

高等院校计算机基础
教育系列课程教材

计算机数值算法 及程序设计

周 煦 主编
王杰臣 主审



中国科学技术出版社

17
03

高等院校计算机基础教育系列课程教材

计算机数值算法及程序设计

周 煦 肖建华 编著
王凤儒 张金良

周 煦 主编

王杰臣 主审

中国科学技术出版社

• 北京 •

图书在版编目 (CIP) 数据

计算机数值算法及程序设计/周煦主编. —北京: 中国科学技术出版社, 1997
高等院校计算机基础教育系列课程教材
ISBN 7-5046-2195-1

I. 计… II. 周… III. ①电子计算机-数值计算-算法理论-高等学校-教材②程序设计-高等学校-教材 IV. TP3

中国版本图书馆 CIP 数据核字 (97) 第 09547 号

中国科学技术出版社出版

北京海淀区白石桥路 32 号 邮政编码: 100081

新华书店北京发行所发行 各地新华书店经售

中国文联印刷厂印刷

*

开本: 787 毫米×1092 毫米 1/16 印张: 17.125 字数: 425 千字

1997 年 9 月第 1 版 1997 年 9 月第 1 次印刷

印数: 1—5000 册 定价: 21.00 元

内 容 提 要

本书试图将数值计算方法与程序设计方法学两者结合起来介绍常用算法及其程序。

作为教材,本书不仅简要地介绍数值计算方法和程序设计方法学的基本知识,而且以讲透算法和结构化程序设计方法为基本出发点,详细地讲述插值、数值积分、常微分方程数值解、方程求根、线性方程组的数值解等常用算法的数学原理、计算公式和算法的分析和设计,以结构化程序设计的方法设计程序流程框图,并对所给出的 FORTRAN77 源程序进行了必要的分析和说明,因而将数值计算方法与程序设计、数学原理与使用计算机解算具体问题有机地结合了起来。本书所介绍的算法具有代表性和实用性。

本书每章开头有内容提要,结束时有小结,并附有习题,既便于教学,又适于自学。

本书可作为工科普通高等学校一般专业学生进行数值计算方法和程序设计教学的教材,亦可作为科技工作者及其他读者的自学参考用书。

责任编辑 张秀智 胡 萍

封面设计 炎 尘

责任校对 孟华英

责任印制 王 沛

机械工业部部属高校计算机基础教育
系列课程教材编审委员会

主任委员 张奠成

副主任委员 袁鹤龄 王杰臣 张明毫

委 员 郝忠孝 梁文林 朱逸芬
贝嘉祥 王肇荣 田瑞庭
陈金华

机械工业部部属高校计算机基础教育 系列课程教材出版说明

《中国教育改革和发展纲要》提出：“要按照现代科学技术文化发展的新成果和社会主义现代化建设实际需要，更新教学内容，调整课程结构。”人类即将进入的 21 世纪，将是高度信息化的社会，作为这种社会的重要基础的计算机，将渗透到社会各个角落。计算机不仅作为一种工具来使用，而且作为一种文化来普及；计算机科学技术不仅是一门独立的学科，而且是所有学科知识结构中的重要组成部分。

多年来，数、理、化及外语等基础教育在我国高等技术人才的素质培养中发挥了十分重要的作用。面对新的形势，加强高校非计算机专业的计算机基础教育，将是促进高校教育质量提高的必要措施。在这方面，除了设备配置、师资培养、计划安排和课程设置等工作外，出好教材是加强计算机基础教育的基本建设。为此，机械工业部部属高校在总结多年来实施非计算机专业计算机基础教育教学经验的基础上，发挥知识群体优势，组织编写本系列教材，供有关专业选择使用。

根据目前各专业课程设置，本系列教材包括 3 个层次：即计算机基础及高级程序设计语言，微型计算机原理及应用，软件技术基础及计算机网络概论。考虑到不同专业对计算机知识要求的差异，部分教材将按不同学时编写。由于计算机技术发展迅速，教材内容将不断更新，因此第一轮教材安排在 1996 年秋全部出齐，以便在使用过程中修改完善，使第二版教材以更新和更好的面目问世。

