

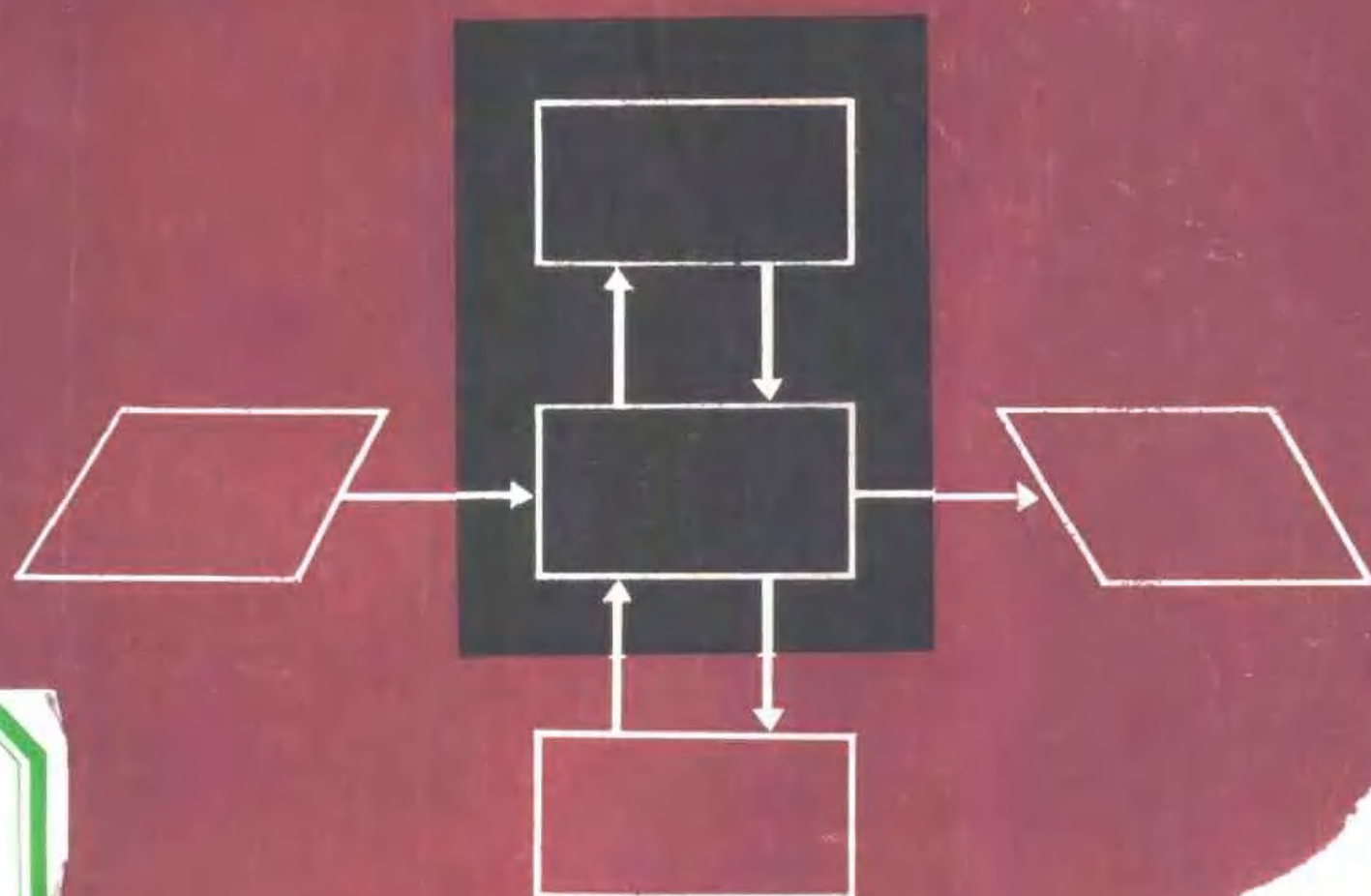
S.C.Loh等 编著

FORTRAN

IV

程序设计语言教材

孙嘉鸿 译 肖业伦 校



人民教育出版社

高等学校教学参考书

FORTRAN IV 程序设计语言教材

S. C. Loh 等 编著

孙嘉鸿 译 肖业伦 校

人民教育出版社

本书是香港中文大学教授乐秀章等编著《FORTRAN IV 程序设计语言教材》和《FORTRAN IV 程序设计语言教材习题解答》的合译本。它以简洁的语言浅显易懂的方式介绍了 FORTRAN 语言的主要内容。书中有丰富多样的例题和习题，并附有习题的全部解答。本书可供高等学校师生及有关科技人员参考，也可供初学程序设计的读者参考。

