

BASIC FORTRAN

BASIC FORTRAN 程序设计题解

郭淑芬 编著

机械工业出版社

BASIC FORTRAN

BASIC FORTRAN 程序设计题解

郭淑芬 编著



机械工业出版社

本书每题都分别用 BASIC和FORTRAN 语言编写程序,有解题方法、编写程序的说明及运行结果等。本题解第一章介绍编程基础知识,并探讨如何提高程序设计的质量;第二章至第六章为题解主体,分别有基本运算;行列式、矩阵、解方程;排序、统计、数表等例题;也有打印各种图形的练习及趣味题。每章习题都由浅入深,既适合于初学者,也适合于已经掌握 BASIC和 FORTRAN 中的一种语言进而学习另外一种语言的读者。

本书可供中学、中专、大专院校师生及科技工作者阅读。

BASIC FORTRAN 程序设计题解

郭淑芬 编著

责任编辑:王中玉 责任校对:刘思璐
封面设计:田淑文 版式设计:罗文莉
责任印制:尹德伦

机械工业出版社出版(北京阜成门外百万庄南街一号)
(北京市书刊出版业营业许可证出字第117号)
人民交通出版社印刷厂印刷
新华书店北京发行所发行·新华书店经售

开本 $787 \times 1092^{1/16}$ · 印张 $25\frac{3}{4}$ · 字数 624 千字
1990年5月北京第一版·1990年5月北京第一次印刷
印数 00,001—2,750 · 定价: 21.00元

ISBN 7-111-01243-7/TP·78

前 言

BASIC 和 FORTRAN 语言是广为使用的计算机语言。随着计算机技术的迅速发展, 计算机的应用日益广泛、深入, 目前中学、大专院校、科研院所和工厂等都使用了大小不等的计算机。计算机不仅用于科学计算、企业管理, 还可绘图、教学、做游戏。程序设计是人与计算机对话的“翻译官”, 要想学会使用计算机就一定要学会用计算机语言编写程序。本书正是为初学 BASIC 和 FORTRAN 语言的读者而编写的。另外, 目前一般的中学、中等专业学校大多开设了 BASIC 程序设计语言课, 而大专院校却较多地开设 FORTRAN 等程序设计语言课; 有些技术人员学过 BASIC 语言, 然而在他们的工程计算中往往使用 FORTRAN 语言等, 或者是学习过 BASIC 语言又再学 FORTRAN 语言。为了帮助这些读者较快地掌握另一种语言而编写了本题解。

本题解的每一题目都分别用 BASIC 和 FORTRAN 语言编程, 很多题目同时提供了多种编程方案, 以便启发思路, 使读者可以从分析比较中扩大视野, 提高程序设计的水平。题解一般选择有代表性的题目, 主要针对工程技术的应用需要, 兼顾趣味性。题目的范围概括了 BASIC 和 FORTRAN 的基本概念和全部内容。本题解不是把人们的精力引去设计趣味题, 在计算机上做各种高难度的游戏, 而是侧重于工程技术方面的基本技能训练和培养, 书中的趣味题都是与数值计算有关的, 曲线和图形的题目也本着这一原则, 都是围绕数值计算和加强 BASIC 和 FORTRAN 程序设计语言的基本功进行训练的, 同时也有实际的使用价值。

题解中涉及基本 BASIC 和扩展 BASIC 以及 FORTRAN IV 和少量的 FORTRAN 77 的内容, 全部 BASIC 和 FORTRAN 程序都是在 68000 微型计算机上调试通过的, 68000 微型计算机为 UNIX 操作系统。

本书在编写过程中和完稿后, 承北京航空航天大学唐荣锡教授审阅, 提出了很多宝贵的意见, 并给予很大的支持; 北京航空航天大学冯远闯同学参加了部分程序的调试; 在此致以深切的谢意。

