

FORTRAN 程序设计

郝柏林 编著

人民邮电出版社

内 容 简 介

*FORTRAN*是国内外广泛用于科技计算的一种算法语言。本书由浅入深地介绍用这种语言为电子计算机编写程序的基本概念、技巧和实例，同时提供了一批可供实际引用的程序块。全书结构既照顾到自学程序设计的入门知识，也为使用过*FORTRAN*或其他语言的读者讲述了一些比较细致的问题。书末附录可供实际程序工作中查阅参考。

本书可供具有理工科大学基础数学知识的工程技术和科研人员、高等学校师生阅读。

FORTRAN 程 序 设 计

郝柏林 编 著

人民邮电出版社出版

北京东长安街27号

河北省邮电印刷厂印刷

新华书店北京发行所发行

各地新华书店经售

开本：787×1092 1/32 1980年3月第 一 版

印张：12 24/32 页数：204 1980年3月河北第一次印刷

字数：291千字 印数：1—22,100册

统一书号：15045·总2347-无689

定价：1.30元

序 言

随着我国社会主义建设事业的蓬勃发展，电子计算机在科学研究和工业生产中日益普及。将来科技人员都需要学会电子计算机的程序设计，就像他们现在会拉计算尺一样。各种应用程序也要和图书期刊一起，变成普通的技术资料。科技工作者会读程序和编程序，得心应手地使用电子计算机这个工具，就能更好地为社会主义祖国的四个现代化作出贡献。

在科学技术计算方面，*FORTRAN* 是当前国际上最流行的算法语言。就是在 *ALGOL* 语言的故乡欧洲，*FORTRAN* 语言的使用也已经遥遥领先。我国自己研制的计算机，也日益普遍地配有 *FORTRAN* 语言的编译程序。本书将按照国际标准化组织 (ISO) 1972 年推荐的标准，向读者介绍这种语言。我们希望具有理工科大学基础数理知识的读者，通过自学和上机实践，能够大致读懂文献中与自己专业有关的 *FORTRAN* 程序，逐步学会编写不太复杂的程序，解决工作中遇到的问题，进而选用图书资料中现成的算法程序块，组织比较大的科技计算程序。这里要强调指出，程序设计的关键，不仅在于掌握算法语言的规定，更重要的是熟悉所涉及到的专业和算法知识；而这本书只能讲述算法语言本身。

本书着重从使用角度介绍 *FORTRAN* 语言。全书可以分成四个部分。前三章，特别是第一、二章，用几个比较浅显的

例子，使读者初步接触FORTRAN语言的主要成分，感受程序设计的过程。使用过BCY、ALGOL等语言，有算题经验的读者，可以跳过前两章。第四、五、六章详细介绍许多具体的语法规则，有较多的实例和习题。通过阅读前六章，伴以适量的上机实践，读者可对FORTRAN程序设计的基本精神有所领会。而在第七、八、九章涉及程序设计中一些更细致的问题时，就以讨论为主，减少了示例。第十章用一个完整的科技程序，演示如何提出问题、分析算法，把它化成宜于编程序的形式，然后尽量选用现成的算法程序模块，组织出比较复杂的程序。书末的一批附录、专备实际工作中查阅，有一些附录兼起索引作用。

毛主席在《中国革命战争的战略问题》一书中指出：“读书是学习，使用也是学习，而且是更重要的学习。从战争学习战争——这是我们的主要方法。”这段话，对于每个学习程序设计的同志，都是十分贴切的。一定要从开始学习时起，就敢于动手编写一些简单程序，并且拿到计算机上去试算。这样很快就能领会到，程序设计其实是一件极其简单的事情。通过更多的实践，将会进一步认识到，在这“极其简单”之中，还蕴藏着千变万化的可能性。熟练地运用简单的规定，可以巧妙地解决相当复杂的实际问题。程序设计和我国传统的围棋艺术，都是简中寓繁、静中有动，有着异曲同工之妙。

