

普通高等教育通信类规划教材

FORTRAN 77

编程操作基础

高元华 王练辉 编

北京邮电学院出版社

FORTRAN 77 编程操作基础

高元华 王练辉 编

北京邮电学院出版社

(京)新登字162号

内 容 提 要

本书以通俗易懂的方式简单明了介绍 FORTRAN 77 语言的基本内容, 程序设计中的基本算法, 源程序在微机上运行的基本操作, 理论联系实际, 便于初学计算机高级语言的读者掌握“算法”、“语言”、“操作”之间的有机联系。

为避免学习“算法”、“语言”枯燥、抽象, 书中各章配有内容适当的例题和习题以及上机练习题, 便于读者举一反三, 掌握本书内容的要领。

本书可作为大专院校非计算机专业本科生、大专生及各类成人、中等专业学校的教材, 也可作为工程技术人员培训班的教材使用。

FORTRAN 77 编程操作基础

编 者: 高元华 王练辉

责任编辑: 王守平

*

北京邮电学院出版社出版

新华书店北京发行所发行 各地新华书店经售

北京通县向阳印刷厂印刷

*

850×1168 毫米 1/32 印张 12 字数 319 千字

1992年8月第一版 1992年8月第一次印刷

印数: 1—5000册

ISBN 7-5635-0094-4/PT·10 定价: 8.40元

前 言

目前,我们正处在一个科学技术飞跃发展的时代,而计算机的广泛应用,则是这个时代的重要标志。近年来,计算机在我国得到迅速的推广和应用,在科学计算、数据处理、通信网络、过程控制等方面,都取得了明显的效益。计算机语言是实现计算机应用的基础,现已成为高等院校学生和广大科技人员必修课程。由于FORTRAN 77 处理问题能力强、语句功能比较全面、便于实现结构化程序设计,不但适合于科技计算也适合于非数值运算,已在国内、外获得了广泛的应用。

本书是1988年出版的《算法·语言·操作》——FORTRAN 77 简明教程一书的修订本。在修订过程中,根据教学实践的需要对原书大部章节重新进行编写,力求以通俗易懂的方式,简单明了介绍FORTRAN 77 语言的基本内容,程序设计中的基本算法(新增一章“基本算法”),源程序在微机上运行的基本操作。阐述过程中理论联系实践,便于初学计算机高级语言的读者掌握“算法”、“语言”、“操作”之间的有机联系。另外,也通过对原书的修订改名为《FORTRAN 77 编程操作基础》。

为使读者把握各章的重点,书中每章的开始均有本章的内容提要;为使读者掌握常用基本语句的用法和编程的基本技巧,在书中编有内容丰富的例题和习题以及上机练习题,希望读者能边学习、边上机实践。为使读者适应计算机语言发展的需要,在书中最后一章,对FORTRAN 77 全集作了简要说明。

本书经过邮电部高校计算机教学指导委员会评审通过推荐出版。可作为电信工程、无线电工程、信息工程、管理工程等非计算机专业本科生《算法语言》课的教材,也可作计算机专业大专班

和在职人员培训教材。

本书一、二、八、十一至十七章由高元华编写,三至七、九、十章由王练辉编写。由于我们水平和能力有限,书中难免有不妥、错误之处,恳请读者批评指正。

编 者

1991.11.北邮

目 录

前 言

第一章 绪 论

§ 1-1 电子计算机简介	(1)
§ 1-2 计算机语言	(5)
§ 1-3 算法	(7)
§ 1-4 框图	(9)
§ 1-5 计算机的硬件和软件	(10)
习 题	(11)

第二章 FORTRAN 语言概述

§ 2-1 FORTRAN 字符集	(14)
§ 2-2 简单的 FORTRAN 源程序介绍	(15)
§ 2-3 FORTRAN 程序书写格式	(21)
§ 2-4 FORTRAN 程序运行步骤	(24)
习 题	(25)

第三章 表达式与赋值语句

§ 3-1 常数	(28)
§ 3-2 变量	(31)
§ 3-3 内部函数	(34)
§ 3-4 算术表达式	(38)
§ 3-5 算术赋值语句	(43)
§ 3-6 逻辑表达式与逻辑赋值语句	(47)

习 题	(53)
-----------	------

第四章 输入、输出(一)

§ 4-1 表控输入	(57)
§ 4-2 表控输出	(59)
§ 4-3 格式输出	(60)
§ 4-4 格式输入	(70)
§ 4-5 不用 FORMAT 语句的格式输入和输出	(75)
§ 4-6 停语句、暂停语句和结束语句	(75)
习 题	(78)