本系列教材除适合于高校非计算机专业本科生、大专生使用外，也可用于成人教育及自学教育教学，同时也可作为工程技术人员自修计算机技术的指导参考。

机械工业部部属高校计算机基础
教育系列课程教材编审委员会

1995 年 6 月 5 日

前 言

计算机技术的应用,已成为现代科学技术和生产力发展的重要标志之一,也是当代大学生知识能力结构的重要组成部分。为适应社会发展,迎接新技术革命的挑战,国家教委拟定了面向 21 世纪教育改革计划,其中非计算机专业的计算机基础教育列为重点项目之一。为此,1994 年 8 月,国家教委高等工科计算机基础课程教学指导委员会提出“从确定知识单元入手,带动计算机基础课进行深层次改革的建议”,明确了计算机基础教育的性质,确定了按主题领域划分的知识单元,提出了推荐课程。

为了促进机械部部属高校非计算机专业计算机基础教育的发展,1994 年 11 月,机械工业部教育司在合肥工业大学召开了部属高校计算机基础教育研讨会,决定从课程教材建设入手,规范教学内容,编写非计算机专业计算机基础系列课程教材,并成立了教材编审委员会。

本教材是非计算机专业计算机基础系列课程教材之一,是依据 1995 年 3 月部属高校计算机基础教育系列课程教材编审委员会通过的大纲进行编写的。

使用计算机进行科学计算时,人们最为关心的是计算机是否能够完成预期的计算任务。当前的计算机,就其本身运算的实质而言,它所能进行的运算,仅仅是加、减、乘、除四则运算与逻辑运算,而一般的数学解析方法,例如求函数的导数、积分函数的原函数、微分方程的通解等,这些都是计算机无能为力的。但是,如果所求的是近似的数值解,那末计算机就大有用武之地,而且一般均可以满足预先提出的计算精度的要求。这正是本教材——《计算机数值算法及程序设计》所要完成的任务。

数值计算方法是一门古老的学科,而同时又是一门新兴的学科。电子计算机的产生和发展大大地促进了数值计算方法的发展,现在数值计算方法已深入到各个学术领域;程序设计方法学是新近产生和发展起来的一门科学,只有把数值计算方法和程序设计紧密地结合起来,把算法变为计算机能直接执行的程序,才能真正使用计算机帮助人们解决各种极其复杂的计算任务。目前,以上两门科学各自都有许多专著和教材,但将两者结合起来介绍尚不多见,本教材试图将数值计算方法和程序设计方法学融为一体,这是一种尝试。

本教材使用计算功能强的 FORTRAN 语言编写出各种常用算法的程序,这些程序都在计算机上调试通过,只要有适当的环境,就可以直接在计算机上用它们来完成某些计算任务。

本教材在介绍各种算法时,大体上按照基本思想、计算公式、算法设计、程序框图设计、源程序和误差的顺序进行叙述,全书以讲透算法为主线,突出结构化、模块化设计思想,详细介绍“自顶向下,逐步细化”和典型模块设计的过程;概念明确,条理清楚,由浅入深,循序渐进,语言流畅,图文并茂,既适于教学作教材使用,又适于一般读者进行自学。本教材可作为一般高等院校的本科生、专科生、研究生、成人教育的教材或教学参考用书,亦可供广大科技工作者自学。

本教材共分 6 章,其中第一章、第三章、第四章由合肥工业大学周煦老师撰稿,第二章由湘潭高等机电专科学校肖建华老师撰稿,第五章由哈尔滨理工大学王凤儒老师撰稿,第六

章由洛阳工学院张金良老师撰稿。全书由周煦老师主编和统稿，由哈尔滨理工大学王杰臣教授主审。

在本教材的编写过程中，自始至终得到机械工业部教育司、机械工业部非计算机专业计算机基础教育研究会、安徽省教育委员会的关怀和指导；得到合肥工业大学、哈尔滨理工大学、洛阳工学院、湘潭机电高等专科学校及有关兄弟院校的支持和帮助；机械工业部教育司张明毫、王清池、李春江同志和中国科学技术出版社张秀智副总编为组织和指导本教材的编写出版进行了大量的卓有成效的工作；王杰臣教授、合肥工业大学孙家启教授和于红光老师、湘潭机电高等专科学校湛新年副教授对本教材的内容和编写提出过许多指导性的意见。对本教材的编写、出版有帮助的部门、单位和同志还有很多，在此不一一列举，一并借本教材出版的机会，谨向上述部门、单位和老师，向支持本教材编写出版的同志们，表示衷心的感谢。

