高等学校教材

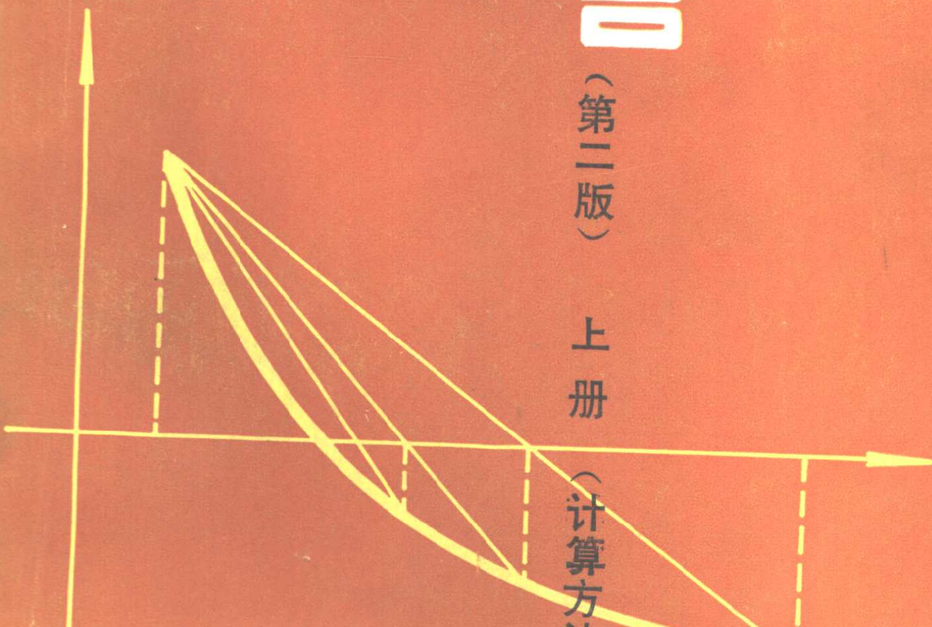
计算方法与算法语言

（第二版）

上册

（计算方法）

张德荣 王新民 高安民 编



高等教育出版社

高等学校教材

计算方法与算法语言

（第二版）

上册（计算方法）

张德荣　王新民　高安民　编

高等教育出版社

内容提要

本书是在1981年出版的高等学校试用教材《计算方法与算法语言》的基础上修改再版的. 全书分上、下两册出版. 上册为计算方法，下册为算法语言. 这次再版，删去了原书第一篇计算方法的少量内容，也增加了不少材料，并对不少内容进行了改写. 对第二篇算法语言作了彻底改写，改为讲述BASIC语言的最新版本True BASIC，并着重介绍结构化程序设计方法. 内容选取的范围和深度恰当，推理比较严密，对算法作了较充分的说明.

本书可作为师范院校的教材，也可作为其它院校有关课程的教材或参考书.

本书下册配有软盘一张(西文一张或汉化一张)，内容包括：True BASIC系统程序、各章例题的源程序、部分习题答案(源程序)，有需要者，可与高等教育出版社软件编辑室联系.

高等学校教材
计算方法与算法语言
(第二版)
上　册
计算方法
张德荣　王新民　高安民编
*
高等教育出版社出版
新华书店北京发行所发行
人民教育出版社印刷厂印装
*
开本 850×1168 1/32　印张 8　字数 190 000
1981年12月第1版 1989年8月第2版 1989年8月第1次印刷
印数 0001—4,710
ISBN7-04-002289-3/TP·53
定价 1.95元

再版说明

这次再版，删去了原书第一篇计算方法部分的少量内容；根据教与学的需要，改写了不少内容；还增加了一些选学材料.

对于原书第二篇算法语言部分，根据计算数学及其应用软件编审组会议的意见，作了彻底改写. 把讲述算法语言 ALGOL 60 改为讲述结构式语言 True BASIC. 介绍 True BASIC 语言的主要内容和编写结构程序的基本方法.

目前常用的 BASIC 语言的版本不少，它们之间，既有共同之处，也有差别，但使用它们编写程序的基本方法大致相同. 考虑到这种情况，书中还介绍了常用 BASIC 版本的一些情况，希望本书的适应范围较广.

书中所列程序的数量和种类都不算少，有些内容有一定难度，希望同志们酌情取舍. 算法和程序的关系极为密切，不少问题，用较多的篇幅先介绍算法. 希望读者不要在对算法不甚了解的情况下，企图熟悉解决问题的程序.

为了使用方便，本书分上(计算方法)、下(算法语言)两册出版. 北京师范大学陈公宁教授审阅了上册的全部内容，北京自动化工程学院谭浩强教授审阅了下册的全部内容，均提出了不少宝贵的建议，编者谨致以诚挚的感谢.

编者对计算数学及其应用软件编审组的全体同志，以及兄弟院校的同行们，也致以诚挚的谢意，感谢他们所给予的支持和帮助. 希望继续提出批评和建议.

编　者

1988 年 6 月于陕西师范大学

初版说明

本书是按照高等师范院校《计算方法与算法语言》教学大纲的要求编写的. 为了适应不同情况的需要，书中列出了少量选学内容. 这些内容，都在标题中加了星号“*”，使用时可酌情取舍.

阅读本书仅需《高等代数》和《数学分析》中的基础知识. 对一些不甚常见的内容，书中作了概括地介绍.

为了书写简便，用“定理2.6.2”表示第二章§6定理2；用“定理2.1”表示本章§2定理1；关于引理和公式，也有类似的记法. 此外，符号“$\triangleq$”表示“定义为”，即此符号左右两端是同一事物的两种不同表示方法.