由于本人水平有限, 书中难免有误, 敬请读者指正。

编 者

1988年1月

目 录

前言

第一章 程序设计基本知识..... 1

§1-1 BASIC 和 FORTRAN 语言..... 1

§1-2 程序设计的一般步骤..... 2

§1-3 数学模型的建立..... 4

§1-4 提高程序设计质量的几点看法..... 6

第二章 基础问题13

§2-1 基本运算.....13

题1. 求圆周长、面积和体积.....13

题2. 计算代数式的值.....14

题3. 求12个数的平均值.....17

题4. 学分、分数和平均分数的输出.....19

题5. 农贸市场税收的计算.....22

题6. 托运行李计价.....25

题7. 四则运算(1).....26

题8. 四则运算(2).....27

题9. 四则运算(3).....29

题10. 四则运算(4)30

题11. 求某个数的倒数32

题12. 求 a_i 与 b_i 乘积之和33

题13. 复数的四则运算35

题14. 摄氏温度与华氏温度的转换36

题15. 加法器运算37

题16. 十-二进制转换.....39

题17. 十-八-二进制转换42

§2-2 三角函数、长度、面积、体积计算.....44

题18. 计算级数 $\sin x$ 的近似值.....44

题19. 函数的计算47

题20. 已知两边和其夹角求第三边47

题21. 求斜面上释放质点的速度48

题22. 求渐开线上指定点的坐标51

题23. 求两点间的距离和中点坐标52

题24. 用自定义语句函数计算表达式的
值55

题25. 用子程序求 M 之值.....56

题26. 用台劳级数求 $\sin^{-1}x$ 的近似值.....59

题27. 求圆锥体的体积60

题28. 求扇形面积60

题29. 计算3个表达式的值 31

题30. 用子程序求第三边 62

题31. 求长方形面积和对角线的长度 ... 63

题32. 求三角形的面积 65

题33. 求梯形面积 67

题34. 满足条件时计算三角形的面积 ... 70

题35. 分五种情况计算三角形的面积 ... 75

§2-3 数值与积分计算..... 81

题36. 求 $N!$ 81

题37. 用子程序计算 $N!$ 82

题38. 求多项式 $\left[-\frac{1}{2} N(N+1) \right]^2$
前 N 项和 (1)..... 86

题39. 求多项式 $\left[-\frac{1}{2} N(N+1) \right]^2$
前 N 项和 (2)..... 87

题40. 求级数 $\sum_{N=0}^k (2^N)$ 的前 N 项和 ... 88

题41. 用台劳级数求 e 的值 89

题42. 求级数 $\sum_{n=1}^k \frac{1}{(2n)^3}$ 前 N 项和 90

题43. 求 $\sum_{n=1}^{10} \frac{X^{n+1}}{N!}$ 的 n 项和 92

题44. 计算人口增长年数 94

题45. 计算 $K, K^2, 1/K$ 的累加和..... 95

题46. 银行储蓄本金和利息的计算 96

题47. 表达式 R 值计算 98

题48. 工业产值翻番计算 99

题49. 定义语句函数计算表达式的值 ...100

题50. 判断与计算103

题51. 梯形法求定积分之一105

题52. 梯形法求定积分之二107

第三章 趣味题.....109

§3-1 正负、奇偶数.....109

题1. 求各负数和负数之和.....109

题2. 求奇数之和与偶数之积.....110

题3. 判断奇、偶数.....111	题10. 求矩阵对角线各元素平方之和 ...177
题4. 分开并求出奇数之和与偶数之和...113	题11. 按行选绝对值最大者180
题5. 求 a_i 与 b_i 的偶数项乘积之和.....114	题12. 将矩阵的各个元素自乘184
§3-2 最大、最小、公因、公倍数.....116	题13. 矩阵的行与列互换186
题6. 求两个数的最小公倍数.....116	题14. 矩阵转置138
题7. 求五个数的最大公因数.....117	题15. 矩阵求逆190
题8. 求最大公因数和最小公倍数.....119	题16. 有特殊要求的两矩阵相乘194
题9. 求 a_i 中最大的数120	题17. 矩阵的综合运算199
题10. 求 6 个数中最大和最小的数122	§4-3 解方程.....202
§3-3 有趣的计算.....123	题18. 解10组一元二次方程的根202
题11. 按相反的顺序输出一组数123	题19. 用子程序解一元二次方程的根 ...205
题12. 鸡兔共笼125	题20. 用牛顿迭代法求三次方程的根 ...213
题13. 百钱买百鸡126	题21. 用牛顿迭代法求五次方程的根 ...215
题14. 素年年号128	题22. 用牛顿迭代法求 $x = \sqrt{a}$ 的根...216
题15. 求 1~1000之间的素数130	题23. 用弦截法求方程的根218
题16. 求整数各位数字之和等于 5 的 数132	题24. 列主元素高斯消元法解方程组 ...220
题17. 1~1000中, 3 或 5 的倍数的 个数与和135	§4-4 分段函数.....223
题18. 填上一个正确的数.....137	题25. 分段函数(1)223
题19. 看谁算得快又对, 该数自身加、 减、乘、除之和为定数137	题26. 分段函数(2)225
题20. 伟人做过的数学题, 巧算人数 ...140	题27. 分段函数(3)227
题21. 100!和1000!末尾有多少个零.....143	题28. 分段函数(4)230
题22. 求取某个大数的最后三位数字 之和144	题29. 分段函数(5)234
题23. 求勾股数146	题30. 分段函数(6)239
题24. 买邮票, 巧安排149	题31. 分段函数(7)241
题25. 数列有趣的组合150	题32. 分段函数(8)246
题26. 哥德巴赫猜想的验证.....152	题33. 分段函数(9)250
第四章 行列式、矩阵、解方程.....155	第五章 排序、统计、数表252
§4-1 行列式.....155	§5-1 大小排序.....252
题1. 计算行列式的值.....155	题1. 从大到小排列三个数.....252
题2. 求每一行各元素之和.....157	题2. 按递增顺序排列20个数.....254
题3. 求每一列各元素平方之和.....159	题3. 考试成绩分类.....257
题4. 求各行各列元素之和.....160	题4. 最差成绩的统计.....261
§4-2 矩阵.....163	题5. 考试成绩分档统计.....263
题5. 按要求输出一个矩阵.....163	题6. 计算平均分.....266
题6. 两个矩阵相减.....165	题7. 考分最高者的序号和成绩.....267
题7. 两个矩阵相加和相减.....169	题8. 按总成绩排名次.....270
题8. 两个矩阵相乘.....171	题9. 求10名学生各门功课的平均分及 总平均分278
题9. 求矩阵主对角线各元素之和.....175	题10. 综合考核学生的学习情况280
	题11. 两组数对应排序284
	§5-2 统计.....287
	题12. 统计年龄287