高等学校教学参考书
FORTRAN IV 程序设计语言教材

S. C. Loh 等编著
孙嘉鸿 译 肖业伦 校

人民教育出版社出版
四川省新华书店重庆发行所发行
重庆新华印刷厂印装

开本 787×1092 1/16 印张 13 插页 1 字数 150,000

1980年1月第1版 1981年5月第3次印刷

印数 23,501—34,500

书号 13012·0423 定价 1.00 元

译者序

《FORTRAN IV 程序设计语言教材》是供初学程序设计的读者使用的一本入门的教材。它以简洁的语言,丰富多样的例题和习题(附有全部答案),以及浅显易懂的方式介绍了 FORTRAN 语言的主要内容。本书不要求读者具备电子计算机和高等数学方面的预备知识,只要具有中学文化水平的人,就可以通过本书的学习初步掌握 FORTRAN 语言的要旨,并初步学会编制计算程序。

本书讲述的是在英国 ICL1900 系列计算机上使用的 FORTRAN 语言(简称 1900FORTRAN),它与标准的 FORTRAN 语言仅在极个别细节上有所差异,这些差别大多在原书中已经指出,其余的在译本中补充指出了。

在翻译和校对过程中,发现原书的一些笔误或者印刷错误,我们直接予以改正,不作解释。在一些我们认为不够清楚的地方,或者用注解的方式作了解释和补充,或者直接在正文中予以补正,但同时注明了哪些字是译者加进去的。

在名词术语方面,我们尽量采用 FORTRAN 语言的标准术语。对本书的习题解答,译者都校对过,有些题目还加了注解。据作者在习题解答的序言中说,这些习题的答案程序都是已经上计算机运行通过了的。

由于本书是一本 FORTRAN IV 的简明教材,只讲述最基本的内容,考虑到大学理工科学生学习的需要,译者补充了四个附录,简单地介绍了逻辑 IF 语句、可调数组、有名公共块和 EXTERNAL 语句。

进一步掌握程序设计的技巧,最好是通过大量上机算题和学习有关手册解决。

译者

1979 年国庆节于北京

序

自从电子计算机发明以来,程序设计就成为自动计算各种题目的必要条件。由于制定了许多种面向问题的程序语言,即使不是专家也有可能编好程序。在所有这些程序语言中,FORT-TRAN 是应用最广的一种,而且它特别适合于应用在数学和科学的题目上。

本书的主要目的是给高等院校学生提供一本关于FORTRAN IV 程序设计语言的现代教科书。我们也希望本书将会证明对数学、工程和自然科学领域的应用,以及对社会科学和管理科学领域的应用均是有用的。本书并不要求读者有计算机和程序设计方面的预备知识,只需学过初等代数就足够了。

本书中有很多例题与习题。因此本书很适于作为带有家庭作业的正规教程之用,也可用作自修读物。在每一章中引入一个新的概念之后,就用例题加以说明。例题中的程序通常与相应的流程图在一起提出来。这种平行的说明方法肯定会促进对计算过程的形式和结构的理解。在每一章的末尾(除第六章外)都给出很多习题。要督促学生做这些习题,因为事实证明,要学会并能灵活应用编程序技巧的唯一办法就是将新获得的知识付诸实践。这些习题将大大有助于牢固地掌握这门学科。这些习题的解答均刊印在《FORTRAN IV 程序设计语言教材习题解答》一书中^①。本书中的有些习题前面冠以星号“*”,它表示这些题目特别有用,并推荐给非理工科的学生。

我们曾经努力使本教材尽可能的准确和生动。FORTRAN IV 的参考手册是编辑本书素材的重要资料来源。我们诚挚地感谢计算机制造厂商们的帮助,他们提供了这些手册的最新版本。

最后我们希望本书对已经在商业和工业界以及政府机关工作的人们将有所帮助。别的部门,例如大学和中学的教师同样会发现本书对他们有用。虽然目前还没有做到,但是我们相信在不久的将来程序设计也会成为香港初等教育的正规课程。

香港中文大学计算中心主任

教授 S. C. Loh (乐秀章)