由于编者的水平有限，加上编写匆忙，难免挂一漏万，恳请读者予以批评指正。

编 者

1996年10月

目 录

第一章 绪论	(1)
1 课程性质及其研究对象	(1)
1.1 课程研究对象	(1)
1.2 学习计算机数值方法及程序的重要性	(2)
1.3 课程基本要求	(4)
2 数值计算方法的基本方法与途径	(4)
2.1 离散变量与离散化	(4)
2.2 逼近	(5)
2.3 递推	(5)
2.4 常用基本递推结构(基本算法小模块)	(7)
2.5 算法的特点	(10)
3 误差	(11)
3.1 误差	(11)
3.2 误差限	(12)
3.3 相对误差	(12)
3.4 有效数字	(13)
3.5 误差来源	(15)
3.6 应用电子计算机进行数值计算时应注意的问题	(18)
4 程序设计方法简介	(21)
4.1 概述	(21)
4.2 程序结构方面的初步知识	(34)
4.3 程序设计方法简介	(36)
小结	(43)
习题一	(44)
第二章 插值	(46)
1 拉格朗日插值	(47)
1.1 概述	(47)
1.2 线性插值	(47)
1.3 抛物插值	(49)
1.4 一般形式拉格朗日插值	(50)
2 插值余项	(54)
2.1 拉格朗日插值余项定理	(54)
2.2 插值余项的事后估计	(55)
3 分段插值	(57)
3.1 基本思想	(57)

3.2	插值结点选择原则	(57)
3.3	分段线性插值	(57)
3.4	分段抛物插值	(60)
4	牛顿 (Newton) 插值	(64)
4.1	差商	(64)
4.2	差商的性质	(65)
4.3	差商的计算	(65)
4.4	牛顿插值多项式	(67)
4.5	牛顿插值的算法设计	(69)
4.6	牛顿插值的程序框图设计	(70)
4.7	牛顿插值的 FORTRAN 源程序	(71)
4.8	牛顿插值的误差	(73)
5	等距节点插值	(74)
5.1	差分及其性质	(74)
5.2	等距节点插值公式	(75)
5.3	向前差分递推表	(76)
5.4	等距结点插值的算法设计	(76)
5.5	等距结点插值的程序框图设计	(78)
5.6	等距结点插值的 FORTRAN 源程序	(79)
6	曲线拟合	(81)
6.1	概述	(81)
6.2	线性最小二乘	(84)
	小结	(94)
	习题二	(94)
第三章	积分的数值方法	(97)
1	概述	(97)
2	梯形积分法	(99)
2.1	方法概述	(99)
2.2	定步长梯形积分	(100)
2.3	变步长梯形积分	(106)
3	抛物积分法	(111)
3.1	方法概述	(111)
3.2	定步长抛物积分	(113)
3.3	变步长抛物积分	(117)
4	龙贝格积分法	(122)
4.1	牛顿-柯特斯积分	(122)
4.2	梯形和抛物积分法的误差	(126)
4.3	龙贝格求积公式	(131)
4.4	龙贝格积分的算法设计	(133)
4.5	龙贝格积分的程序框图设计	(134)