题13. 发工资前的准备工作	290	题2. 打印 $\sin x$ 和 $x/3$ 的曲线	344
题14. 统计选票	296	题3. 打印 $\sin x$ 和 $2\sin(2x + 3.14159/6.0)$ 的曲线	348
题15. 简单的工资统计表	301	题4. 打印 $y = (1/3)\sin x + (1/2)\sin 2x$ 的曲线	351
题16. 到学院寻找学生	304	题5. 打印簸箕曲线 $y = 8a^3/(x^2 + 4a^2)$	359
题17. 身高统计	308	题6. 打印 $x = 1 + y(y + 1)/2$ 的曲线	363
题18. 某车间年龄构成频率分布	311	题7. 打印 $y = e^{-\frac{x^2}{225}}$ 的曲线	366
§5-3 各种数表	315	§6-2 图形	369
题19. 三角函数表	315	题8. 打印杨辉三角形(1)	369
题20. 1~20的开平方表和指数函数表	322	题9. 打印杨辉三角形(2)	371
题21. 求 1~30的 N , $1/N$ 和 $1/N^2$ 的数表	324	题10. 打印杨辉三角形(3)	373
题22. 求 1~30的常用对数和自然对数表	326	题11. 有趣的数字排列	375
题23. 求 $N = 1 \sim 20$ 的二次、三次方及开立方、开平方数表	328	题12. 打印方阵图	377
题24. 求 X^2 , X^3 , X^5 的数值	330	题13. 打印枫树叶	379
题25. 求奇数的立方根和偶数的平方根	332	题14. 打印对顶圆锥	381
题26. 打印 X , X^2 , X^3 , X^{10} , X^{16} 的表格	334	题15. 打印圆筒套圆锥	384
题27. 打印“九九”表	338	题16. 菱形与矩形的组合图形	386
题28. 编制平方根表	339	题17. 打印六角星	387
第六章 曲线和图形	342	题18. 打印梅花图	391
§6-1 曲线	342	题19. 打印枫叶组图形(1)	393
题1. 打印 $\sin x$ 和 $\cos x$ 曲线	342	题20. 打印枫叶组图形(2)	396
		题21. 打印枫叶组图形(3)	398
		参考文献	399

第一章 程序设计基本知识

§1-1 BASIC和FORTRAN语言

1. BASIC和FORTRAN语言的特点

1) BASIC语言的特点

BASIC语言是高级语言的一种。BASIC是Beginner's All-purpose Symbolic Instruction Code (初学者通用符号指令代码) 一词的缩写。BASIC语言是目前国际通用的计算机程序设计语言。小型和微型计算机一般都配有BASIC语言, 它是一种适合初学者使用而又实用的计算机高级语言。其特点是:

BASIC语言比较简单, 其命令和语句中使用的词以及运算符号与英语中使用的单词和数学中的符号差不多, 比较直观, 易于理解和记忆。如英语中“PRINT”的意思是“打印”, 在BASIC语言中它也代表“打印”, 使计算机打印出你所需要的内容。

BASIC是一种会话式的语言, 通过计算机的终端使人和计算机进行对话。当输入源程序时, 计算机能立即检查语句中的句法错误。如有错, 用户可以立即在键盘上修改程序。当修改后的程序运行时, 计算机又能查出程序中的语法错误。可以在计算上边算边改, 直到得出满意的结果为止, 十分方便和灵活。

BASIC语言适用面很广泛。它不仅适用于科技方面的计算, 而且还具有一定的数据处理的能力。特别是扩展BASIC增加了字符串处理和自选打印格式的功能, 用于小型的数据处理和事务管理方面是非常方便的。

BASIC是一种小型的算法语言, 由于它允许的数值的范围、简单变量的个数、以及数组的维数、自定义函数的自变量个数等都有一定的限制, 因此它的应用范围也受到一定的限制。但是, 一般科技计算和事务管理中的小型题目, 用BASIC语言编程是比较适宜的。

综上所述, 由于BASIC具有突出的特点, 因而被广泛应用。它虽然是一种国际通用的算法语言, 但是每一种计算机所用的BASIC语言的具体规定还会有某些细小的区别。因此在使用前, 必须查阅你所使用计算机系统的BASIC说明书。

2) FORTRAN语言的特点

FORTRAN也是目前国际上广泛流行和使用的一种高级语言。FORTRAN是Formula Translation (公式翻译) 的缩写。其特点是:

FORTRAN是一种在科学计算中使用极为广泛而且比较成熟的程序设计语言, 它的标准程序库内容非常丰富, 许多生产、科技问题都有标准程序可以利用。这对于加快工作进度和提高工作效率是十分有利的。