本书初稿是近几年在中国科学院物理研究所举办的FORTRAN学习班的讲义。定稿过程中，考虑了许多同志的有益见解，从一些同志所编的程序中抽取了有技巧或有错误的实例。第十章是与张淑誉同志合写的。卢琪同志准备了附录A和G、协助改写了前三章，并分担了出版过程中的一些技术性工作。仲萃豪、王寿松、韩淑娟同志仔细阅读了手稿，提出许多

宝贵建议。谨向他们表示感谢。

程序员是经常犯错误并在克服错误中前进的。这本书也不例外。我们诚恳地期望读者提出批评和建议。

作者 一九七九年二月

目 录

序 言

第一章 引论	(1)
第一节 电子计算机.....	(1)
第二节 程序设计自动化和FORTRAN语言	(3)
第三节 输入输出设备和读写语句.....	(7)
一、字符 (7) 二、字符输入输出设备 (9)	
三、简单的读写语句 (11)	
第四节 电子计算机的服务方式.....	(14)
第二章 FORTRAN程序设计简例	(18)
第一节 求平方根的程序.....	(18)
第二节 计算三角形面积的程序.....	(27)
第三节 框图符号说明.....	(31)
第三章 FORTRAN程序的结构	(34)
第一节 块内分行和行内分格.....	(34)
第二节 四种程序块.....	(36)
一、主程序 (36) 二、子程序 (37)	
三、函数 (37) 四、初始数据块 (38)	
第三节 子程序和函数的使用.....	(38)
第四节 用切线法解超越方程.....	(44)
一、问题的提出和分析 (44) 二、切线法子程序 (45)	
三、三个程序块之间的关系 (51)	
第五节 程序的装配.....	(56)

第四章 常数、变量和表达式	(59)
第一节 FORTRAN字符和名字	(59)
第二节 各种类型的常数和变量	(61)
一、整常数和整变量(61) 二、实常数和实变量(63)	
三、双精度常数和变量(65) 四、复常数和复变量(66)	
五、双精度复常数和复变量 (67)	
六、逻辑常数和逻辑变量 (67) 七、文字常数 (68)	
八、隐含说明和参数说明 (69)	
第三节 数组和下标变量	(71)
一、一维数组 (72) 二、多维数组 (73)	
第四节 各类型运算对象小结	(77)
第五节 表达式	(78)
一、算术表达式 (78) 二、关系表达式 (81)	
三、逻辑表达式(83) 四、表达式中的混合类型运算(84)	
第五章 基本的计算和控制语句	(86)
第一节 赋值语句	(87)
一、算术赋值语句 (87) 二、逻辑赋值语句 (90)	
第二节 停和暂停语句	(91)
第三节 无条件转语句	(93)
第四节 条件语句	(95)
一、逻辑条件语句 (95) 二、算术条件语句 (99)	
第五节 计算转和赋值转语句	(103)
一、计算转语句 (104) 二、赋值转语句 (106)	
第六节 程序实例和习题	(108)
一、阶乘和二项式系数 (108)	
二、大量数据的简单统计处理 (111)	
第七节 循环语句和继续语句	(115)

一、控制变量和循环参量 (118)	二、循环的嵌套 (120)
三、与循环有关的分枝和转移 (123)	
四、控制变量和循环参量的引用与赋值 (128)	
第八节 程序实例和习题	(131)
一、累加、累乘和计数 (131)	二、矩阵运算符程序 (134)
三、计算切比雪夫多项式的程序 (135)	
四、分段拉氏插值子程序 (136)	
第六章 输入输出和格式描述	(141)
第一节 通道号和输入输出名单	(142)
一、通道号 (142)	二、输入输出名单和隐含循环 (144)
第二节 格式读写语句和格式描述	(146)
一、格式语句 (146)	二、格式符简表 (147)
三、隔离符 (148)	四、记录 (149)
五、纵向格式描述 (151)	
第三节 格式符介绍	(153)
一、整数格式符 I (154)	二、定点数格式符 F (156)
三、浮点数格式符 E (157)	四、广义浮点数格式符 G (158)
五、双精度浮点数格式符 D (159)	六、逻辑格式符 L (160)
七、空格符 X (160)	八、文字格式符 H (161)
九、字符格式符 A (163)	十、比例因子 P (165)
第四节 程序实例和习题	(166)
一、切比雪夫多项式表 (166)	
二、打印曲线的子程序 (168)	
第五节 专用读写语句	(173)
一、对话式读写 (174)	二、二进制读写 (174)
第六节 二进制数据的处理	(175)
一、按位运算和“掩码”技巧 (175)	