^① 在中译本中,这些习题解答直接附在本书末尾,不另成册。——译者注

目 录

第 一 章

绪论	1
计算机系统的基本部件	1
存储器	1
控制器	2
算术与逻辑运算器	2
输入设备	2
卡片	3
卡片读入机	3
纸带	4
纸带读入机	4
输出设备	4
行式打印机	4
高速外围设备	4
计算机的程序编制步骤	4
习题	5

第 二 章

FORTRAN 程序	6
算术赋值语句	6
输入-输出语句	6
控制语句	6
说明语句	6
子程序语句	6
FORTRAN 程序纸	6
语句标号区	6
语句区	6
标识区	8
基本字符	8
名字	8
数据的值与类型	9

整型数	9
实型数	9
名字的类型	10
整型	10
实型	10
变量	10
数组	10
习题	11

第 三 章

算术赋值语句	13
算术表达式	13
算术运算符	13
算术表达式求值的运算顺序	13
表达式类型	14
算术赋值语句	14
习题	15

第 四 章

WRITE 语句	18
联机行式打印机	18
READ 语句	18
FORMAT 语句	19
格式说明	19
区域描述符	19
I 型描述符	21
F 型描述符	24
X 型描述符	25
STOP 语句	25
END 行	25
注释行	25
习题	28

第 五 章

H型描述符.....	29
A型描述符.....	30
E型描述符.....	31
用H型描述符实现走纸控制.....	32
输入-输出语句的执行.....	33
习题.....	36

第 六 章

DIMENSION 语句.....	38
类型语句.....	38
EQUIVALENCE 语句.....	39
计算机的控制卡片.....	40
程序结构.....	41
GO TO 语句.....	44
无条件 GO TO 语句.....	44
计算 GO TO 语句.....	44
流程图绘制原则.....	46

第 七 章

算术 IF 语句.....	48
习题.....	52

第 八 章

DO 语句.....	55
CONTINUE 语句.....	58

嵌套循环结构.....	67
习题.....	72

第 九 章

函数和过程子程序.....	74
标准函数.....	74
语句函数.....	76
FUNCTION 子程序.....	77
RETURN 语句.....	80
SUBROUTINE 子程序.....	80
CALL 语句.....	82
COMMON 语句.....	84
语句的顺序.....	84
习题.....	85

附 录

供非理工科学生做的多样化习题.....	87
---------------------	----

译 者 附 录

附录A 逻辑 IF 语句.....	89
附录B 子程序中的可调数组.....	92
附录C 关于 COMMON 语句的 补充说明.....	93
附录D EXTERNAL 语句.....	95
FORTAN IV 程序设计语言教材习题解答	98
中英名词对照表.....	200

第一章

绪 论

FORTRAN 是英语 FORMula TRANslation(公式翻译)的缩写,主要是为科学上的应用而设计的一种程序设计语言。它在 1956 年首次应用,1966 年 3 月美国标准协会(American Standards Association 即 A. S. A)把各种类型的 FORTRAN 语言统一标准化成为二种类型,即:

FORTRAN II 和 FORTRAN IV

这种语言允许使用者将程序编写成类似于标准数学符号的形式,例如:

FORTRAN 语句	数学符号
A=3.	$a=3$
B=A+C	$b=a+c$
C=A*B	$c=a \times b$
D=(A-B)**2-(A+B)/C	$d=(a-b)^2-\frac{a+b}{c}$

本书的主要目的是把 FORTRAN IV 介绍给那些对计算机了解很少的人。

计算机系统的基本部件

为了设计计算机程序,必须对计算机的各个组成部分有个一般的概念。程序员并不需要了解计算机运算的电子学原理或者计算机的逻辑设计,但是他应该了解计算机系统的各个基本部件。仔细观察图 1-1 所示的系统图就能对这些部件之间的关系有个较清晰的了解。

存 储 器

计算机的存储器分成很多个存储单元,其中的每一个存储单元用一个地址来标识,可存储指令或数据。

一个存储单元也称为一个计算机字(WORD)或字节(BYTE)①。它由一连串二进制数位(bit,比特)所组成。每个二进制数字有两个状态,即“0”或“1”。所有信息在机器内部均以二进制数字表示。每个存储单元中的二进制数位的数目在不同计算机中是各不相同的。例如,

