

FORTRAN

结 构

程 序 设 计

史济民 编著

四川科学技术出版社

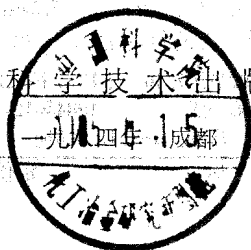
75.87221
176

FORTRAN 结构程序设计

史 济 民 编著

四川科学技术出版社

一九八四年·15



内 容 简 介

本书系统地介绍了国内外广泛应用的计算机FORTRAN语言的两种最新文本——FORTRAN77与标准FORTRAN (亦称FORTRAN66), 并结合讲解语言的内容, 循序渐进地讲解了结构程序设计的指导思想和主要方法, 是一本把学习语言和学习程序设计方法融为一体的新型课本。全书取材新颖, 注重应用, 对算法、文卷、程序设计步骤等各设专章介绍。例题较多, 利于实用, 启发思维。可作为大学学生、工程技术人员与管理干部学习FORTRAN语言及程序设计方法的教材, 也可供具有高中以上文化程度的人员作为自学计算机软件的书本。

FORTRAN结 构 程 序 设 计

四川科学技术出版社 (成都盐道街三号)

四川省新华书店发行 七二三四工厂印刷

开本787×1092毫米1/16印张16.5 插页2 字数398千

1984年12月第一版 1984年12月第一次印刷

印数: 1—31, 000册

书号15298·65

定价: 2.85元

前 言

随着新技术革命的到来,广大读者迫切要求学习计算机基础知识。对大多数人来说,高级语言常是他们学习计算机知识的入门课。因此,向读者提供适用的书本,为他们学会编程序创造有利条件,已成为推广计算机应用的重要环节。

一本好教材,应该兼顾内容的先进性、实用性和趣味性。在本书的编写中,作者力求按这三个要求选材,在坚持先进、实用的前提下达到生动有趣。

内容新颖

FORTRAN是当代流行最广的一种语言,而结构程序设计则是新的程序设计方法,是软件工程的重要基础。本书的最大特色,就是以结构程序设计思想指导FORTRAN语言的教学,把学语言同程序设计方法有机地结合起来,使读者既学了语言,又懂得了新的程序设计方法。

FORTRAN现有两种文本:一是标准FORTRAN(亦称FORTRAN66),一是FORTRAN77。由于后者对标准FORTRAN作了多方面的改进,其应用正迅速普及。本书是国内第一本系统讲述FORTRAN77的教材,亦可兼作标准FORTRAN的课本。

注重应用

鉴于计算机在非数值计算方面的大量应用,本书除讲解科学计算的例题外,还介绍了简易制图、事务管理、文字处理乃至日常生活应用等各种实例。全书有例题190余个,广泛多样,便于读者举一反三,适应多种应用的需要。

算法设计在程序设计中占有重要的地位。本书结合大量例题,较系统地介绍了一些常用的算法,并在第九章集中讨论了算法的效率、描述方法与计算机算法的特点,对初学的读者尤有裨益。

随着微型计算机普遍使用磁盘,文卷(file)的用途越来越广。以往的语言书很少讲述文卷,已不能满足读者的需要。本书第十五章专门讨论了FORTRAN的文卷语句,介绍了应用实例。

启发思维

学语言是有趣的智力活动。但有些书本孤立地介绍语法的限制,把学语言变成背语法,把本该兴趣横生的思维变成索然无味的死记硬背。本书把语法规则尽量与计算机基础知识联系起来讲解,使读者不仅知其然,也知其所以然,不必死记教条。

本书还适当选用有趣和带启发性的例题,例如百鸡问题、民意测验、计算机处理信件、处理密码等。既有利于开阅读者思路,启发思维,又能激发读者学习和使用计算机的兴趣。

适用面广

全书共15章,分3个单元。第一单元含第一至四章,讲述程序设计的初步知识。第二单元含第五至九章,讨论程序的基本结构、数据结构、常用算法与程序设计技巧。第十章起为第三单元,包括辅程序、文卷、字符信息处理,以及结构程序设计的步骤与方法等。循序渐进,由浅入深。有部分章节前打有*号,初学者可酌情不学或少学。

11
13/5

鉴于标准FORTRAN在国内有大量用户，本书前十章着重介绍标准FORTRAN，同时讲解FORTRAN77的扩充部分。第十一章以后FORTRAN77的比重逐渐增大，并在附录一至四中列出两种文本的主要差异，以供比较。只使用标准FORTRAN的读者可以跳过FORTRAN77的扩充部分，并不影响其余内容的系统性。