FORTRAN程序设计语言, 目前不再是专用于科学技术中的程序设计语言, 愈来愈多的工商企业也使用FORTRAN语言, 因为他们认识到用FORTRAN语言设计的程序, 既简单效率又高。

FORTRAN的词汇仅由几十个词组成。如果知道句型的话, 这些词也可以通过近10种句型得到。虽然FORTRAN语言是比较简单的, 但它是精确的, 在科学计算中发挥了很大的

作用。它既可以计算很小的题目，又可以计算大型的题目，比 BASIC 语言更受科技工作者的欢迎。

在设计好一个 FORTRAN 源程序以后，还必须经过编辑、翻译、连接和运行等几个步骤，比运行一个 BASIC 程序要复杂一些，即用 FORTRAN 语言上计算机解题的指令要多一些。

FORTRAN 程序设计语言还具有标准化程度高的优点，这个优点使得程序容易移植。就是说，程序可以适用于不同的计算机。或者说，对于不同的计算机具有通用性。已编写好的源程序几乎可以不加修改（有时需要作一些微小的修改）就能在不同型号的计算机上实现。应该指出的是，在具体的计算机上解题时，源程序总是要根据计算机的具体情况作一些小的修改或扩充。在上机前用户应查阅你所使用的计算机的 FORTRAN 使用说明书。

2. BASIC 和 FORTRAN 语言的主要区别

从 BASIC 和 FORTRAN 语言的特点可以看出，这两种语言的主要区别在于：BASIC 语言适合初学者的数值计算和事务管理的中、小型题目，它是一种对话式的语言，用户可以通过显示屏幕与计算机对话，彼此提出问题和回答问题。这种“会话”功能对于用户很有益处，当源程序输入计算机以后，若有错误，用户可以立即通过键盘修改程序，边算边改，直到得出满意的结果，BASIC 操作系统的“热情”服务，大大地吸引了广大用户。FORTRAN 语言基本语法的难度比 BASIC 大，但适用面比 BASIC 大，适合于处理中、大型的科学计算和事务管理的题目。

BASIC 与 FORTRAN 程序设计语言有不同的“单词”与“语法”，如 BASIC 语言中的“单词”LET、PRINT 等，FORTRAN 中的 READ、WRITE 等；它们还有自己的“语法”，如 BASIC 和 FORTRAN 语言中的一些规定。我们使用的汉语天天讲，我们学习英语时，对英语的单词和句型要天天记，要常用……。学习 BASIC 和 FORTRAN 程序设计语言也要记，但死记硬背 BASIC 和 FORTRAN 程序设计语言的“单词”和“语法”的各种规定，既枯燥无味，也容易忘记。如果掌握了其特点及程序设计的基本方法后，即使一些具体规定忘记了，只要查一查书马上可以解决问题。

BASIC 和 FORTRAN 程序设计语言是一门实践性很强的课程。要学好和掌握它，只有多动手编写程序并上机调试，才便于学习和掌握。

BASIC 和 FORTRAN 程序设计入门并不难，但是没有足够多的例子还是不能很好理解和消化的，况且，一个好的例子胜过一百句空话，它往往能给人以有益的启示，启迪思路去解决难度大且比较复杂的问题。

§1-2 程序设计的一般步骤

程序设计应当包括接受任务、分析问题，一直到最后通过计算机运行获得正确无误的结果的全过程，一般有如下几个步骤：

1. 明确要解决的问题。在接受任务时应确切地弄清楚问题的性质、任务和要求。例如，首先要知道所要解决的问题的初始条件是什么，需要什么，要求获得什么样的结果，对输出打印格式有什么要求。另外还应判断是否有必要在计算机上解此问题等等。看起来似乎很简单，然而这却是很重要的一步。所以在提出问题和接受任务的过程中，往往需要

反复磋商多次才能最后确定，以便得到一个比较满意的方案。

有的同志往往拿到题目后，不加以深入的思索就动手设计程序，到最后才发现未能满足要求而不得不多次返工修改，这一点是从事程序设计的同志所切忌的。

2. 分析问题、建立数学模型。这一步是把自然界和社会生活中的现象经过一定的简化和提炼，用数学语言来描述的过程。如把一个物理问题用一个联立方程来表示。

3. 选择适当的计算方法。根据不同的计算方法所编的程序运行时间和计算误差相差十分悬殊。一些常见的科技问题（如微分方程、积分、矩阵运算……）都已积累了成熟的算法，并编好了 BASIC 或 FORTRAN 子程序。通过查阅有关算法汇编，可得到现成的算法和程序，只要代入具体数据即可。一般的计算机内存中都有程序库，使用时按名字调用该子程序即可。另一方面，在建立了数学模型之后，找出用计算机解题的近似方法。例如求一个函数的定积分，用计算机解此问题不同于纯数学的方法。计算机是通过数值计算来求一个数值解，而这个解往往是近似解。求定积分可以用矩形法、梯形法和辛甫生法。每一种方法的近似程度不同，所花费的计算时间不同，要根据所要解决问题的要求选择合适的方法。

