAN 77全集 INQUIRE 语句的 询问说明符表·····	245
附录 4	IBM-PC 微型计算机 FORTRAN 语言 上机操作简介·····	246
附录 5	IBM-FORTRAN 错误代码及信息对照 一览表·····	263
参考文献	·····	280

第一章 计算机及程序的概述

计算机的科技水平、生产规模和应用程度，已经成为衡量一个国家现代化水平的显著标志。

§ 1.1 计算机的硬件和软件

我们知道，计算机系统是由计算机硬件和软件两部分构成的。

一、计算机的硬件

硬件是指计算机装置本身，它分为主机和辅助设备。主机包括控制器、运算器和存贮器三个部分，其中运算器与控制器又称为中央处理机（CPU）。控制器是计算机的指挥控制中心，其功能是向机器的各个部分发出控制信号，来使整个机器自动地、高效率地、协调地进行工作。运算器是在控制器的控制下直接完成各种算术运算、逻辑运算和其它操作的部件。存贮器是用来存贮计算步骤、原始数据，中间数据和最后结果的部件。辅助设备包括输入设备（例如，键盘，卡片读入机和纸带输入机等）、输出设备（例如，终端显示屏幕，宽行打印机等）和外存贮器（例如，磁盘，磁带和磁鼓等）。计算机硬件的组成方块图如图 1-1 所示。

二、计算机的软件

软件是计算机按着人的意志所规定的步骤，来自动地，协调

地进行运算、处理和控制的程序部分的总称。计算机装置本身能看得见，摸得到，是硬的，称之“硬件”，计算机的程序部分，相当于算盘的口诀和运算规则，“二一添作五，三一三十一”等是看不见，摸不到的，相对硬是软东西，称之“软件”。研究计算机装置本身固然重要，但是，为了要充分地开展计算机的效能，来自动地，快速地和高效率地进行计算，不研究计算机的软件是不行的。硬件好比人体本身，软件好比是人所掌握的知识，它反映了计算机的“文化水平”。人固然重要，人的知识对科学技术的发展和四化建设更为重要。在某种程度上说，工程技术人员更关心软件的发展。软件是一种无形的财富，它凝聚了人们使用电子计算机的大量经验和技巧。硬件只有在软件的调动指挥下，才能获得生命力。硬件加上软件才构成了使人类的处理能力大幅度提高的计算机系统。

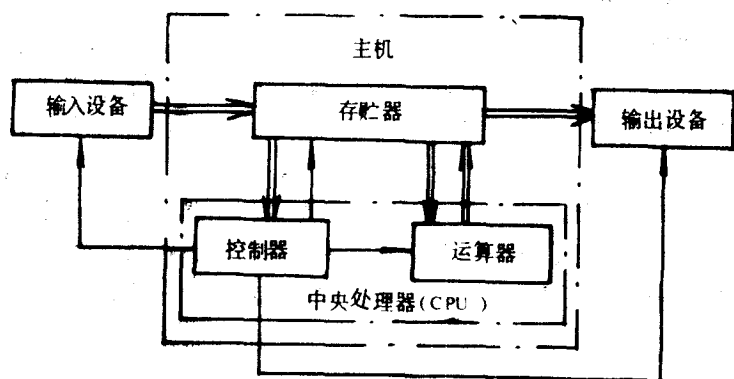


图 1-1 计算机硬件的结构框图

计算机的软件大致可分两个方面：

1. 系统程序：由厂家提供，它是用来提高计算机的使用效率、发挥和扩大计算机的功能和用途、简化程序设计与使用方法的一套程序。它包括三部分：

(1)语言程序(编译系统):将用“语言”编写的程序(源程序)翻译成计算机能认识的目的程序,相当于给机器配上了“翻译员”。

(2)管理程序(操作系统):使计算机系统各部分协调动作,从而发挥计算机功能,提高计算机工作效率的程序,相当于计算机管理自己的“管理员”。

(3)服务程序:为了方便用户,计算机所提供的服务程序(如诊断,排错,编辑等),相当于“服务员”。

2. 应用程序:用户提供,为使用计算机解决实际问题而编制的一些程序。

§ 1.2 对程序的初步认识

上机计算的整个过程是在控制器的指挥下进行的,而控制器的“思维”,却是按人们事先规定好的操作顺序和计算内容来完成的。这种事先规定好的操作顺序和计算内容,实际上是一组完整的指令,称为程序。编制程序的过程,就称为程序设计。