① 字节是存储信息的基本单元,一般由 6 至 12 个二进制数位组成。一个字由一至几个字节组成。一个字占的二进制数位越长,则因其有效位数多,故精度也高。不过对一定存储器容量的计算机来说,字长长的,虽然计算精度高,但存储单元的数量就少了。——译者注

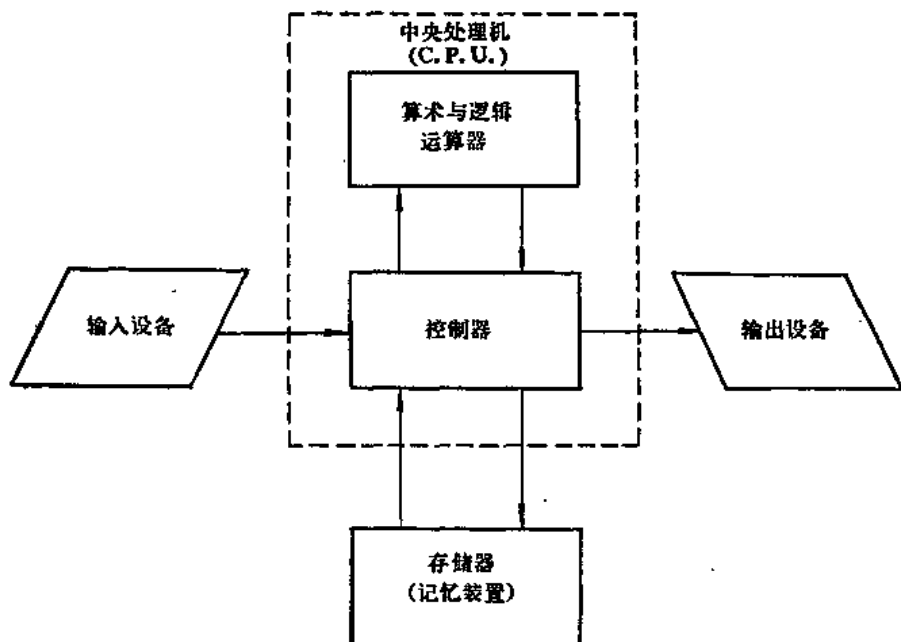


图 1-1 计算机系统的基本部件

IBM^①1130 型计算机的一个存储单元是 16 比特的字, IBM360 型计算机的一个存储单元是 8 比特的字节, 而 ICL^②1900 系列计算机的一个存储单元则是 24 比特的字, 等等。

一台计算机存储器的存储单元总数总是 K(即 1024)的倍数。例如一台 8 K16 比特字长的计算机系统有 8192 个存储单元, 每个存储单元有 16 个比特。

控 制 器

计算机的控制电路解释存储在存储器内的指令。以后它就要求其他部件进行操作, 例如启动卡片读入机读卡片, 用行式打印机打印出一行信息, 执行某些算术运算等等。

算术与逻辑运算器

运算器在控制器的指挥下执行算术运算, 即加、减、乘、除, 和逻辑运算。

输 入 设 备

指令和必需的数据通过输入设备输入至存储器内。用于输入的基本外围设备包括卡片读入机与纸带读入机。技术上更为先进的计算机系统则可使用磁带、磁盘或打字机(终端设备)作为输入设备。

① IBM 是美国国际商业机器公司(International Business Machine Corporation)的缩写, 该公司生产 IBM360 和 370 系统等计算机。——译者注

② ICL 是英国国际计算机有限公司(International Computers Limited)的缩写, 该公司生产 ICL1900 和 2900 系列等计算机。——译者注

卡片

标准卡片有 80 列, 每一列有 12 个穿孔位置。在一列中以不同的穿孔组合来表示一个字符。每一张卡片可以用穿孔机最多穿 80 个字符。(见图 1-2)