在结构程序设计与语言知识的安排上，本书采取了“相互渗透、有机结合”的方法。随着语言的进程，自然地向读者介绍有关的结构程序设计知识，使两部分内容如水乳交融、浑然一体，互为补充，相得益彰。

由此可见，本书既是一本FORTRAN77与标准FORTRAN的语言书，也是一本结构程序设计与常用算法的入门书，一本通过各种实际程序介绍计算机应用的普及课本。

几年来，本书曾多次用作大学本科生、非计算机专业研究生、大学教师、以及工厂科技人员、企事业单位管理干部训练班的教材，课堂学时从40至60学时不等。从我们的初步实践看，它（1）能使更快地掌握程序设计规律；（2）能使从开始就养成良好的程序设计风格；（3）便于学生进一步学习软件工程知识；（4）能节省计算机专业学生的学习时间（与分开学习语言与结构程序设计两门课相比），和（5）能使非计算机专业学生在增添很少语言学时的条件下学会程序设计方法的新知识。有些具有高中文化程度的读者用作自学课本，效果也很好。

感谢与希望

把结构程序设计融合到语言教学中，是一种新尝试。使作者感到高兴的是，在本书编写和试用的过程中，天津大学许镇宇、清华大学谭浩强、西南交通大学潘启敬、成都电讯工程学院张志浩、江明德等前辈和同行先后在一些会议上热情支持这一改革，给作者很大的鼓舞。在校内，何明儒、伍良富、向孟光等许多同志对书稿提过宝贵意见，陈纪同志协助编写了各章习题，验证了例题程序。借此机会，作者向所有关心、帮助过本书的同志表示衷心的感谢！但是，由于这是一个新的工作，加上作者知识有限，不当与错误在所难免，作者诚恳希望读者批评指正。

史 济 民

1984年6月于成都科技大学

目 录

第一单元

第一章 程序和语言	1
1.1 计算机程序	1
1.2 程序设计语言	1
1.3 计算机系统的组成	2
1.4 FORTRAN 程序	4
1.5 计算机解题步骤	6
1.6 什么是结构程序设计	8
习 题	9
第二章 数据的表示	10
2.1 信息与数据	10
2.2 字符编码	10
2.3 数值数据的机器表示	11
2.4 FORTRAN的数据类型	12
2.5 整型与实型常数	12
2.6 整型与实型变量	14
2.7 字符型数据简介	15
习 题	16
第三章 基本输入输出语句	17
3.1 概述	17
3.2 格式读/写语句	17
3.3 格式语句	18
3.4 I、F、E型字段描述符	19
3.5 X和H型字段描述符	23
3.6 描述符重复数和描述符组	26
*3.7 格式读/写语句的执行过程	26
3.8 输入输出的纵向格式	27
3.9 输入输出语句组成的简单程序	30
习 题	32
第四章 算术计算	35
4.1 五种基本的算术运算	35
4.2 用算术表达式计算	35
4.3 算术赋值语句	38

4.4 函数	41
4.5 简单的数值计算程序	43
习 题	45

第二单元

第五章 基本的程序结构	47
5.1 程序流程图	47
5.2 三类基本的程序结构	48
5.3 结构化程序	50
习 题	50
第六章 选择程序设计	51
6.1 转移语句	51
6.2 无条件转移语句	51
6.3 逻辑条件语句	53
6.4 使用逻辑条件语句的多重选择结构	59
6.5 分块条件语句	62
6.6 分情形结构与计算转语句	67
*6.7 其它转移语句	70
习 题	71
第七章 循环程序设计	74
7.1 用条件转移语句实现循环	74
7.2 循环结构的类型	78
7.3 循环语句及计数型循环的自动控制	79
7.4 输入型循环与条件型循环	87
7.5 多重循环	91
7.6 FORTRAN控制结构小结	96
习 题	97
第八章 数组	99
8.1 数组与数组元素	99
8.2 数组的说明	100
8.3 数组的输入与输出	102

8.4	数组运算中的常见操作	108
8.5	应用数组解题举例	110
8.6	使用与不使用数组的比较	120
	习 题	122
第九章	算法	125
9.1	算法的重要性	125
*9.2	算法的描述	126
*9.3	算法的效率	128
9.4	计算机算法的种类和特点	131
9.5	程序设计技巧	132
	习 题	133