二、五八单位译码 (179)	
第七节 再论格式输入	(181)
一、格式输入注意事项 (183)	
二、数组格式与自带格式输入 (185)	
三、读语句中的错误处理 (188)	
第八节 文件的输入输出	(189)
第九节 磁带的读写	(191)
第十节 绘图和显示	(195)
一、静态曲线显示子程序 (195)	
二、数字绘图仪子程序 (197)	
第七章 函数和子程序	(202)
第一节 语句函数	(202)
第二节 基本函数	(204)
一、内部函数和基本外部函数 (205)	
二、引用基本函数的注意事项 (206)	
第三节 函数分程序	(207)
一、函数的名字和类型 (208)	
二、哑元和实元的对应关系 (209)	
第四节 子程序	(213)
第五节 外部语句	(214)
第六节 四种过程的对比	(219)
第八章 各程序块间的联系	(221)
第一节 公用区和等价语句	(221)
一、无名公用区 (222) 二、有名公用区 (224)	
三、等价语句 (225)	
第二节 程序块间数据传送方法的对比	(228)
第三节 数组运用中的一些问题	(230)

一、后继函数和自动后继 (230)	二、变界数组 (232)
第四节 变量的初始赋值	(236)
一、初值语句 (237)	二、初始数据块 (239)
第五节 函数引用中的副作用	(240)
第九章 程序的组织和调试	(244)
第一节 程序的总体规划	(244)
第二节 提高程序质量的措施	(247)
一、选择计算方法 (248)	二、提高运算速度 (250)
三、压缩存储用量 (254)	四、保证必要精度 (255)
第三节 程序的调试和查错	(260)
一、调试措施 (260)	二、调试步骤 (262)
三、常见错误举例 (265)	
第十章 科技程序实例——高阶线性自动调节系统的	
稳定性和暂态过程	(267)
第一节 几个算法子程序	(267)
一、实矩阵求逆程序 (267)	二、一般实矩阵的特征值 (270)
三、常微分方程组的初值问题 (279)	
第二节 自动调节系统算法分析	(281)
一、传递函数 (282)	二、连接矩阵和给定函数 (284)
三、形成基本方程组 (286)	四、分离代数部份 (287)
五、总传递函数和系统的稳定性 (296)	
第三节 框图和程序	(298)
第四节 计算实例和输出形式	(312)
习题答案	(317)
附录 A FORTRAN 语句简表	(335)
附录 B FORTRAN 规定对照表	(339)
附录 C 基本函数表	(346)

附录D	七位信息交换代码	(352)
附录E	五单位第二类国际电报代码	(354)
附录F	二、八、十、十六进制对照表	(355)
附录G	十套常用算法子程序	(356)
一、	求一般实矩阵特征值子程序 <i>ELHQR</i>	(356)
二、	主元消去法解线性方程组子程序 <i>PIVOT</i>	(357)
三、	实矩阵求逆子程序 <i>MINVRT</i>	(359)
四、	一般实矩阵奇异值分解子程序 <i>SVD</i>	(359)
五、	求实多项式全部零点子程序 <i>RPOLY</i>	(367)
六、	解非线性方程组的拟牛顿法子程序 <i>NLES</i>	(385)
七、	最小二乘法作曲线拟合子程序 <i>LESQR</i>	(387)
八、	一重积分函数子程序 <i>QUAD</i>	(389)
九、	龙格——库塔法解常微分方程组初值问题子程序 <i>RGKT</i>	(391)
十、	快速富里叶变换子程序 <i>XFFT</i>	(391)
附录H	本书中的子程序和函数索引	(393)
参考文献		(395)