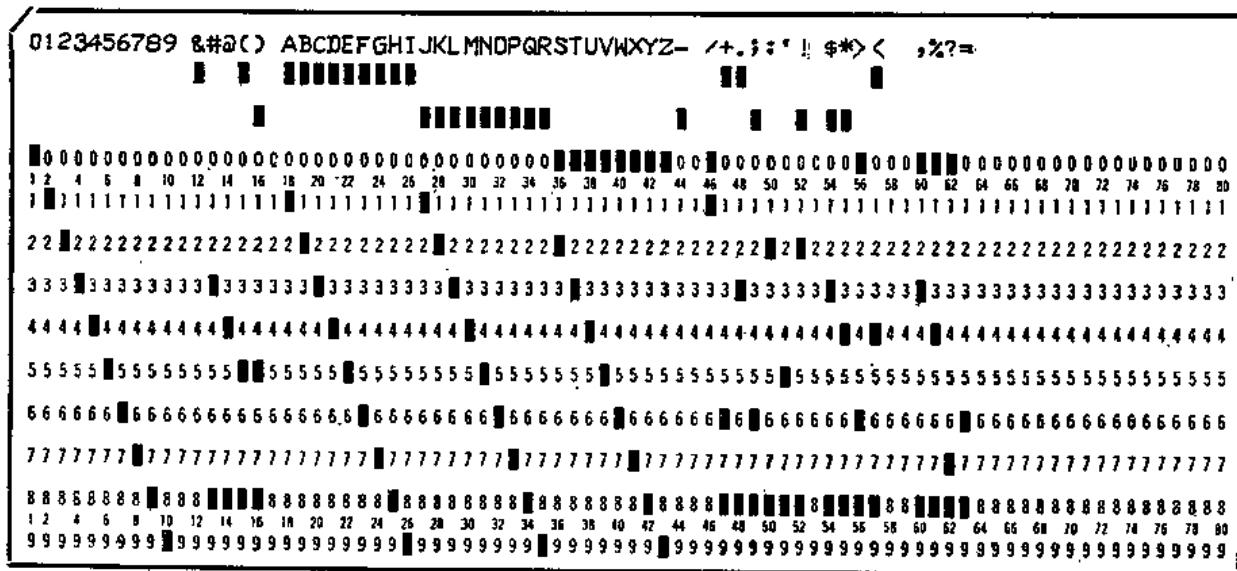


图 1-2 80 列穿孔卡片①

卡片读入机

在中央处理机的控制下, 通过卡片读入机可以把卡片上以穿孔表示的信息输入至存储器内, 其正常速度为每分钟 300 至 1600 张卡片。

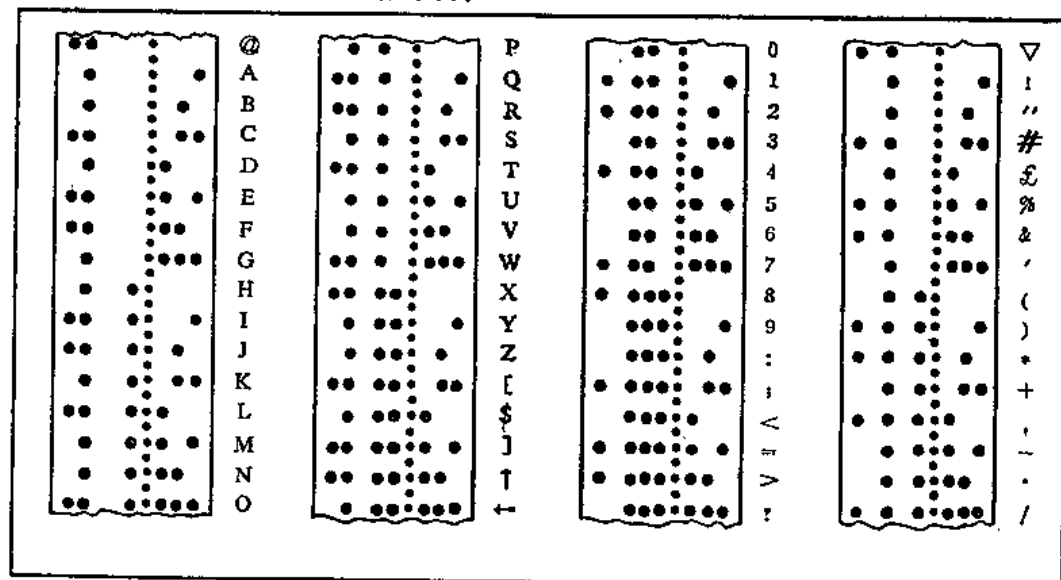


图 1-3 I. C. T. 8 孔位编码表

① 此 80 列穿孔卡片是以 EBCDIC 代码穿孔, ICL1900 系列计算机上的穿孔代码与 EBCDIC 代码略有不同。——译者注

② I. C. T. 为 International Computers & Tabulators, Limited 之缩写, 即英国国际计算机与造表机公司。——译者注

纸带

类似于卡片,纸带上一列有8个孔位,可以以不同的穿孔组合表示一个字符,或一个纸带控制符号。(见图1-3)