第三单元

第十章	逻辑型、双精度型与复型	
	数据	134
10.1	逻辑型数据	134
10.2	逻辑运算	135
10.3	逻辑变量的应用	137
10.4	双精度型数据	144
10.5	复型数据	146
	习 题	148
第十一章	输入/输出续介	151
11.1	G 格式	151
*11.2	比例因子	152
11.3	格式读/写语句的简化	153
11.4	自由格式输入/输出	155
*11.5	FORTRAN77的其它输入/ 输出格式	157
	习 题	160
第十二章	字符信息处理	163
12.1	字符型数据	163
12.2	字符串的分割与合并	168
12.3	字符表达式与赋值语句	169
12.4	字符串的比较	171
12.5	字符处理函数	175
12.6	格式数组及其应用	177

	习 题	179
第十三章	辅程序	183
13.1	辅程序的功用和种类	183
13.2	函数辅程序	184
13.3	外部语句和内部语句	188
13.4	子程序辅程序	190
13.5	虚实结合	193
13.6	公用语句与公用结合	196
*13.7	等价语句	201
13.8	数据块辅程序和赋初值语句	203
13.9	通用辅程序举例	206
*13.10	用于辅程序的其它语句	212
	习 题	214
*第十四章	程序设计的步骤与方法	
		218
14.1	程序设计的一般步骤	218
14.2	模块化程序设计	221
14.3	逐步求精的设计方法	223
14.4	小结	228
	习 题	228
第十五章	文卷	232
15.1	文卷的种类和性质	232
*15.2	磁介质文卷的读/写过程	234
15.3	文卷读/写语句和无格式 读/写语句	235
15.4	文卷的打开与关闭	237
15.5	文卷定位语句	240
15.6	应用举例	242
	习 题	248
附录一	FORTRAN语句一览	251
附录二	FORTRAN程序段中的 语句排列顺序	253
附录三	FORTRAN的标准函数	254
附录四	FORTRAN66与FORTRAN77 的主要差异	258
附录五	上机实习	260

第一章 程 序 和 语 言

1.1 计 算 机 程 序

数字计算机是近代科学技术的一项重大成果。现代的计算机，其运算速度可达每秒几百万次，甚至几亿次。但是，不论一台计算机的速度有多么快，功能有多么强，它的一切行动都要听程序的指挥。任何计算机，都只能执行程序安排它做的事情，对程序“亦步亦趋”。离开了程序，最好的计算机也将“寸步难行”。

你想使用计算机完成某项任务（科学计算、数据处理、自动控制或其它）吗？那么，首先请按照任务的要求编写好相应的程序，然后把程序输入计算机，让计算机遵照执行。实际上，程序可看作由人事先规定的计算机的操作步骤。每一步骤的内容由指令或语句来描述。这些指令或语句，将告诉计算机“做什么”和“怎样做”。

1.2 程 序 设 计 语 言

编写程序的过程称为“程序设计”，而编写程序使用的语言则称为“程序设计语言”。

1. 机器语言 最早使用的程序设计语言叫“机器语言”。这类语言的特点是：

(1) 采用二进制代码，指令的操作码（如+，-，×，÷等）和操作数地址（注）均用二进制代码表示；

(2) 指令随机器而异，不同的计算机有不同的指令系统。

众所周知，计算机采用二进制，其逻辑电路也是以二进制为基础的。因此，这种用二进制代码表示的程序能够被计算机直接理解和执行，这是机器语言的最大优点。但这种语言冗长，难记，而且因机而异（称做“面向机器”），给编程序的人带来许多麻烦。

2. 汇编语言 于是，人们想出了用符号（称为助记符）来代替机器语言中二进制代码的方法，设计了“汇编语言”。用汇编语言书写的程序，其指令的操作码和操作数地址全都用符号表示。例如，求A、B两数之和C，其汇编语言程序可能包括以下指令：

	操作码	操作数地址	说 明
指令 1：	LDA	A	；取出A (LDA = Load Accumulator)
指令 2：	ADD	B	；加上B
指令 3：	STA	C	；存入C (STA = Store Accumulator)

采用“助记符”来表示操作码，大大方便了记忆，但是它们却不能直接被机器识别和理解。所以用汇编语言书写的源程序送入计算机后，先要由“汇编程序”将这些符号“代真”

注：一条指令通常包括两部分：操作码告诉计算机做什么操作，地址码告诉计算机到哪里去取或存操作数。

(即代入真实的二进制代码)，变成以机器语言形式表示的“目的程序”，然后才能由计算机执行。汇编程序还具有为数据分配存储空间等功能，使程序员的工作更加方便。

3. **高级语言** 机器语言和汇编语言都是面向机器，依赖于机器的。而高级语言则面向用户，基本上与所用的计算机无关。

自五十年代初高级语言问世以来，人们已经设计了几十乃至上百种高级语言。其中如FORTRAN, ALGOL60, BASIC, COBOL与PASCAL等，都是它们中间的著名代表。