4.6 龙贝格积分的 FORTRAN 源程序	(134)
5 高斯 (Gauss) 求积	(137)
5.1 引言	(137)
5.2 高斯积分的提出	(137)
5.3 高斯积分法求积过程	(139)
5.4 变步长高斯求积	(141)
小结	(149)
习题三	(149)
第四章 常微分方程数值解法	(152)
1 概述	(152)
1.1 研究常微分方程数值解法的必要性	(152)
1.2 一阶常微分方程的初值问题	(152)
1.3 常微分方程初值问题数值解法的一般方法	(152)
2 欧拉折线法和改进的欧拉折线法	(153)
2.1 欧拉折线法	(153)
2.2 改进欧拉折线法	(158)
3 龙格—库塔法	(162)
3.1 概述	(162)
3.2 龙格—库塔法的基本思想	(164)
3.3 龙格—库塔法的计算公式	(164)
3.4 龙格—库塔法的算法设计	(165)
3.5 龙格—库塔法的程序框图设计	(165)
3.6 龙格—库塔法的 FORTRAN 源程序	(166)
3.7 龙格—库塔法的误差	(167)
4 一阶微分方程组与高阶常微分方程初值问题的数值解法	(168)
4.1 一阶微分方程组初值问题的数值解法	(168)
4.2 高阶常微分方程初值问题的数值解法	(172)
小结	(177)
习题四	(178)
第五章 方程求根	(179)
1 二分法	(179)
1.1 有根区间的确定	(179)
1.2 二分法求根	(182)
2 迭代法	(187)
2.1 迭代法的基本思想	(187)
2.2 迭代法的数学原理	(187)
2.3 迭代法的算法设计	(187)
2.4 迭代法的程序框图设计	(188)
2.5 迭代法的 FORTRAN 源程序	(188)
2.6 迭代格式的收敛问题	(190)

3 加速迭代法	(192)
3.1 加速迭代法的基本思想	(192)
3.2 加速迭代法的数学原理	(192)
3.3 加速迭代法的算法设计	(194)
3.4 加速迭代法的程序框图设计	(194)
3.5 加速迭代法的 FORTRAN 源程序	(194)
4 牛顿法	(197)
4.1 牛顿法的基本思想	(197)
4.2 牛顿法的数学原理	(197)
4.3 牛顿法的算法设计	(198)
4.4 牛顿法的程序框图设计	(199)
4.5 牛顿法的 FORTRAN 源程序	(199)
4.6 牛顿法的收敛问题	(201)
5 弦截法	(202)
5.1 弦截法的基本思想	(203)
5.2 弦截法的数学原理	(203)
5.3 弦截法的算法设计	(204)
5.4 弦截法的程序框图设计	(204)
5.5 弦截法的 FORTRAN 源程序	(205)
小结	(207)
习题五	(207)
第六章 线性方程组的数值解法	(209)
1 迭代法	(209)
1.1 迭代法的基本思想	(210)
1.2 迭代法的计算公式	(210)
1.3 迭代法的算法设计	(212)
1.4 迭代法的程序框图设计	(214)
1.5 迭代法的 FORTRAN 源程序	(214)
1.6 判断迭代法收敛的几个常用条件	(216)
2 约当 (Jordan) 消去法	(217)
2.1 简单的约当消去法	(218)
2.2 选主元的约当消去法	(226)
3 高斯 (Gauss) 消去法	(236)
3.1 高斯消去法	(236)
3.2 选主元高斯消去法	(245)
4 追赶法	(252)
4.1 三对角方程组	(252)
4.2 追赶法的基本思想	(253)
4.3 追赶法的计算公式	(253)
4.4 追赶法的算法设计	(254)

4.5 追赶法的程序框图设计	(255)
4.6 追赶法的 FORTRAN 源程序	(256)
小结	(257)
习题六	(257)
参考文献	(260)

第一章 绪 论

内容提要 对于新时代的高等学校的学生来说,掌握使用计算机解决其业务领域中遇到的问题,是必须具备的能力,否则将无法适应社会主义经济建设和科学技术日新月异飞速发展的形势。计算机数值计算方法是计算机科学首先发展,并至今仍是计算机科学重要的研究对象。由于本门课程涉及计算机程序设计和数值计算方法两方面的内容,在学习本课程前应具备高等数学和计算机高级程序语言方面的基本知识。本章将着重介绍计算机数值计算方法及程序这门课程的性质及研究对象,数值计算方法的基本概念与办法,数值计算中的误差,程序设计方法等。

1 课程性质及其研究对象

在电子计算机问世之前,为了解决某些复杂的计算问题,不少科学家为之献出了大半生甚至毕生的精力。例如,1867年法国天文学家达拉姆尼花了整整20年的时间,了解天体运动的一个摄动级数展开式,以解决月球运行轨道的数值解;又如18世纪英国数学家香克斯用了毕生的精力,于1873年把圆周率 π 计算到了小数点后707位。科学家把主要精力花费在复杂的计算上,这并非妥善之举。随着科学技术和社会的发展,大量复杂的计算问题也随之摆到了人们的面前,迫使人们去研究妥善解决计算问题的方式。