第五章 控制语句

§ 5-1 无条件转移语句 (GO TO 语句)	(83)
§ 5-2 逻辑条件语句 (逻辑 IF 语句)	(85)
§ 5-3 算术条件语句 (算术 IF 语句)	(92)
§ 5-4 计算转移语句	(95)
§ 5-5 标号赋值语句与赋值转移语句	(98)
§ 5-6 块 IF 及其相关语句	(100)
习 题	(108)

第六章 数 组

§ 6-1 数组与数组说明语句	(113)
§ 6-2 数组的输入和输出	(118)
§ 6-3 数据赋初值语句 (DATA 语句)	(121)
习 题	(123)

第七章 循 环

§ 7-1 循环语句 (DO 语句)	(126)
§ 7-2 关于循环的进一步说明	(128)

§ 7-3 继续语句.....	(131)
§ 7-4 关于循环的嵌套.....	(132)
§ 7-5 循环的控制转移.....	(134)
§ 7-6 程序举例.....	(136)
习 题	(144)

第八章 基本算法

§ 8-1 累 加.....	(149)
§ 8-2 累 乘.....	(150)
§ 8-3 筛 选.....	(151)
§ 8-4 枚 举.....	(153)
§ 8-5 递 推.....	(154)
§ 8-6 迭 代.....	(155)
§ 8-7 求最大值和最小值.....	(157)
§ 8-8 排 序.....	(158)
§ 8-9 查 找.....	(160)
§ 8-10 分类统计.....	(162)
习 题	(164)

第九章 输入、输出 (二)

§ 9-1 分隔符、走纸控制符.....	(166)
§ 9-2 格式说明的重复使用.....	(169)
§ 9-3 G 型描述符.....	(172)
§ 9-4 逻辑型描述符.....	(173)
§ 9-5 比例因子 mp	(174)
习 题	(177)

第十章 字 符

§ 10-1 字符型常数	(179)
--------------------	-------

§ 10-2	字符型变量和字符型数组	(180)
§ 10-3	字符表达式和字符变量的赋值	(181)
§ 10-4	字符关系表达式	(184)
§ 10-5	字符型数据的输入、输出	(185)
§ 10-6	字符型内部函数	(191)
习 题		(193)

第十一章 语句函数

§ 11-1	语句函数的定义	(196)
§ 11-2	语句函数的使用	(197)
§ 11-3	语句函数的程序举例	(199)
习 题		(201)

第十二章 子程序

§ 12-1	概 述	(204)
§ 12-2	函数子程序	(206)
§ 12-3	子例程子程序(SUBROUTINE 子程序)...	(214)
§ 12-4	外部语句和内部语句	(222)
§ 12-5	可调数组	(226)
§ 12-6	可变返回点	(227)
§ 12-7	结构化程序设计的概念	(229)
习 题		(231)

第十三章 FORTRAN 中的其它语句

§ 13-1	程序语句	(235)
§ 13-2	参数说明语句	(236)
§ 13-3	双精度型运算和复型运算	(237)
§ 13-4	公共(用)语句	(240)
§ 13-5	等价语句	(246)

§ 13-6 数据块子程序	(252)
习 题	(254)

第十四章 磁盘文件

§ 14-1 概 述	(258)
§ 14-2 无格式读、写语句	(260)
§ 14-3 打开(OPEN)语句和关闭(CLOSE)语句	(261)
§ 14-4 BACKSPACE 语句和 REWIND 语句 ...	(267)
§ 14-5 使用磁盘数据文件程序举例	(270)
习 题	(275)

第十五章 应用举例

§ 15-1 多套数据读入方法	(277)
§ 15-2 复数运算	(279)
§ 15-3 实系数和复系数的列主元高斯消去法——适 用于直流电路、交流电路的计算方法	(283)
§ 15-4 网络函数计算	(291)
§ 15-5 一种实用绘图子程序	(296)
习 题	(301)

第十六章 基本操作

§ 16-1 文件的命名	(304)
§ 16-2 磁盘简介	(306)
§ 16-3 PC-9801 FC 机键盘介绍	(307)
§ 16-4 DOS 的启动	(312)
§ 16-5 盘的格式化和文件的拷贝	(313)
§ 16-6 运行 FORTRAN 程序的步骤	(315)
§ 16-7 常用的 DOS 命令简介	(322)

§ 16-8 汉字键入简介	(326)
上机练习题	(327)

第十七章 FORTRAN 77 全集简要说明

§ 17-1 关于总体方面	(332)
§ 17-2 说明语句	(335)
§ 17-3 数据赋初值语句	(336)
§ 17-4 控制语句	(336)
§ 17-5 输入/输出语句	(337)
§ 17-6 格式说明	(343)
§ 17-7 函数和子程序	(346)
附录I FORTRAN 77 内部函数	(349)
附录II 可执行语句和非执行语句表	(353)
附录III 程序单位中语句和注释行的顺序	(354)
附录IV FORTRAN 77 语句形式表	(355)
附录V ASCII 字符集	(357)
附录VI MS-FORTRAN 77 错误信息说明	(358)