高级语言的最大优点，是面向用户，形式上更接近于算术语言和英语语言。仍以A、B的和数C为例，采用高级语言时的语句可能是：

ALGOL 60:	C := A + B
BASIC:	LET C = A + B
FORTRAN:	C = A + B
COBOL:	ADD A TO B GIVING C

由此可见，与汇编语言相比，高级语言不仅易学、易用，而且写出的程序更加简练。由于它基本上独立于机器，即使对计算机知之不多的人，也能够编写出可用的甚至很好的程序来。当然高级语言程序也不能直接为计算机理解和执行，需要依靠编译程序来编译。其过程可表示为：

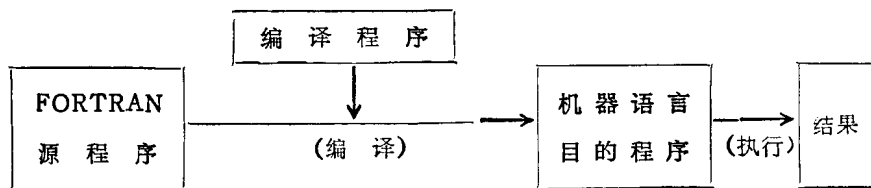


图1.1 源程序的编译与执行过程

有些高级语言如BASIC写的源程序，不象FORTRAN那样整个地被译成目的程序后再执行，而是由一个称为“解释程序”的程序，对源程序边解释边执行。

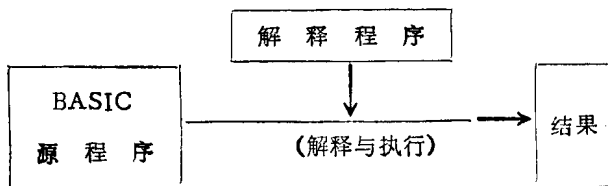


图1.2 源程序的解释与执行过程

无论编译程序或解释程序，都要事先送入计算机的内部存储器（简称内存）中，才能对源程序（此时也在内存中）进行编译或解释。由于解释方法不产生完整的目的程序，所以占用内存空间较小，但它耗费的机器时间远比用编译方法占用的机时为长。

1.3 计算机系统的组成

1. **基本组成** 现代的计算机系统，由硬件与软件组成。硬件是机器设备包括的总称，

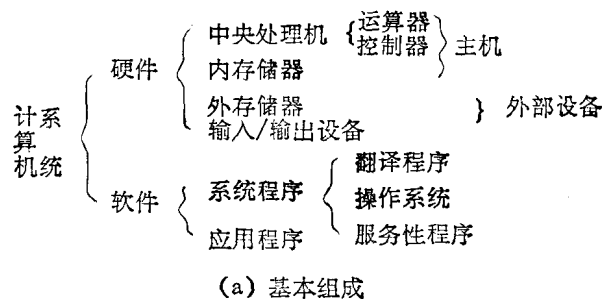
对数据进行处理中央处理机（CPU），存放程序和数据的存储器，以及通称为外部设备的输入设备和输出设备。中央处理机又包括运算器与控制器，它与内存储器一起构成计算机的主机。

软件是程序的总称。它又可分为两类。

（1）系统程序（系统软件）——常指与硬设备配套出售的基本软件。例如上述的各种翻译程序（汇编程序，编译程序与解释程序），管理整个系统的操作系统，和各种服务性的程序（如诊断程序，检查程序）等。

（2）应用程序（应用软件）——指计算机公司（工厂）为便于用户使用计算机而开发的程序，通常以软件包的形式出售。例如数学计算软件包，绘图软件包，企业管理软件包，等等。有些计算中心为本系统用户编写的专用程序库，也属于应用软件。

图1.3表示计算机系统的基本组成及硬件结构。



2. 存储器 是计算机存放程序和数据的场所，又分为内存（主存）和外存（辅存）。从程序设计的角度看，对存储器主要应了解以下的问题。

（1）字长——内存通常划分为单元。每一单元可存放的二进制位数称为字长。如CROMEMCO-CS2微型机字长为8位，而DJS-130机字长16位，MV/6000计算机字长32位等等。字长越长，计算机能表示的数的范围越大，数的精度也越高。字长也可以字节为表示单位，一个字节等于8位。如MV/6000字长32位，也可说字长4个字节。