1.1 课程研究对象

有些复杂的计算问题,是无法用人工计算完成的,例如1948年美国原子能研究中有一计算问题,需要进行900万道运算,大约相当于1500名工程师一年的计算工作量;要准确预报天气变化情况,就要解算成千上万个偏微分方程组,这大概相当于数万名工程师一年的计算工作量。不难看出,上述计算问题,都是人工计算(包括使用电动计算机一类的计算工具)所难于或根本无法胜任的计算任务。为解决这样的计算任务,于是一门新的科学——计算机数值计算方法及程序随之迅速地发展起来。

1.1.1 数值计算方法

为了解决科学技术中的问题,首先要根据提出的问题,建立数学模型,然后进行解算。从数学模型求解的角度看,有解析法和数值解法、图解法等。例如,已知物体作匀变速直线运动时,其速度随时间变化的规律为 $v=v(t)=at$,其中, a 为常量,今欲求 $t=0$ 到 t 这段时间内物体所通过的路程 s 。显然,由物理学知:

$$\begin{aligned} s &= \int_0^t v(t) dt = \int_0^t at dt \\ &= \frac{1}{2} at^2 \Big|_0^t = \frac{1}{2} at^2 \end{aligned} \quad (1-1)$$

式(1-1)即为此数学模型的解析解。

若所给出的运动不是匀变速直线运动,而是非匀变速直线运动,如 $v(t)=\sin^2 x$,或 $\sin x/x$,则式(1-1)就无法使用解析法求出,而只能使用数值解法、图解法求出其近似解。

我们把求数值解的问题称为数值计算问题，把求数值解的方法称为数值计算方法，简称为计算方法或数值方法。

计算方法是数学的一个分支，是一门古老的学科，已有几百年的历史。它的研究对象是求解各种数学问题的数值方法及有关的理论，其主要内容有插值计算、数值积分计算、常微分方程数值解法、方程求根和线性方程组的数值解法等。

1.1.2 科学计算

计算方法的历史虽久，但只依赖于人工计算，因而，其发展速度十分缓慢。随着电子计算机的诞生，科学计算随之诞生。

非人工计算所能胜任而必须借助于电子计算机进行的科学技术的这样一类计算称为科学计算。例如，前文提到的原子能数学模型计算、天气预报计算都属于科学计算。

1.1.3 计算机数值计算方法

研究科学计算的方法、途径及其理论的科学称为计算机数值计算方法。电子计算机，使得过去无法计算的问题得以解决，极大地提高了数值计算的实用性，同时数值计算方法为适应电子计算机解题特点而获得了飞速的发展，两者相辅相成，相互促进，不断发展。

学习计算机数值计算方法，有两点值得注意：

①计算机只会做四则运算，更严格一点讲只会做加法，其他运算都是化作加法来进行的，例如求对数、三角函数运算、求幂、求平方根、求积分、解微分方程、解线性方程组等，最终都是化作加法计算的。由此，在使用计算机进行数值计算时研究算法就显得特别重要。

②以“快”取胜。计算机把各种运算都化成加法进行，于是增加了“转化”手续，要花费去更多的时间。但计算机以“快”取胜，“转化”与做加法都以极高速度进行，是其他任何计算工具无法比拟的，因而计算的高速度是计算机最显著的特点。

1.1.4 程序设计方法学

要使用计算机解决问题的一个先决条件，就是要让计算机理解并执行人给出的各种指令。程序是人们根据解题需要而设计、编制的解题方法和步骤，从计算机执行动作的角度来看，也可看成是一系列用计算机语言表示的指令的有序集合。为让计算机帮助解决问题，也就必须设计、编制程序，人们用计算机能识别的语言设计、编制、调试程序的过程，也就是通常人们所说的程序设计。为了用计算机语言编写出高质量的程序，使计算机能正确、快速、顺利地帮助人们解决问题，就必须对程序设计的过程及方法进行研究。研究程序结构、功能，程序设计的原则、方法及其理论的科学就是程序设计方法学。