4. 根据已定的计算方法画出流程图（即框图）。确定了问题的计算方法以后就要画流程图，为了对程序的执行过程有一个更清楚的了解，常常用流程图作分析程序的辅助手段。所谓流程图是计算方案的形象描述，比较直观。它以一个框代表一段程序（包括一个或多个语句），以箭头表示程序的流向。在编制程序时，常用框图帮助我们进行程序设计或介绍程序的结构。有了流程图，即使对比较复杂的程序，也能做到思路清楚，层次分明，从而减少编程中的错误，或者能较快地搞清楚编程思路。只要算法和流程图是正确的，程序设计就不会出现大的逻辑性错误。

5. 根据流程图编写源程序。将流程图中各基本结构代以具体的高级语言（BASIC 或 FORTRAN），按照严谨的规格化的方式写出所编写的源程序。写源程序时应注意以下几点：

1) 充分使用注释行，对程序的整体说明（题目）、各逻辑段落的说明都应该加注释行。在适当的地方加空注释行，便于阅读。

2) 适当加括号（对于表达式），不能肯定加括号会不会出错时最好加上，但不要加得太多，括号层次太多也容易出错。如 $\frac{a}{bc}$ 若写成 $A/B * C$ 就错了，而应写成 $A/(B * C)$ ；如 $\frac{a}{-b}c$ ，若写成 $A/-B * C$ 就错了，应写成 $A/(-B) * C$ 。

3) 用 READ 读数据，如果数据量大或数据有一定的结构要求时（如矩阵），此时读入的数应马上用写语句（WRITE 或 PRINT）按容易检查的格式打印输出，可以避免读入数据时出错。

4) 可用“ERR = ...”来检查输入数据中的一些错误。用限制输入数据数值的办法来检查另一类错误（例如分数小于 0，大于 100 是错的等等）。

5) 程序中所有的输出打印结果都应该有标题（表头）或标识部。如表头：

X	Y	Z
:	:	:

或标识部：

X = ...	Y = ...	Z = ...
X = ...	Y = ...	Z = ...

这样输出的计算结果整齐、醒目、直观，提高了程序

的可读性。

6) 程序中在可能的情况下, 尽量用 IF 条件转移语句代替 GOTO n 无条件转移语句, 并注意控制转移语句的转移方式。避免程序转来转去而使人很难读懂, 以至出错。

7) 行数最少的程序, 不一定是最好的程序。

6. 将写好的源程序输入给计算机进行调试、检测和修改。一个大型且比较复杂的程序往往不是一次能成功的, 需要多次调试、修正、更改源程序, 先用调试数据取得正确的结果以后, 再将实际计算问题的数据带入正式运行。

7. 整理资料写出报告。检查所得的计算结果是否全部满足预定的要求。如果符合要求, 则整理结果, 写出程序说明书。如: 题目任务要求; 数学模型和计算方法; 流程图; 源程序清单; 输入数据; 运行结果等。一个好的有使用价值的程序往往不是只用一次就完了, 它可以多次被使用或提供给其他人使用。有了程序说明书才能帮助别人理解和使用该程序。当然对于小的程序可不必有这样的要求, 但是即使是初学者也应当养成保存资料和分析结果的良好习惯。

图 1-1 示出程序设计步骤的流程。

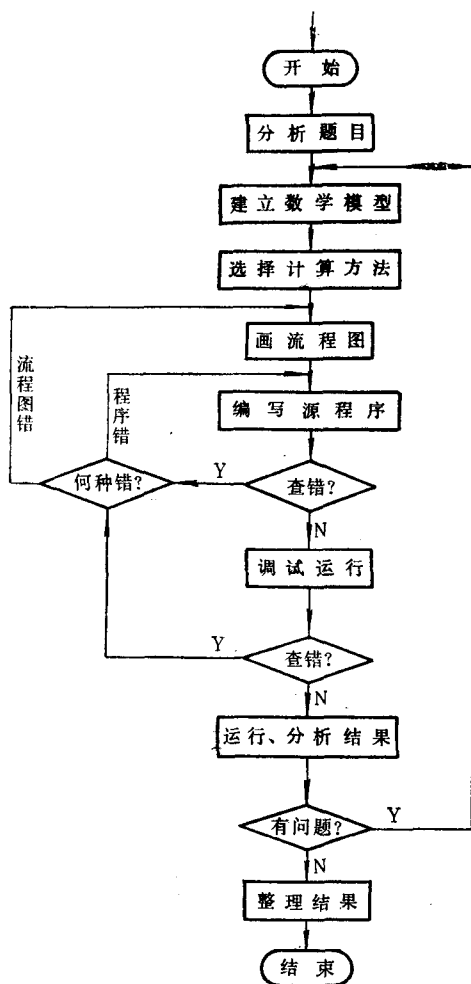


图 1-1

§1-3 数学模型的建立

1. 问题提出

用计算机算题必须进行程序设计。为了解决某一个具体问题, 人们用计算机能够接受的程序设计语言来描述给定的数据及对数据处理的过程。其中 BASIC 程序是按照 BASIC 程序设计语言的规定编写的程序, FORTRAN 程序是按照 FORTRAN 程序设计语言的规定编写的程序。不管是 BASIC 还是 FORTRAN 程序的编写, 都要根据给定的具体问题, 进行分析研究, 找出其中内在的联系, 去掉无用的内容, 确定解题的数学模型 (对于数值型计算)。由数学模型进一步确定具体的解法, 画出流程图, 写出源程序。数学模型建立的正确与否, 是关系到程序计算结果的关键。