我们知道,计算机只认识二进制数和按一定的规则由0及1所组成的一条条指令,这些指令控制计算机进行相应的动作。这种二进制的指令是面向计算机的,它能为计算机直接接受,因此称之为机器指令,而机器指令的集合就是机器语言。用机器语言编写出的程序称之为手编程序。机器语言又称为低级语言。

用机器语言编程序工作量大,十分繁杂枯燥,直观性差,难编难记难读难懂,且受机器限制通用性差等,对非计算机专业人员学习和使用计算机将造成很大的困难。

针对这种情况,人们又采用助记符,创造了汇编语言。用助记符来代替二进制数字代码,这种符号语言程序不能在机器上直接执行,还得把它变成机器语言程序才能计算,也就是需要专门

的编译程序，把汇编语言程序逐条地变为机器语言程序。可见，采用符号的程序，在一定的程度上简化了编写程序的工作，但它和手编程序差不多，仍较烦琐。计算机的运算速度快与编写程序效率低之间的矛盾仍未解决。

事物总是不断发展的，50年代中期出现了高级语言FORTRAN，随后ALGOL，COBOL，BASIC，PL/1等其它许多高级语言也相继出现。高级语言的出现是计算机发展史上的一大进步，它既接近于数学语言，又吸收了人们所习惯用的语言，易于掌握，便于使用。

用高级语言编写的程序，称为源程序。源程序并不能直接为计算机所接受，还需要一种起翻译作用的编译程序，来将源程序变成计算机所能接受的机器语言程序——目的程序，也称目标程序。有了高级语言之后，人们可以不必学习机器指令，也不需要深入地懂得计算机的原理和内部结构，便可方便地利用计算机这一先进的工具进行运算了，这就为计算机的推广和应用扫除了极大的障碍。

下面通过引例，对FORTRAN语言的源程序作个初步认识。

【例1-2-1】已知三角形 $\triangle ABC$ 的边长分别为 $a=45.7\text{mm}$ ， $b=79.4\text{mm}$ ， $c=105.3\text{mm}$ ，试编写一个计算三角形 $\triangle ABC$ 面积的FORTRAN语言的程序。

由初等数学可知，已知三角形三边之长，求其面积 A ，用海伦公式：

$$A = \sqrt{s(s-a)(s-b)(s-c)}$$

其中

$$s = \frac{a+b+c}{2}$$

本例的FORTRAN语言的源程序如下：

C EXAMPLE1

```
      READ(1,1)A,B,C
1     FORMAT(3F6.3)
      S=(A+B+C)/2.0
      AREA=SQRT(S*(S-A)*(S-B)*(S-C))
      WRITE(6,2)A,B,C,AREA
2     FORMAT(1X,2HA=,F6.3,5X,2HB=,F6.3,
*        5X,2HC=,F7.3,5X,5HAREA=,F9.3)
      STOP
      END
```

04570 0 0 7 9 4 0 0 1 0 5 3 0 0

第一行是注释行，起注释说明的作用，在其第一列上写有“C”；第二、三行是提供数据的输入部分，它是由读语句(READ)和与其相应的格式语句(FORMAT)来实现的；第四、五行是进行赋值的运算部分；第六行到第八行是输出部分，它是由写语句(WRITE)和与其相应的格式语句来实现的，把三边及面积的值按规定的格式打印输出，其中第八行的第六列有续行标志*，表示第七、八行为同一个语句；第九行是停语句STOP，表示程序运行停止；第十行是结束行标志END，表示一个程序段的结束。

【例1-2-2】 已知四个三角形的每个边分别为 (a,b,c) ， $(2a,b,c)$ ， $(a,b,c/2)$ ， $(4,5,6)$ ，求它们的面积之和。

本例的FORTRAN语言的源程序在第6页上。

整个程序是由主程序和子程序组成的，子程序是函数子程序，它是根据海伦公式已知三边的长求三角形面积值的，该子程序从函数子程序定义语句开始，到END结束行为止。其中，第二行是类型说明语句，第五行是返回语句RETURN，表示执行完子程序后返回主程序。主程序的第四行到第七行是同一个语