1.1.5 计算机数值计算方法及程序

将计算机数值计算方法和程序方法学交叉起来，结合在一起研究，就是计算机数值计算方法及程序。确定科学计算的算法，并运用计算机语言将算法编制成程序，用以解决科学计算问题，就是计算机数值计算方法及程序的研究对象。

1.2 学习计算机数值方法及程序的重要性

在学习了计算机程序设计语言后，应不失时机地学习计算机数值计算方法及程序，其重要性表现在：

(1) 选择正确的算法是解决问题的必要前提

前面已提到，要借助于计算机解决问题，就必须编写正确的程序，而要能编写出正确的程序，除了正确掌握计算机语言外，还必须确定正确的算法。算法从数学的角度看，可以看作解题的方法和步骤；从使用计算机的角度看，算法也可以看作是按解题要求选择合适的操

作所组成的操作序列。没有正确的算法也就不可能借助于计算机解决问题。因此，选择正确的算法是使用计算机解决问题的必要前提，也是计算机数值计算方法及程序这一门课程研究的一个重要内容。

(2) 使计算机能进行它所不会或无法完成的那些运算

前面说到，计算机只会做加法，其他运算都是化作四则运算，最终化为加法来进行的。如经常遇到的运算： $\ln x$ ， $\sin x$ ， e^x ， $\sqrt{x}$ 等都化为四则运算，最终均化为加法。如何将其他运算化为四则运算，最终化为加法运算，这也是本门课程所要研究的一个内容。

值得一提的是，有些计算任务看起来只有加、减、乘、除，很容易，但计算工作量却大得惊人，计算机也只能“望洋兴叹”！在人们心目中，计算机进行加法运算的速度极快，似乎不管计算工作量如何巨大，计算机都能及时完成，但这是人们的一种错觉，事实并非如此，确实存在着计算机都无法完成的计算任务。如计算一个 25 阶行列式的值，按行列式理论，展开后共有 $25!$ 项，每项内包含 25 个数字元素相乘，即计算每一项需要做 24 次乘法，由此计算整个行列式的值需要进行的乘法次数为：

$$25! \times 24 \approx 3.7227 \times 10^{26}$$

要让计算机完成这么多次的乘法运算，需要多久的时间？现以 2.5×10^8 次/秒的计算机来进行此项运算任务，该计算机每一年能完成的计算次数为：

$$365 \times 24 \times 60 \times 60 \times 2.5 \times 10^8 = 7.884 \times 10^{15}$$

要完成 $25!$ 行列式计算所需的时间为：

$$T = 3.7227 \times 10^{26} \div (7.884 \times 10^{15})$$

$$\approx 4.7 \times 10^{10} \text{ (年)}$$

这是一个天文数字，这说明计算机实际上是无法完成这么巨大的计算任务的。那末计算机是不是连一个 25 阶行列式的值都无法计算呢？当然不是。事实上只要选择一种合适的算法，该行列式只要几秒钟就能完成计算。几秒钟和几百亿年，这是多么鲜明的对照，而这也告诉人们研究算法是多么重要。

(3) 提高程序编写质量

要正确使用计算机，必须首先编写出高质量的程序，要编写出高质量的程序，就必须满足两个基本条件：

① 选择正确的算法。程序和算法两者是相辅相成的，程序是算法的表达方式，算法是程序所要表达的实质内容。高质量的程序离不开高质量的算法。

② 使用正确的程序设计方法。当前，随着计算机科学的迅速发展，一门新的科学——程序设计方法学正在形成并逐步走向成熟。运用程序设计方法学所阐明的原则、原理与技术来设计编写程序，就能保证程序的正确性，增加程序的可读性和可维护性，这对于大型程序的设计有特别重要的意义。

如何满足以上两个条件，正是本门课程所要研究的内容。

(4) 促进思维发展

选择算法和编写程序来解决科学计算问题，这是一种与数学演绎推理解决计算问题不同的思维方式，这两种思维方式相互矛盾，又相互统一，相辅相成。学习掌握算法和编写程序的方法，对于完善思维方式，培养和提高思维能力，促进思维的发展具有重要的意义。

(5) 推动计算机科学的发展

计算机的出现，为计算机数值计算方法和程序设计方法学的发展提供了条件。在计算机