第一章 绪 论

内容提要 本章主要介绍一些有关算法语言常用的术语,如:什么叫算法、语言,什么叫硬件、软件,什么是字长等。同时为适应读者能尽快使用计算机的需要,简单地介绍了有关电子计算机的特点、应用、结构,这些都是为掌握本书内容的基本知识。其中,有关算法的概念则是贯穿本书的一条主线,希望读者予以足够的重视。

电子计算机是本世纪最重大的工业革命成果之一。自从1946年世界上第一台计算机问世以来,它已被越来越广泛地应用于科学计算、工程设计、数据处理以及人们日常生活等各个领域,并已成为减轻人们的体力与脑力劳动的有效工具。在“教育要面向现代化,面向世界、面向未来”的形势下,掌握使用计算机技能已成为新时期人才培养的一个重要组成部分。

§ 1-1 电子计算机简介

1.1.1 电子计算机的特点

1. 计算速度快

所谓计算机的运算速度,是指一秒钟内能够存取指令的数目。指令即是指示计算机进行基本操作的命令。比如有一台计算机,它具有一秒钟存(或取)20万条指令的能力,则这台计算机的运算速度为20万次/秒。我国研制的“银河”计算机,其运算速度为每秒1亿次,又如ACOS-450机为270万次/秒。国外已有每秒达十几亿次的计算机。

2. 存储量大

电子计算机的存储容量,是以字节(byte 简称为“B”)为单位计算的。

计算机中数的表示方法一般都采用二进制,即只有0和1两个数字,例如10110101就是二进制数。二进制数的每一个数位称为位。在计算机中作为一个整体进行运算、存储和处理的一组二进制数称为计算机字,简称为字。又把计算机的每个字所含的位数称为字长。在计算机中用二进制数表示信息的最小单位,称为字节。通常以8位二进制数为一个字节(简称为B),则4位为半个字节,16位为2个字节。每1024个字节称为1千字节(简称为kB),而 1024×1024 个字节称为1兆字节(简称为MB)。

不同机型的计算机,其内存储器容量是不同的。大型机或巨型机的内存储器容量一般在几MB以上;中、小型机一般在几十kB到几MB;而微型机一般在几kB到1MB左右。

在计算机中一个普通的数据常用4个字节表示。因此一个内存储量仅为64kB的微机,就可存放一万个左右的数据。

3. 运算精度高

在计算机中运算的数据一般有6~7位有效数字,若使用双精度类型的数据可达15~16位有效数字。

4. 自动工作

在计算机运行期间,人们只要事先编好程序将其存入计算机,则计算机即能按指令自动操作,完成计算并打印出计算结果。

由此可见,电子计算机是一种以高速度、高精度、具有存储能力、由程序控制自动进行操作的电子设备。

1.1.2 电子计算机的用途

随着计算机的发展,它的应用日趋广泛与深入。从工业、农业、科研、教育、国防……直到人们的日常生活(实际上电子计算机就是俗称的“电脑”),处处都显示它的“才华”。概括起来,计

其相互关系如图 1-1 所示。

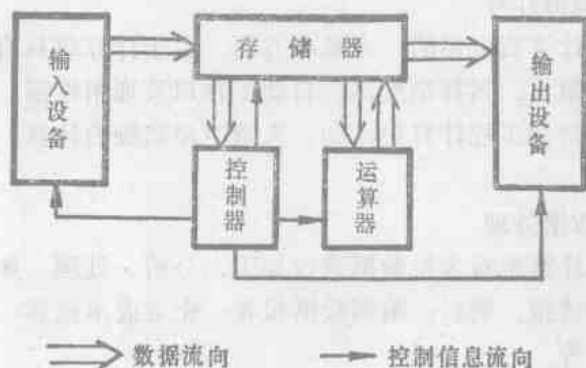


图 1-1 计算机结构框图

为了便于理解计算机的结构，我们用算盘的 计算 进行“类比”。使用算盘进行计算时，“算盘”本身犹如一个“运算器”；人的“脑和手”控制算盘就是一个“控制器”，而所需的计算内容和结果，一般记录在纸上，这纸就是一个“存储器”。

1. 存储器

用于存放程序和数据，并可按需要向存储器存、取数据。一般为了管理上的方便，可把一个存储器分为若干个单元，每个单元好象一个大旅馆中一个个房间，每个存储单元均有一个编号，该编号常称为地址。但需注意，不要把存储单元的地址和存储单元中的内容两者混淆，前者是指存储单元的标志，即用来查找（或访问）指定的存储单元，后者是指所需存（或取）的内容。另外，存储器还有一个特点：当其内容被“取”出之后，并不改变存储单元中原有的内容，只有向该存储器单元存入新的内容时，其原来的内容才会被“冲”掉，这样对存储器单元的“存”与“取”应理解为“写入”与“读出”。