2. 如何从工程实际问题中提炼出正确的数学模型

工程实际问题往往是比较复杂的。对于工程实际问题的数值计算, 有时不能一眼就看出它的数学模型。当拿到问题以后, 要进行分析研究, 简化结构, 确定边界条件, 构造出数学

模型。数学模型的构造不是靠主观想象的，而是把自然界和社会生活中的现象经过一定的简化和抽象，才能建立起数学模型。由于数学模型直接影响计算的质量，故要求所建立的数学模型应易于在计算机中实现。

3. 举例

例1 已知一个三角形三条边 a 、 b 、 c 的长度，求三角形面积。

(1) 任务是求三角形的面积公式。

(2) 几种方案可求三角形面积

1) 已知三角形的底边长 a 和高 h ，则三角形的面积公式为

$$S = \frac{1}{2} ah$$

2) 已知三角形的两个角 α_1 和 α_2 与两个角所夹的边长 a ，则三角形的面积公式为

$$S = \frac{1}{2} a^2 \frac{\sin \alpha_1 \sin \alpha_2}{\sin(\alpha_1 + \alpha_2)}$$

3) 已知三角形的三条边长 a 、 b 、 c ，则三角形的面积可用海伦公式求得，即

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

式中 $p = \frac{1}{2}(a+b+c)$

4) 已知三角形的两个边长 a 、 b 和它们的夹角 α ，求三角形面积时，可用公式 $c = \sqrt{a^2 + b^2 - 2ab\cos\alpha}$ 求出第三条边的长度 c ，然后利用海伦公式求出三角形的面积。

从上面的分析可以看出，根据已知条件的不同，可以用四种方法求出三角形的面积。根据本题的已知条件，最后确定解此问题的数学模型为海伦公式。

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

式中 $p = \frac{1}{2}(a+b+c)$

例2 百钱买百鸡的问题。若鸡翁一，值钱五；鸡母一，值钱三；鸡雏三，值钱一，以百钱买百鸡，问鸡翁、母、雏各几何？

题中所给出的条件是用100元钱买100只鸡和公鸡每只5元，母鸡每只3元，小鸡每3只1元。要解此题没有现成的数学公式，应根据已知的条件来解。先设应买公鸡为 x 只，母鸡为 y 只，小鸡为 z 只，已知总鸡数为100只，就得出第一个方程 $x+y+z=100$ ，由于已知每一只的钱数和总钱数，就确定了第二个方程 $5x+3y+\frac{1}{3}z=100$ 。由此建立了所需的数学模型：

$$\begin{cases} x+y+z=100 \\ 5x+3y+\frac{1}{3}z=100 \end{cases}$$

例3 研究车轮缘上某点的运动。

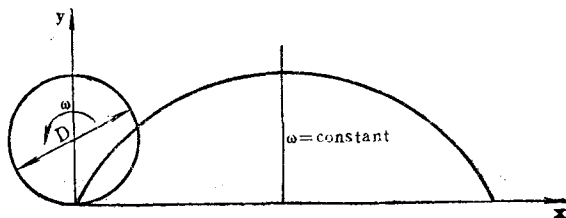
为了解决这个问题先利用平面几何描绘出车轮缘上某点的运动轨迹，见图1-2。这就是数学模型1。如果继续用几何方法求解（图解法），当然可以，但改用解析几何（模型2）

的方法求解这个问题就方便得多。利用解析几何可以把研究对象简化为有序实数系的代数问题，可以列出参数方程（数学模型3）。

$$\begin{cases} x = (\theta - \sin\theta)D/2 \\ y = (1 - \cos\theta)D/2 \end{cases}$$

将参数方程求导，可得出 x 和 y 方向的运动速度，就是微积分公式（数学模型4）：

$$\begin{cases} v_x = (\omega D/2)(1 - \cos\theta) \\ v_y = (\omega D/2)\sin\theta \\ \omega = \frac{d\theta}{dt} = \text{const. (常数)} \end{cases}$$



最后我们还得用到矢量，求得合成速度为

$$v = \sqrt{v_x^2 + v_y^2} = \omega D \sin \frac{\theta}{2}$$

图 1-2

这样，还可得到 $t-v$, $t-s$ 的关系，也可利用模型1画出结果。

从此例可以看出，一个简单的问题就用到4种数学模型。

从以上3个建立数学模型的例中不难看出，要建立一个正确的数学模型，首先要把所给的问题弄清楚，已知条件是什么，要求什么，搞清实际问题的来龙去脉，将所给的问题进行简化，找出解决问题的主要逻辑关系，再将特定的场合推广到一般。经反复推敲，即可确定解决问题的数学模型。