（2）存储容量——字长×单元数=存储容量。假如把内存比作旅馆，则存储单元好比房间，存储地址好比房号，字长相当于每间房能容纳的客人数，如下表所示：

旅 馆	存 储 器
房 间	存储单元
房 号	存储地址
客人数/每间房	字 长

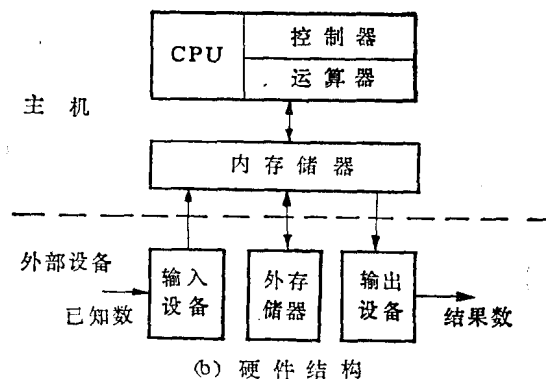


图1.3 计算机系统的基本组成与硬件结构

存储容量通常用KB或KW为表示单位,其中K表示 2^{10} ($=1024$), B表示字节, W表示字。例如CS—2微型机的内存容量为64KB (即 $64 \times 1024 \times 8 = 524288$ 二进制位), 130机的最大内存容量为32KW,也相当于64KB (因130机字长为2字节), MV/6000的内容量则达到2048KB或2MB (2兆字节)。显然存储容量越大则能够存储的信息越多,允许在内存中同时运行的程序就越多、越大。

(3)内存与外存——内存直接配合CPU工作,存取信息的速度很高,但由于容量一般较小,需要外存来支援。常用的外存有磁盘、磁带、磁鼓等。外存容量一般为内存的数十乃至数百倍,但存取速度要慢得多。

计算机运行时,内存中除了用户程序外,还要存放支持用户程序运行的系统程序 (见图1.4)。为了节省内存空间,通常只把当前运行的程序与数据调入内存,暂时不用的程序与数据一律放在外存。等用到它们的时候,再从外存调入。

(4)读出/写入时的特点——从存储器读出一个数时,读出单元的内容并不改变。但若将一数写入某存储单元,则该单元原存的内容即被破坏 (或冲掉),代之以新的内容。如图1.5所示, A, B为两个存储单元,原来分别存有 a、b 两数。现在要求 $A \rightarrow B$, 即把A的内容读出送B。执行这一操作之后, A的内容不变,仍保持为 a, B的原内容 b 被冲掉,也变成 a。这个特点在程序设计中十分有用,必须牢牢记住。

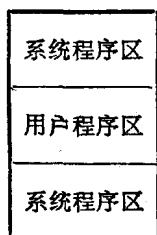


图1.4 内存空间分配示意图

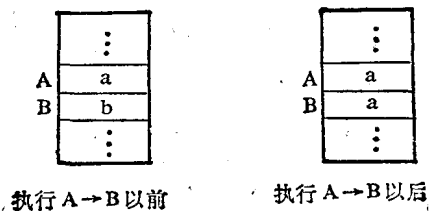


图1.5 存储器的读写特征

3. 输入/输出设备 (I/O设备)

(1)功能——用于人一机通信,是计算机与外部世界交换信息的接口和桥梁。

(2)常用的I/O设备:

输入设备有——纸带输入机,卡片输入机,键盘等。

输出设备有——字符显示器,行印机等。

有的设备兼有输入输出的功能,例如电传打字机,控制台打字机,终端等。

(3)设备逻辑号 当程序中用到某台外部设备进行输入输出时,要给该台设备指定一个逻辑号,使编译程序能够知道本次输入 (输出) 用的是什设备。不同的计算机可能配备不同的外部设备,逻辑号也随机器而异。使用前应查看随机器供应的语言手册,以免出错。