第一章 引 论

初学计算机，会感到它盘根错节，互相牵扯，似乎无从下手。这里重要的是“全局在胸”和“单刀直入”两条。“全局在胸”，是说对于计算机各个部分的功能和联系，先要有一个大致的了解。“单刀直入”，是说应当结合自己的工作需要，从简单处作起来，边干边学，在实践过程中增加对整体和局部的了解。这一章先介绍一些关于计算机和算法语言的基本概念，使读者初步作到“全局在胸”。从下一章开始，“单刀直入”，请读者体验编程算题的全过程。

第一节 电子计算机

人类早在远古的生产实践中，就形成了数和运算的概念。随着生产的发展，数的运算也日趋复杂，于是出现了各种运算工具。从算盘、手摇计算机到电子计算机，生动地反映了这一发展过程。

一个运算过程常常涉及大量原始数据、中间和最终的计算结果。要把这些数保存在“存贮器”中。记录纸是最原始的存贮器，但是容量有限，读写很慢。电子计算机采用了各种磁性材料制备的存贮器，用电信号读写，速度可以达到每秒钟存取百万个以上数据。现在半导体大规模集成电路存贮器，速度更快，容量更大，正在继续发展，将来很有希望取代磁性存贮器。

存贮器的主要指标是容量和存取时间。容量以字数和字长（二进制位数）表示。每个字，又称为存贮单元，可以存放一个或几个数，一个数也可能占几个字，根据数据的性质和精确程度而定。有些计算机把每个字分成几节，例如32位的一个字分成四个字节，每个字节占八位。比较存贮器容量时，不管是用字或字节作单位，通常都把1024（ 2^{10} ）字（节），简称为一千字（节）。我们说，某台计算机容量是十三万字或四百万字节等，都是这样算的。上面讲的是主存贮器。为了提高处理速度，许多计算机还备有更快更小的超快速存贮器，同时尽量使各种存贮和运算部件高度重迭并行地动作。

存贮器不仅是用来存放数据。计算机应当作的事情也用顺序排列的数字表示出来，放在存贮器里。这些数字，每一位都有特殊的意义，有的代表要作的事情，如加减乘除、取送读写、比较大小等等，这些位的数字，叫做操作码。另一些位的数字则代表要处理的数据放在存贮器的那个单元中，叫地址码。操作码和地址码构成指令。由人工精心排列好的一串指令组成“程序”。计算机就是在“程序”的指挥下进行各种操作以完成特定功能的。程序和数据外表没有什么差别，都是放在存贮器中的数字。

除存贮器外，电子计算机的另一个核心部分是中央处理机。随着工艺水平的提高，中央处理机的体积越来越小，处理速度日益提高。中央处理机自动地按一定顺序从存贮器中取出指令，对它的每一位进行分析解释，按指令要求迅速完成处理，包括运算和读写。对于算法语言的使用者，中央处理机的功能反映为语言的处理能力，关系比存贮器更为间接，我们就不再详述。

概括起来说，存贮器和中央处理机构成主机。为了使主机

与人以及外部现场发生关系，还需要有各种配套设备。主机和各种配套设备，通常称为“硬件”。硬件好比没有受过训练的头脑和四肢。一个计算机厂家不仅要生产硬件，更重要的是必须使自己的产品“受教育”，为它们配备“编译程序”，使它们能读懂和执行不是用指令，而是用“语言”写出来的程序。还要配上让计算机管理自己的程序（“操作系统”）和方便用户的程序（“服务程序”），以及其他各种程序。这些程序通称为“软件”。软件好比是知识，它反映计算机的“文化水平”。这是一种无形的财富，其中凝聚了人类使用电子计算机的大量经验和技巧。硬件在软件调动指挥下，才能获得生命力，成为使人类的处理能力大幅度提高的计算机系统。

一个软件或硬件工作者必须对计算机有更细致的了解。在工作中使用算法语言的科技人员，为了充分发挥计算机的潜力，也应当懂得一些有关软、硬件的知识。然而这不是使用计算机的先决条件，可以在实践中逐步学会。

第二节 程序设计自动化和FORTRAN语言

每种类型的计算机都有自己独特的指令系统。小型机只有十几条指令，大型机可有几百条指令。为了给一台计算机编制程序，就必须懂得它的“机器语言”，即掌握它的指令系统。编制程序时，无论是存贮分配、指令执行顺序、输入输出组织，都得由程序员一条一条地精心安排。这是一种极其烦琐、容易出错的工作。检查和排除错误，也要花费大量的人力。在使用机器的实践中，人们很快认识到可以把编制程序和检查错误的繁重劳动在很大程度上交给计算机自己去作，即实现程序设计自动化。