2 运算器

用其完成各种算术运算、逻辑运算以及字符运算。

程序“翻译”，目前机器语言仅在机器内部中使用。

2. 汇编语言

由于机器语言难于记忆，为了帮助人们记忆指令，方便编写程序，出现了汇编语言。在汇编语言中，每一条指令都以助记符号形式出现，而且与机器语言的指令一一对应，它用助记符号代替二进制代码串，例如用 ADD 表示“加法”。

虽然汇编语言比机器语言有了进步，但仍存在通用性差，编程效率低，不易阅读的缺点，因此又发展了高级程序设计语言。

3. 高级语言

机器语言、汇编语言都是面向机器的语言，所以又统称为低级语言，但对广大用户来讲，往往只希望使用计算机解决具体问题，对计算机内部的构造并不需要关心和熟悉。为了克服广大用户使用计算机的困难，人们又设计了高级语言（又称为“程序设计语言”）。使用高级语言时，用户不必了解计算机的内部构造，即不依赖于具体机器，故通用性好，又由于高级语言与人们习惯上使用的自然语言和数学语言比较接近，因而容易被广大用户所理解和接受，为此计算机的应用才日益广泛地深入到各个领域。

但需指出，高级语言必须经过“翻译”变成机器语言，计算机才能执行。这种“翻译”任务是由“编译程序”来完成的，其工作过程如图 1-2 所示。



图 1-2

编译程序可将使用高级语言编写的程序(又称为源程序)翻译为机器语言程序(又称为目的程序或目标程序)。这里顺便指出所谓“程序”即是为解决某一问题而设计的有序指令的集合。

常用的高级语言有:

FORTRAN(Formula Translation)主要用于科学计算,它是第一个被广泛采用并且目前仍在广泛使用的语言。

ALGOL 60 (Algorithmic Language)也是一种科学计算的语言,具有分程序嵌套结构。

BASIC (Beginner All-purpose Symbolic Instruction Code),它是由 FORTRAN 等高级语言中提炼、简化而来,程序结构简单,语法浅显易懂,适合于初学者。但 BASIC 语言发展也很快,现代的 BASIC 语言的功能已大大增强,应用也相当广泛。

COBOL (Common Business Oriented Language),这是一种为处理商业资料而设计的语言。

PL/1 (Programming Language/1),它是 IBM 公司开发的多用程序设计语言。

PASCAL (pascal 是十七世纪法国哲学家 Blaisl Pascal 的名字,他发明了计算器),它是在 ALGOL 语言基础上发展起来的结构化程序设计语言。

C 语言,这是一种适合于系统程序设计的语言,常用在管理信息系统、分布式处理系统、知识信息处理系统以及图形处理系统等方面。

§ 1-3 算 法

所谓算法是指为解决一个问题而采取的有限步骤。即如何使用计算机一步步地进行工作的具体过程。如:我们用高级语言编写的程序,指挥计算机按一定的指令工作,这就是一种计算机的算

法。下面举例说明如何设计一个计算的算法。

例如，求 $1 + \frac{1}{3} + \frac{1}{5} + \cdots + \frac{1}{99}$ 的值，可采用如下算法：

- (1) 设 $M=0$ (M 作为累加变量用的存储单元)；
- (2) 设 $N=1$ (对变量 N 赋初值)；
- (3) $M + \frac{1}{N} \rightarrow M$ (累加)；
- (4) $N+2 \rightarrow N$ (计数)；
- (5) 若 $N \leq 99$ ，转去执行(3)及其后的各步骤(迭代)，否则执行(6) (分支)；
- (6) 打印出 M 的值 (输出结果)。

写出算法之后，再根据它选用某种高级语言写出程序。程序设计的关键在于设计一个好的算法。一个好的算法应当是：能够获得正确的结果；容易阅读理解，即易读性好；计算机执行时具有较高的效率(占用较少的机时和计算机的内存)。

“算法”应具有以下几个特点：

(1) 有限性 一个算法应当在执行有限步之后就结束，不当包含无终止的循环(死循环)。

(2) 确定性 在算法中每一个步骤，必须是确切定义的，而不应当含糊不清或模棱两可。

(3) 具有零个或多个输入量。所谓输入量是指在算法开始执行前，对算法给出的最初量。

(4) 可执行性 算法中的每一步都是能够准确地进行的。如进行 $1/N$ 的除法运算中，当 N 等于零时(即 $1/N$ 溢出)，则算法无法执行。

(5) 算法执行完毕后有一个或多个输出量。

关于常用的基本算法，我们将在第八章中再作具体介绍。