句，分别写在四行上，其中四次调用函数子程序 AREA。

1. 主程序

C EXA	MPLE2	
		READ(1,5)A,B,C
5		FORMAT(3F10.5)
		SUM=AREA(A,B,C)
	*	+ AREA(2.*A,B,C)
	*	+ AREA(A,B,C/2.)
	*	+ AREA(4,5,6)
		WRITE(6,10)SUM
10		FORMAT(1X,4HSUM=,F15.5)
		STOP
		END
00045	7	0 0 0 0 0 0 0 7 9 4 0 0 0 0 0 0 1 0 5 3 0 0 0 0

2. 子程序

	FUNCTION AREA(LA,LB,LC)
	REAL LA,LB,LC
	S=(LA+LB+LC)/2.0
	AREA=SQRT(S*(S-LA)*(S-LB)*(S-LC))
	RETURN
	END

从上面的两个引例，可以得到如下的印象：

(1) FORTRAN 语言程序的特点是分段结构。可以概括成 16 个字：程序分段，段又分行，行按格写，顺序执行。这就是说一个 FORTRAN 的源程序可以由一个主程序段或者一个主程序段再加上若干个子程序段组成，每个程序段都有一个结束行标志 END，在一个程序段里，每个语句通常占一行，一行写不下可分行来写，但必须在第六列上加续行标志，每个语句的字符按

格去写，每个字符占一格，程序的执行一般按语句先后的顺序去执行。

(2)每段程序一般由以下四个部分组成：

- a. 提供数据部分
- b. 运算部分
- c. 输出部分
- d. 说明部分

(3)程序可写在程序纸上，行的书写分为三个区域：

a. 标号区 第1列到第5列用来写语句标号，标号最多为5位数字，它的范围是1到99999之间无正负号的整数。数字前后和数字中间的空格均不起作用。

b. 续行区 第6列为续行标志区，除了零和空白以外的字符都可以作为续行标志，例如在例1-2-1和例1-2-1中的*号，就是续行标志。续行的数目，有的机器允许最多9行，也有的机器允许最多19行。

c. 语句区 从第7列到第72列为语句区，填写语句可以从第7列以后的任意列开始写，一格内只能写一个字符（空白也是字符）。

顺便指出：第73列到第80列为程序员的备注区，不用键盘送入机器内，对运算不产生任何影响。输入数据可以从第1列开始一直到80列。

(4)语句的类型分两大类：

a. 可执行语句 在目的程序中产生相应的指令，从而使计算机产生一定动作的语句，称为可执行语句，例如，赋值语句、读语句、写语句、停语句等。

b. 非执行语句 在目的程序中不产生任何指令，只在编译阶段对计算机提供一定的编辑信息的语句，称为非执行语句，例如，格式语句、说明语句等。

(5) 程序的元素 有拉丁字母 A~Z, 阿拉伯数字 0~9 以及专用字符 = + - * / (), . 等等。

习 题 一

1. 什么是计算机的硬件及软件? 它们分别都包括哪些部分?
2. FORTRAN 语言的程序纸每行共多少列? 分为几个区? 语句部分写在哪些列? 续行标志写在哪一列上?
3. FORTRAN 语言程序的特点是什么? 程序段的最后一行应写什么? 它是否是语句? 它能否带有标号?
4. 每段程序一般主要由哪几个部分组成? 其语句的类型分几类?

第二章 FORTRAN 语言的基本成分

§ 2.1 FORTRAN 语言的基本字符

每一种语言都有它自己一套特定的符号, FORTRAN IV 规定它的基本字符共47个, FORTRAN 77规定它的基本字符为49个。

1. 拉丁字母: A~Z, 26个
2. 阿拉伯数字: 0~9, 10个
3. 专用字符: FORTRAN IV 共 11 个, 它们是: $\square$ (空白), = (赋值号), + (加号或正号), - (减号或负号), * (乘号或星号, ** 为乘幂号), / (除号或斜杠), , (逗号), . (小数点或限制符), ((左括号),) (右括号), \$ (美元号或币符)。

FORTRAN 77 的基本字符, 除以上47个字符外, 还增加了两个字符: ' (撇号) 和 : (冒号)。

有的机器最多可扩充到64个字符, 例如, 还有 & (和号), % (百分号), | (竖杠), ? (问号), # (井字号), ! (惊叹号) 等等。使用时, 要注意参看具体机器手册。

源程序可以通过键盘敲入其字符, 也可以用卡片通过卡片读入机输入。当用卡片输入源程序时, 每张卡片共分12行, (即 0~9 行、11和 12 行) 和 80 列 (图 2-1所示), 穿一张卡片对应源程序的一行, 卡片上每一列对应一个字符, 其编码表如表 2-1-1。