为此要规定一套与具体计算机无关的“算法语言”。算题人员只须用这种语言写出一篇简练文字，称为“源程序”。每种计算机在研制生产过程中，要由专业程序员用“机器语言”编写一套“编译程序”，它能够接受和阅读源程序，检查其中错误，最后把它翻译成用机器语言表述、能够直接由机器去执行的“目标程序”。机器语言虽然五花八门，算法语言却可以比较统一。这是因为用各种机器语言编写程序，都有许多共性的东西：要对某些存贮单元取(送)数，要完成一定运算加工，要多次重复执行某一段程序(“循环”)；要判断某种条件是否成立而去执行不同的程序段(“分支”)，多处重复出现的某组指令，要只写一次，重复套用(“子程序”)等等。这样，早在只用机器语言编写程序的时期，就逐步形成了为存贮单元起名字、赋值，在程序中实现循环、分支、子程序等等概念。在算法语言的规定中，这些概念被进一步概括和系统化。

用算法语言解题的过程示于图1—1中。图中着意显示了程序设计是一个循环往复，渐趋正确的认识过程。

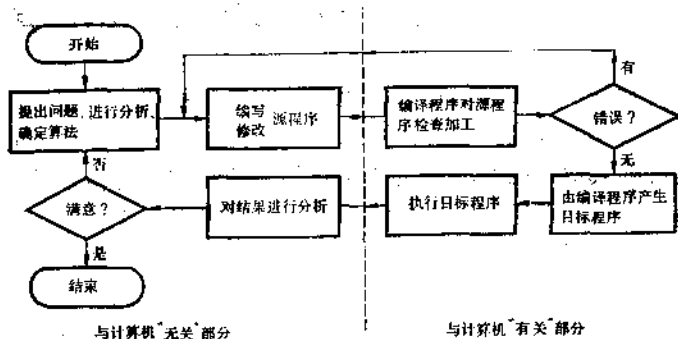


图 1—1 用算法语言解题的过程

图1—1中由算题人员用算法语言编写的程序，特称为“源

程序”。编译程序是计算机的一个构成部份，它把源程序翻译成机器能够执行的“目标程序”。源程序可以由计算机印出来。知道算法语言的人原则上都可以读懂源程序（这里说“原则上”，是因为事实上还应具备关于所算问题的专业知识）。目标程序通常印不出来，即使印出来也只有经过专门训练的程序员才能阅读。因此，这本书里讲到的程序都是源程序。

正是由于编译程序是计算机的一部分，这本书中甚至混用“编译程序”和“计算机”这两个词儿。例如，“计算机查出错误”，其实是“编译程序查出错误”。

从五十年代中期以来，制定了多种算法语言，有的语言适用于一般科技计算，有的语言适宜作行政管理和经济业务等等。同时在实践中形成了关于算法语言本身和它们的编译技术的比较系统的理论。这已经是计算机科学的一个重要部份，有专业队伍从事研究和设计。一般计算机用户，只须熟悉和运用某一种或几种语言，来解决各种实际问题。

本书介绍的FORTRAN语言，是五十年代中期为某些IBM计算机设计的。它得名于FORMula TRANslator(公式翻译)一词。FORTRAN语言经历了几个发展阶段，形成了大小两种方案，从六十年代初开始标准化。1966年制定了两级美国标准：标准FORTRAN(大致相当过去的FORTRAN IV)；标准基本FORTRAN(大致相当从前的FORTRAN II)^[1]。

目前FORTRAN已经成为科技计算中最通用的一种国际语言，文献资料中有大量用它写成的程序。我国DJS-6，DJS-8计算机，以及DJS-100和DJS-200系列也都配有这种语言。1972年国际标准化组织(ISO)为它制定了推荐标准^[2]。这个标准分为三级：完全的、中间的和基本的

• 方括弧中是书末参考文献的序号。