1.4 FORTRAN 程 序

FORTRAN来源于英文的Formula Translation,故亦称“公式翻译语言”。它是世界

上最早推广应用的高级语言，也是当前国际上最流行的高级语言之一。它广泛地用于数值计算，也可应用于事务管理和数据处理。

1. 文本 FORTRAN有过许多文本，目前流行的主要有两种。

(1) 标准FORTRAN 大致相当于FORTRAN W。它是美国国家标准协会 (ANSI) 于1966年公布的，故有时也称作FORTRAN 66。

(2) FORTRAN 77 系在标准FORTRAN的基础上扩充修改而成，比标准FORTRAN增添了许多新功能，1978年4月由美国ANSI正式公布。

本书将同时介绍这两种文本，并比较两者的差异。

2、一个简单的FORTRAN程序。

问题：已知圆的半径，求它们的面积与周长。

程序：

```
C FIND THE AREA AND CIRCUMFERENCE OF A CIRCLE
  READ (3, 100) R
  AREA = 3.1416 * R * R
  CIR = 2.0 * 3.1416 * R
  WRITE (3, 110) R, AREA, CIR
  STOP
100  FORMAT (F10.3)
110  FORMAT (1X, 3F10.3)
END
```

3、程序的结构与组成 一个FORTRAN程序由一个或多个程序段组成。当程序包括多个程序段时，其中必有一个为主程序段，其余为辅程序段。上例的程序只包括一个程序段。

程序 { 主程序段 (1个)
 辅程序段 (0个或多个)

每个程序段含有若干行。上述的程序有9行，现简述如下。

第1行：为注释行 (Comment line)。其作用是帮助阅读程序的人了解整个程序段或者其中某些语句的用途。它们不是程序的必要组成部分，删去后不会影响程序的执行。注释行均以字母C开头，编译程序在一行的开头遇到字母C，就知道该行是注释行，不对它进行编译。如果一行写不完，可在继续行中接着写，但继续行也要以C字母开头。

第2—6行：为执行语句，包括1条输入语句，2条计算语句，1条输出和1条停止语句。

第7—8行：为格式语句，句前的数字为语句号。语句号为100的格式语句规定了输入的格式，110号格式语句规定了输出的格式。

FORTRAN规定，语句号一律由数字组成，其范围为1—99999，最多不超过5个数字。值得指出的是：

(1) 只有被别的语句引用的语句需要语句号，其它语句不需要，

(2) 语句号的大小并不代表语句执行次序。

第9行：为结束行 (FORTRAN 66) 或结束语句 (FORTRAN 77)。它标志一个程序段的终结，所以总是写在程序段的最后一行。当一个程序含有多个程序段时，每一程序段的最

后一行都要写上END。END行不允许有标号。

4、程序书写格式 为了便于编译程序对源程序进行编译，也为了方便人们阅读程序，FORTRAN的源程序要求按一定格式书写。标准的程序纸每张可写若干行（行数不作统一规定），每行有80格。这80格分为3个区：

(1)标号区 占每行第1至5格，用于书写语句标号。没有标号的语句此区为空白。

(2)续行区 占第6格，用于填写继续行的标志。一条语句最多可占20行，即包括一个起始行和19个继续行。凡FORTRAN语言中的合法字符，除去“0”和空格符外，均可用作续行标志。编译时遇到续行标志，就把该行当作上一行的续行。

标号只能在始行填写，故续行的标号区必为空白。

(3)语句区 占7~72格，用以填写语句的内容。

第73~80格是留给程序员作注释与标注用的，不属于程序的组成部份。把源程序输入计算机时，不输入第73—80列的内容。（请注意，注释行虽然也不是程序的必要组成部份，但要随源程序一起输入计算机）。

需要说明的是，上述格式适用于书写源程序的所有语句行。对于注释行的始行与续行，第1格一定要写C。空一至数格后，填写注释内容，至72格为止。当用程序纸书写原始数据时，不受这些规定的限制，从1至80列均可记录数据。