纸带读入机

在中央处理机的控制下通过纸带读入机可以把纸带上以穿孔表示的信息输入至存储器,其正常速度为每秒钟250至1000个字符。

输出设备

计算结果可以通过输出设备,打印在纸上或者记录在其他介质上,以备检查或储存之用。

用于输出的基本外围设备包括卡片穿孔机、纸带穿孔机和行式打印机。

有时也可用磁带或磁盘作为输出设备,供数据或信息的外部储存之用。

行式打印机

计算结果可以用行式打印机打印出来。每一印刷行有96个至160个打印位置。打印的速度范围为每分钟100行至2000行。

高速外围设备

高速外围设备例如磁盘、磁带与磁鼓等可以用来作为输入与输出设备。

计算机的程序编制步骤

计算机程序是一组指令,它由程序员根据某些规则或某种算法(计算步骤)而编写,以解决给定的问题。在计算机能够遵守这些指令去完成有顺序的运算之前,此程序必须穿在卡片上,以便计算机可以从输入设备上读入此程序(源程序卡片组),并把它编辑(翻译)为机器语言(目的程序)。把程序从源语言翻译成目的语言的工作由称为编译程序的特种程序来完成。机器语言程序一经生成,计算机即可执行指令,它们是在收到适当的数据(如果有的话)后才产生的。在程序编制过程中,语法上的(编码上的)和逻辑上的错误均可能发生。当出现语

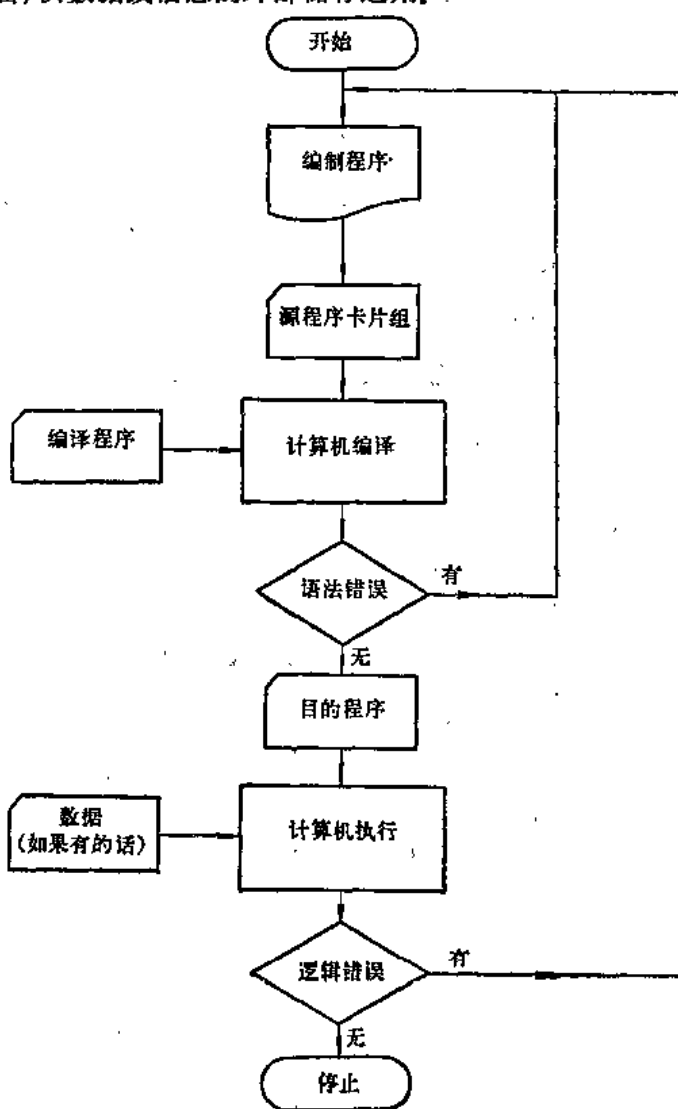


图1-4 计算机程序编制步骤

法上的错误时, 计算机的运行往往中断(或根本不能启动机器), 并且机器会给出一个信息。当出现逻辑上的错误时, 计算机的运行通常或者得不到结果, 或者得出错误的(使人误解的)结果。出错是常有的事。从计算机程序中找出并纠正错误的过程, 以及得出一种经过考验的和可以工作的程序的过程称为调试。计算机程序的编制和调试的整个过程可以用图 1-4 归纳说明。