工程实际中的问题有数值型的也有非数值型的，在确定了数学模型以后，都要找出计算和求解此问题的具体方法，也就是问题的解法。正确的解法是使数学模型得到实施的重要保证。

§1-4 提高程序设计质量的几点看法

在电子数字计算机日益广泛地应用于各个领域的今天，对于程序设计质量的要求也在不断提高。除了数学模型和算法正确这一前提外，还要注意程序结构清楚、易读性好和效率高等方面。

1. 节约时间，减少操作次数

1) 选用最快的操作，减少重复计算。

在编写程序时，应尽量选择节省机器时间的具体算法。

据统计，计算机处理不同运算的时间比例如表1-1

从表1-1中可以看出，在小型计算机上，进行一次乘法所需要的时间是进行一次加减法所需时间的8倍，除法则为20倍。因此在编写程序特别是较大的程序时，应考虑如何提高运算速度的问题。

如： $y = 2x$ ，写成 $Y = X + X$ 比写成 $Y = 2.0 * X$ 要快8倍。因为计算机作加减法快于乘除法，作乘法快于乘幂。例如 $y = x^3$ ，写成 $Y = X * X * X$ 比写成 $Y = X ** 3$ 要快。遇

表1-1 计算机处理不同运算的时间比

机 种 \ 运 算	加 减 法	乘 法	除 法
大 型 机	1	2	3
中 型 机	1	4	8
小 型 机	1	8	20

到作 $\sqrt{X}$ 的运算,用 $X * 0.5$ 就不如用基本函数 $SQRT(X)$ (或 $SQR(X)$),因为在计算机内 $X * 0.5$ 通常作 $EXP(0.5 * ALOG(X))$ 处理,要两次用到基本函数。

如: $y = 3.1x^4 + 1.4x^3 - 5.0x^2 + 6.2x - 3.0$ 写成 $Y = (((3.1 * X + 1.4) * X - 5.0) * X + 6.2) * X - 3.0$ 只有4次乘法。写成 $Y = 3.1 * X * * 4 + 1.4 * X * * 3 - 5.0 * X * * 2 + 6.2 * X - 3.0$ 共需要做13次乘法,比上面的写法多3倍的时间。

又如: 计算 $M = 1 + 2 + 2^2 + \dots + 2^{63}$

程序 1

$M = 1.0$

DO 10 I=1,63

T=2.0 * * I

10 M=M+T

STOP

END

程序 2

T=1.0

M=1.0

DO 10 I=1,63

T=T*2.0

10 M=M+T

STOP

END

显然程序 2 比程序 1 大大节省时间。因为程序 1 在计算第 I 项时是计算 2 的 I 次方,而程序 2 只需将前一次的 T 值再乘以 2 即可。乘 I 次方的运算比做乘法花的时间多。

2) 进行乘幂运算时,指数若能写成整数决不能写成实型数。如 $Y = A^5$, 应写成 $Y = A * * 5$ 而千万不要写成 $Y = A * * 5.0$ 。因为后者比前者要多花几十倍的时间,精度也差。

3) 程序中多次重复使用的表达式,最好用一个中间变量寄存,在需要的地方写上中间变量名即可,不必每次重复计算。

如求一元二次方程的根:

100 ROOT1=(-B+SQR(D))/(2.0*A)

110 ROOT2=(-B-SQR(D))/(2.0*A)

不如写成

```

80 E = SQR(D)
90 F = 1.0/(2.0 * A)
100 ROOT1 = (-B + E) * F
110 ROOT2 = (-B - E) * F

```

4) 程序中多次用到某些常数, 不必每次重复写出, 减少重复计算。可将其赋给一个变量, 然后引用。如 π , $\sqrt{2}$, $\sqrt{3}$, 应写成 $PI = 3.1415926$, $A1 = 1.414213$, $A2 = 1.73$, 或者用参数语句说明。如 `PARAMETER PI = 3.1415926, A1 = 1.414213, A2 = 1.7321`。此后程序段中出现这些常数时用上变量名代替, 可省去一批赋值语句。

5) 在循环体内数值不变化的部分, 应提到循环体外进行计算。如:

```

50 A = X * X + 2 * X
60 FOR I = 1 TO 100
70 Y = 0.1 * I
80 S = S + SQR(A) * X ^ I
90 NEXT I

```

其中要计算100次 $SQR(A)$, 但其实 $SQR(A)$ 是不变化的, 故写成下面的形式可以节省几十倍的时间。

```

40 A = X * X + 2 * X
50 B = SQR(A)
60 FOR I = 1 TO 100
70 Y = 0.1 * I
80 S = S + B * X ^ I
90 NEXT I

```

6) 在循环体内计算或迭代计算时, 为防止程序运行时出现死循环或者结果发散无法停机, 需要在循环体内的适当地方加上循环次数或迭代次数的限制。

如:

```

DO 100 I = 1, N2
3 A = B(I) * * I
Y(I) = X(I) * * 2
IF(A) 1, 1, 3
1 ...
100 CONTINUE

```

此程序当① $B(I) < 0$ 及 I 为偶数时, ② $B(I) > 0$ 时, 出现死循环。应写成

```

DO 100 I = 1, N2
3 A = B(I) * * I
IF(A) 1, 1, 100
1 ...
100 CONTINUE