1.5 计 算 机 解 题 步 骤

利用计算机解题，如果采用FORTRAN语句，其过程可分为3个阶段。

1、准备阶段

- (1)弄清题目要解决的任务，看能不能、值不值得用计算机来解。
- (2)按照任务的要求确定解题的算法。记下算法的步骤或画出它的流程图。

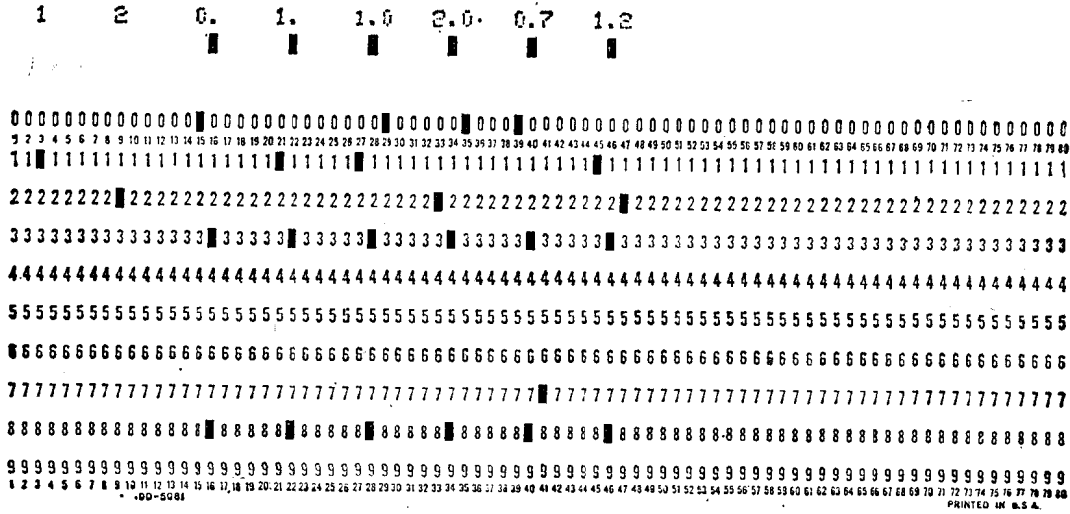


图1.6 FORTRAN卡片

(3)按选定的算法写好FORTRAN源程序，并写在程序纸上。

(4)如果源程序用穿孔纸带或穿孔卡片（通称FORTRAN卡片）输入，则先在穿孔机上把纸带或卡片按程序穿好孔。

穿孔纸带适用于配有纸带输入机的计算机。当源程序写好后，首先按约定的编码把程序穿在纸带上。每穿好一行，打一〈回车〉符，接着穿下一行。要注意每行不能超过72个字符，即纸带上两个〈回车〉之间不多于72排孔。

FORTRAN卡片常用于国外的大、中型计算机。记录程序的卡片，每卡片可记录源程序的一行（始行或续行），但行末不需要穿〈回车〉符。图1.6表示卡片的格式。每卡含80列，相当于一行中的80格，每列有12行，用于记录一个字符的编码。

2、编译阶段

(1)把编好的源程序通过纸带输入机（若为穿孔纸带）、卡片阅读机（若为穿孔卡片）或键盘等输入设备送入计算机的内存，并由先已存放在内存中的FORTRAN编译程序对源程序进行编译。

(2)编译过程中如发现源程序有语法错误，编译程序将通过输出设备显示或打印“出错信息”，告知用户出错的行号和错误性质。

(3)用户修改源程序，并把改好的程序再次送入计算机内存进行编译，若仍有错误，则继续修改，直至源程序不再有语法错误为止。

(4)编译程序将没有语法错误的源程序编译为机器能够理解的目的程序，放在内存存储器中，准备执行。

3、执行阶段

(1)用户发出启动运行的命令后，放在内存中的目的程序指令即被逐条地取到中央处理机。然后由控制器对每条指令进行分析，并发出指令所需的各种控制信息。

执行指令需要的各种数据信息，有的已经预存在内存的数据区中，有的由用户当时从键盘送入（内存）。程序运行时，这些数据将按照指令的要求陆续被调入运算器，进行运算或其它处理后把结果（中间结果或最终结果）送回内存。

(2)在执行中也可能发现错误，例如结果数超出计算机许可的范围，发生“溢出”，有些输入数据的类型与程序中说明的程序不一致，等等。这时，计算机将通过输出设备显示运行“出错信息”，告知用户改正。

(3)如果没有发生运行错误，则将运行结果从内存送至输出设备显示或打印。至此一次运行结束。

编译程序的功能

由上可见，用FORTRAN编写的源程序，只有在FORTRAN编译程序的支持下，才能在计算机上运行。没有配置FORTRAN编译程序的计算机，就不能执行用FORTRAN写的源程序。编译程序的基本功能为：

(1)把源程序翻译为机器语言程序，以便计算机识别和执行；

(2)为程序中用到的一切常量和变量分配存储单元；

(3)当程序在编译或执行过程中发生错误时，发出（显示或打印）“出错信息”，把出

错的位置和性质告知用户。

1.6 什么是结构程序设计

以上,我们简述了与程序设计直接有关的一部分软件、硬件知识。在结束本章的讨论之前,还要提一下贯穿于本书的一个重要内容——结构程序设计(structured programming)。

结构程序设计是近十几年发展起来的新的程序设计方法。与传统的程序设计方法(现在也称为非结构程序设计方法)相比,它们在程序设计的主要目标和方法上,都有许多重要的不同。