习 题

1. C. P. U. 代表什么意思? 说明 C. P. U. 中所包含部件的名称。
2. 一台 64K 24 比特字长的计算机系统, 其存储器中有多少个存储单元? 同时给出比特的总数。
3. 一张卡片上最多能穿多少个字符? 在卡片的一列中如何穿孔以表示字符 'A'? (卡片上一列的穿孔位置从顶端至底部之编号为 10^①, 11, 0, 1, 2, 3, 4, 5, 6, 7, 8 和 9.)
4. 在什么条件下程序员应该重新编制他的程序?
5. 一个程序在计算机上的成功运行通常要通过哪两道关口的考验?

① 有的计算机卡片, 每一列的穿孔位置编号, 从顶端开始第一个编号称为 12, 而不是称为 10, 其余的则相同。——译者注

第 二 章

FORTRAN 程序

一个 FORTRAN 程序是由称为语句的一些句子所组成的。共有五种类型的语句，即：

算术赋值语句

执行算术运算，例如加、减、乘、除等，然后将结果值赋予一个变量或数组元素^①。

输入-输出(I/O)语句

描述输入-输出动作，它可在计算机内存储器 and 输入-输出介面之间传递数据。

控制语句

控制语句使程序员能支配程序的流向，以及终止程序的执行。

说明语句

通过说明变量的类型、数组的维数、输入或输出数据的格式等，指定数据的存储区域。

子程序语句

定义在程序中用到的 FUNCTION 子程序与 SUBROUTINE 子程序。

前三种语句是可执行语句，它们规定了具体的操作。后二种语句是不可执行语句，它们仅为 FORTRAN 编译程序顺利完成编译提供必需的信息。

FORTRAN 程序纸

FORTRAN 程序是写在 FORTRAN 程序纸(见书末插页图 2-1)上的，然后把卡片或纸带上以穿孔表示的程序输入到计算机中。

程序纸上的每一行有 80 列，相当于一张卡片上的列数，它分为三个区域(参看图 2-2)：

语句标号区(从第 1 至第 5 列)

任何可执行语句，在必要时均可以在第 1 至第 5 列上标以自 1 至 99999 的语句标号，以便别的语句引用它。在一个程序段内所有语句标号必需是唯一的，即不能重复；它们没有顺序的意义^②。如果使用注释行，则第 1 列上必需有一字母 C(见第四章)。

语句区(从第 6 至第 72 列)

语句内容写在第 7 至第 72 列。如果一个语句太长，则可以在下面各行继续写。第一行叫做起始行，后面跟着的叫做继续行。最多可以有 19 个继续行。

① 原文为“算术语句”，我们改译为“算术赋值语句”，而且增加一句：“然后将结果值赋予一个变量或数组元素”。——译者注

② 语句标号不必按数字大小依次排列，可在规定数字范围内任意选择。

属于不可执行语句的 FORMAT 语句，永远是被输入-输出语句引用的，所以必须有标号。——译者注

[illegible]

起始行的第 6 列应为 0 或空白,而继续行的第 6 列上应为除 0 或空白以外的任何字符。

标识区(从第 73 至第 80 列)

这一区域留给程序员用任何字符组来标识每一行,例如用有序数来标识每一行,这些标识字符可有可无,它在程序编译时是不起作用的。

基本字符

编制 FORTRAN 程序时使用下列基本字符:

字母和数字:	字母	A 到 Z
	数字	0 到 9
特种字符:	(	左括号
	)	右括号
	=	等号
	+	加号
	-	减号或连字符
	*	星号或乘号
	/	斜线或除号
	,	逗号
	.	句号或小数点
	'	撇号
		空白

名字

在编 FORTRAN 程序时可以用作变量的某一组数据或某一项数据,必需有一名字。变量、数组或函数的名字是一连串字母、数字字符,第一个字符必需是字母。

在一个名字中最多字符的数目取决于所用的计算机型号,如下表所示:

机 器 型 号	在一个名字中的最多字符数目
IBM 1130	5
IBM 360	6
ICL 1900 系列	32

本书采用 ICL1900 系列的规定,即在一个名字中最多有 32 个字符。

例:

正确的名字: A1, BCODE, INCOMETAX

不正确的名字: 1A48(因为第 1 个字符不是字母)

XY*(因为*不是字母或数字)