```

7) 能够用循环增量法产生的数据, 就不用读语句赋值, 这样既节省机时, 又不容易出错。

如第四章的题31, $R = 55, 60, 65, \dots, 200$, 计算分支函数 S 的值, 程序写成

```
DIMENSION R(20)
READ(*, 10)R
10 FORMAT(10F6.1/10F6.1)
```

需要用两个数据记录输入 R 数组各元素的值, 显然不好。下面采用循环增量法产生的数据具有更大的优势。

```
R = 55.0
10 IF(R. LT. 140) S = 17000.0 - 0.485 * R * * 2
:
R = R + 5.0
IF(R. LE .200.0) GOTO 10
```

8) 查找下标变量耗费机时, 应尽量避免重复的下标寻找。

$$\text{如计算 } x = \frac{B_1 + B_2}{B_1^2 + B_2^2}, \quad y = \frac{B_1 - B_2}{B_1^2 + B_2^2}$$

写成

```
X = (B(1) + B(2)) / (B(1) * B(1) + B(2) * B(2))
Y = (B(1) - B(2)) / (B(1) * B(1) + B(2) * B(2))
```

计算速度不如写成下面形式快。

```
P = B(1)
Q = B(2)
U = B(1) * B(1) + B(2) * B(2)
X = (P + Q) / U
Y = (P - Q) / U
```

9) 计算机处理一维数组比多维数组快。如要求把一个 10×10 的数组中所有负元素变成正元素, 则写成:

```
DIMENSION A(10, 10)
DO 10 I = 1, 10
DO 10 J = 1, 10
10 IF(A(I, J).LT.0.0) A(I, J) = -A(I, J)
```

为了节约机时, 将二维数组等价于一维数 $B(100)$, 并对数组 B 进行运算。写成下列形式:

```
DIMENSION A(10, 10), B(100)
EQUIVALENCE (A(1, 1), B(1))
DO 20 N = 1, 100
20 IF(B(N).LT.0.0) B(N) = -B(N)
```

10) 应尽量避免不必要的不同类型变量的混合运算。虽然计算机允许不同类型变量进行混合运算, 但在运算中, 是将阶次低的变量升高阶次的变量后才能混合运算, 这种类型转

换是要花机时的。

如: $4 * A * 16$ 应写成 $4.0 * A * 16.0$

11)调用子程序时(FORTRAN)实参虽规定可以是数组元素,但最好用简单变量为好,这样可以简化目标程序。

如: `CALL RR(D(I), X)`

可改写成 `A = D(I)`

`CALL RR(A, X)`

12)计算机与外存储器交换信息占的时间长,因此应尽量减少与外存储器的信息交换,实在必要时应集中成批交换。

13)一般打印输出是在行式打印机上进行的,有些中间结果可以边计算边输出,以节约机时。这条内容和第12条所说的相矛盾,因为这里谈的问题不是孤立的,要全面考虑选择自己的做法。

14)运算精度高是用计算机做题的一大优点,它可保持小数点以后有效数字十几位。对于精度,其一它取决于内存单元含有多少个二进制的位数,如32位机就能保证小数点后7位有效数字。其二选择精确度高的计算方法可以提高精度。其三如果需要,可以选用双精度常数或变量;一个双精度常数占2个内存单元,有效数字可达16位,对科学计算中的题目已经是足够了。其四在计算机内存中整型数是精确值,实型数是近似值;在精确运算时,如果参加运算的都是整数时,应尽可能用整型数表示;只有非整数的或整数的值超过计算机能表示的范围,或不同类型的数进行混合运算时,才写成实型数形式。其五对实型数运算的精度可以用条件语句中给定的误差值是否满足来控制,例如用牛顿迭代法解方程的根当给出的条件为 $|X_{n+1} - X_n| \leq 10^{-6}$ 时就停机,后次求出的根与前次根的差值 $\leq 10^{-6}$ 就满足了精度要求;如果精度要求不高可以写成 $|X_{n+1} - X_n| \leq 10^{-4}$,要求高可以写成 $|X_{n+1} - X_n| \leq 10^{-7}$ 等。

2. 节省计算机内存

计算机的内存有限,而一些较大的程序往往需要占用较大的内存容量,节省内存的目的是为了利用本单位现有的计算机,用小机器算较大的题目。内存容量紧缺的矛盾主要在FORTRAN程序设计语言显得比较突出。减少内存容量的主要方法有:

1)尽量减少源程序的长度。如用DATA语句给变量赋值而不用赋值语句。如

`A = 1.0`

`B = 2.0`

`C = 3.0`

`D = 4.0`

`E = 5.0`

可以写成

`DATA A, B, C, D, E/1.0, 2.0, 3.0, 4.0, 5.0/`

2)程序太长时,可以将程序分段,利用外存实现程序覆盖。急需的程序调入内存中运行,不急需的程序段放在外存设备中,当需要的时候调入内存,在某一适当的位置上覆盖原有不用的程序。这种操作用得比较多,是由操作员来实现的。

3)利用调外存的办法节约使用内存单元,将某些数据放在外存(磁盘、磁带)中,在程