在早期,由于计算机的速度很低,存储量小,评判一个程序的优劣,总是把效率放在首位。为了提高效率,程序员们千方百计追求程序设计的“技巧”,以便从中减少一条语句或指令,或者省用一个存储单元。但越是技巧性高的程序,往往越难被别人读懂。随着硬件的改进,速度与容量千百倍地提高,程序的效率已不是主要的矛盾。另一方面,软件的规模却越来越大,验证、维护和修改程序的费用急剧上升。因此人们首先关心的问题,已不是提高程序的效率,而是怎样使程序容易被人读懂,容易验证它的正确性,以节约维修的费用。这就是这一新的程序设计方法——结构程序设计发生和发展的背景。

经过多年的探索和实践,人们终于发现,为了提高程序的易读性与可靠性,必须从改进程序的基本结构入手。显然,这些基本结构必须具有简单清晰的特点,既便于阅读,也便于书写,从而易于保证和验证程序的正确性。简要地说,结构程序设计是“按照一组能够提高程序易读性与易维护性的规则进行程序设计的方法”(注1)。“在保证程序正确可靠的前提下,它首先考虑程序的清晰,然后考虑程序的效率”(注2)。从六十年代末期到现在,结构程序设计已发展成为一套系统的科学方法,不仅得到国外软件工作者的广泛承认和欢迎,也在国内受到越来越多的重视。

本书将结合FORTRAN语言的内容,向读者介绍结构程序设计的基本思想,使读者不仅了解这一程序设计方法上的新发展,而且从一开始就养成良好的程序设计风格。

注1: Baker, F.T. Organizing for Structured Programming, Lectures Notes in Computer Science vol. 23, 1974.

注2: Hughes, J.K. and Michtom, J.I., A Structured Approach to Programming, Page 75.

习 题

- 1、什么是程序？简述程序的作用。
- 2、简述机器语言和高级语言的特点与主要区别。
- 3、何谓源程序和目的程序？它们的关系是什么？
- 4、计算机硬件有哪些部分组成？常用的输入、输出设备有哪些？
- 5、说明下列名词的区别：（1）字节与字长；（2）系统软件与应用软件，（3）CPU 和主机。
- 6、存储器读出和写入时有何特点？
- 7、试述FORTRAN源程序的结构与格式。
- 8、编译程序的主要功能是什么？
- 9、以FORTRAN语言为例，简述计算机解题的主要步骤。
- 10、什么是结构程序设计？它的目的和实现这些目的的主要途径是什么？

第二章 数据的表示

2.1 信息与数据

计算机是信息处理的自动机。用计算机解题,从实质上来说,就是用计算机进行信息处理。

信息在计算机中一律用数据表示。按照不同的性质,计算机中的数据可划分为以下类型:

1、数值数据(包括整型、实型); 2、非数值数据(包括字符型、逻辑型)。

数值数据系指可以进行算术运算的数据。例如3加5, 7乘8, 其中的3, 5, 7, 8都是数值数据。非数值数据则相反,它们不进行算术运算。例如“中山路60号”是一个地址,它是非数值数据。尽管其中也包含60这个数字,但它不需要运算,仍属非数值数据。

数值数据可分为整型与实型。非数值数据可分为字符型与逻辑型。除逻辑型留待第十章讨论外,其余的将在以下各节陆续讲述。

2.2 字符编码

1、FORTRAN字符集 任何语言都包括一定的字符。例如英语中的字母,日语中的假名等。各种高级语言也都有它们的字符集。标准FORTRAN的字符集包含47个字符。它们是:

英文字母字符26个: A、B、C、……X、Y、Z

数字字符10个: 0、1、2、……8、9

特殊字符11个: +、-、*、/、(、)、=、,、.、\$、空格

这47种字符都允许在标准FORTRAN源程序中出现。如果用了其它的字符,就被认为是非法的,编译程序将拒绝接受。

FORTRAN77的字符集,比标准FORTRAN多两个特殊字符(单引号'和冒号:),字符总数为49个。

2、ASCII码 在计算机存储器中,字符数据一律用二进制编码表示。实用的字符编码有许多种。其中在小型和微型计算机中流行最广的,要算ASCII码,即美国标准信息交换码(American Standard Code for Information Interchange)。

ASCII码是一种7位代码,每一种字符由7位二进制码表示。通常一个字节存储一个字符。字节的最高位用作奇偶校验位,其余7位存字符编码。

当FORTRAN源程序送入计算机后,即以ASCII码的形式存放在内存的程序区,以便于计算机识别。当把内存中的源程序输出打印时,ASCII码又转换为相应的字符,以便于人们阅读(见图2.1)。