THEUNIVERSITYCONTRIBUTIONANDTAXES

(因为超过 32 个字符)

数据的值与类型

在 FORTRAN 语言中, 量的数值与类型在不同计算机中是各不相同的^①, 数值的大小范围取决于计算机存储器的容量(见表 2-1)。

表 2-1

值的类型	计算机型号	IBM 1130	IBM 360	ICL 1900 系列
	存储单元 容量单位	字 (16 个比特)	字节 (8 个比特)	字 (24 个比特)
整型	所占存储单元数目	1 个字(16 个比特)	4 个字节(32 个比特)	1 个字(24 个比特)
	二进制数的数值范围	$-2^{15}+1$ 到 $2^{15}-1$	$-2^{31}+1$ 到 $2^{31}-1$	$-2^{23}+1$ 到 $2^{23}-1$
	十进制数的数值范围	-32767 到 32767	-2147483647 到 2147483647	-8388607 到 8388607
实型	所占存储单元数目	2 个字(32 个比特)	4 个字节(32 个比特)	2 个字(48 个比特)
	指数的大致范围 ^②	-38 到 38	-75 到 75	-76 到 76
	有效数字的大致数目	7 位十进制数	7 位十进制数	11 位十进制数

本书采用 ICL 1900 系列

整型数

整型数的大小范围取决于计算机型号, 如表 2-1 所示。

例: 29, -300, 30779, -90000

实型数

实型数包含一个小数点, 或一个十进制数指数, 或者两者都有。它可以用下列任何形式表示:

形式	例
$\pm n$.	+3. 或 3.
$\pm n.n$	-4.379
$\pm .n$	.345 或 +.345
$\pm n.E \pm s$	-4.E-10
$\pm n.nE \pm s$	40.67E4 或 +40.67E+4
$\pm .nE \pm s$	-.768421E-4
$\pm nE \pm s$	-10E-10

此处 n 是一串十进制数字, E 表示对 10 进行乘方, 而 s 是一整数, 故 $40.34E-4$ 代表 40.34×10^{-4} 。

① 意思是: 量的数值范围在不同计算机中是不同的(见表 2-1)。此外, 可用的类型在不同计算机中也是不同的。除了整型和实型以外, 有些计算机还可以使用双精度型、复型和逻辑型的变量和常数。——译者注

② 指 10 的乘方数, 例如在 IBM1130 中是 10^{-38} 到 10^{38} 等。——译者注

名字的类型

整型

若不用类型语句(见第六章)加以说明,则整型名字^①的第一个字符应该是 I, J, K, L, M, N 中的一个。整型名字只能包含整数值。

例: ICL, KING, INCOME, LOST

实型

若不用类型语句加以说明,则实型名字^①的第一个字符必须是除 I, J, K, L, M 和 N 以外的一个字母。实型名字只能包含实数值。

例: RAMDA, BETA1, REALVALUE, TAX

变 量

变量是一个单项数据,它的值在程序运行过程中可以改变。每个变量由程序员赋予一个名字,以便引用它。

例: X, QUANTITY, VARY, INCOMES, ITEMS

数 组

数组是一群变量,它可用同一个名字称呼。数组中的每一个变量称为数组的一个元素。为了引用数组的每一个元素,要指明数组的名字及其下标。

例: ARRAY(1)	名字为 ARRAY 的一维实型数组的第一个元素。
NAME(1, 1)	二维整型数组 NAME 的第一个元素。
A13(3, 4)	二维实型数组 A13 的第 3 行第 4 列的元素。
BLOCK(3, 8, 5)	三维实型数组 BLOCK 的第 3 行第 8 列第 5 层的元素。

数组通常最多可以有三维,但对 ICL1900 系列计算机则最多可以有 32 维。

一个二维数组由两个下标值加以描述。例如,有 4 个在校学生,每入学 6 门课程。为了储存他们在这 6 门课程上的得分成绩,分配了 24 个存储单元给 SCORE(4, 6)。SCORE 是赋予这些存储单元的二维数组的名字。

如果每个学生的得分写在一横行上,则我们可以得出一个有 4 行(每个学生一行)和 6 列(每门课程一列)的表格,如表 2-2 所示。

上述表格的记录总数为 $(4 \times 6) = 24$ 。因而为了储存这个二维数组,必需指示计算机留出 24 个相邻的存储单元,每个对应于表中的一个记录。

^① 这里的名字包括变量、数组和函数的名字。——译者注