本书试用过几次，使用的总学时是：计算方法72学时，算法语言30学时，上机实习12学时. 各章的学时分配，比大纲的相应部分略有增减.

西北大学数学系刘国良同志阅读了计算方法手稿，并试用过几次，还有些兄弟院校也试用了本教材或者阅读了交流讲义，他们都提出了修改意见，特此致谢.

吉林大学李荣华同志、北京师范大学刘贵贤同志和七机部二院二十六所袁兆鼎同志审阅了本书，都提出了不少宝贵的意见，谨致以衷心的感谢.

我系系主任魏庚人老师对本书的编写予以鼓励和支持，谨于此致谢.

本书第一篇计算方法由张德荣同志编写，第二篇算法语言由王新民、高安民二同志编写.

近年来，计算数学这一学科发展很快，编者对大纲内容的深度

和广度掌握不够，且限于水平，选材不当及错漏之处在所难免，恳请兄弟院校的老师和读者批评指正.

编　　者

一九八一年六月于陕西师范大学

上册目录

第一章　引　论

§1　数值问题和算法

在解决实际问题的长期过程中，形成了计算方法这门学科. 它的产生和应用比较早，自从快速通用电子数字计算机广泛使用以来，它又有很大的发展. 目前，它的理论和应用仍在继续向深度和广度方面发展.

计算方法涉及的问题相当广泛. 对于具体部门，则因需要不同而各有侧重. 本书选择一些常见的实际问题，通过对这些问题的讨论，说明计算方法的基本理论和基本特点.

有关的数学理论对于实际计算非常有用. 数学问题的理论分析往往为结果的计算奠定必要的基础，提供很有价值的线索. 例如，极限定理的证明能够给出逐步逼近的计算过程. 当然，计算方法也有许多特点. 解决实际问题时，应当根据问题的要求，计算工具的性能，选择良好的算法，以较高的效率，得出有用的结果. 为此，还应在理论分析的基础上，进行深入的数值分析.

解决实际问题时，往往先把它归结为数值问题. 所谓数值问题，指的是由一组已知数据（又称输入数据），求出一组结果数据（又称输出数据），使得这两组数据之间，满足预先指定的某种关系. 由于数字计算机的广泛使用，使越来越多的实际问题，能归结为数值问题而得到解决.

计算机能完成加、减、乘、除等算术运算，还能完成一些逻辑运算. 用完全确定的运算规则（包括运算的逻辑顺序），对某一类数值问题的输入数据按某种预定的规则进行有限次运算，判断此数

值问题是否有解，在解存在的情况下，给出输出数据，此种过程称为算法．当解不存在时，算法应能作出明确地判断，最好指出解不存在的关键．

计算方法主要讨论如何制定数值问题的算法，得出足够准确的数值解．与此同时，还将讨论解的特性，以及各种方法的优缺点及其适用范围．

本章先对计算机数系和计算的误差作一些初步介绍．

§2 浮 点 数

1 定点数

设r为大于1的整数，a_i为$0,1,\cdots,r-1$中的某一个．位数有限的r进制正数可以写成

$$x \triangleq a_{l-1}a_{l-2}\cdots a_0.a_{-1}a_{-2}\cdots a_{-m}. \tag{1}$$

x有l位整数，有m位小数．因为进位制的基数是r，所以

$$x = a_{l-1}\times r^{l-1}+a_{l-2}\times r^{l-2}+\cdots+a_0\times r^0$$
$$+a_{-1}\times r^{-1}+a_{-2}\times r^{-2}+\cdots+a_{-m}\times r^{-m}. \tag{1'}$$

当$l=4, m=4, r=10$时，

109.312,　　0.4375,　　4236,

分别表示为

0109.3120,　　0000.4375,　　4236.0000.

这种把小数点永远固定在指定位置上，而位数有限的数，称为定点数．当$l=m=4$而$r=10$时，八位定点非零数中，绝对值最小和最大的数分别为

$$\pm 0000.0001, \qquad \pm 9999.9999.$$

由此可见，定点数所能表示的数的范围非常小．

2 浮点数

设s是十进制数，p是十进制正负整数或零．数x可以用

s 和 10^p 的乘积来表示，即

$$x \triangleq s \times 10^p. \tag{2}$$

再设 s 的整数部分等于零，即 s 满足条件

$$-1 < s < 1, \tag{3}$$

则 x 的小数点位置主要由整数 p 决定．即使 s 的位数和 p 的位数之和不大，也能表示绝对值相当大的数以及绝对值相当小的数．形如(2)而满足条件(3)的十进制数x，称为十进制浮点数．s 和 p 分别称为浮点数 x 的尾数和阶数．如果尾数的小数位数等于有限正整数 t，则把 x 称为 t 位浮点数．

此外，如果还要求尾数 s 小数点后第一位数字不等于零，也就是要求尾数 s 满足条件

$$10^{-1} \leqslant |s| < 1, \tag{4}$$

则形如(2)而满足条件(4)的浮点数称为十进制规格化浮点数．例如数

$$0.004012, \qquad 0.3217, \qquad 284.5$$

的规格化浮点数分别为

$$0.4012 \times 10^{-2}, \quad 0.3217 \times 10^{0}, \quad 0.2845 \times 10^{3}.$$

数零不能用规格化浮点数来表示．

上面定义了十进制浮点数和十进制规格化浮点数．设用任何大于 1 的整数作为数制的基数，并设 s 是 r 进制数，p 是 r 进制正负整数或零，则形如

$$x \triangleq s \times r^p \tag{2'}$$

并满足条件(3)的数称为 r 进制浮点数．如果再让 s 的小数点后第一位数字不等于零，即 s 满足条件

$$r^{-1} \leqslant |s| < 1, \tag{4'}$$

则形如(2′)并满足条件(4′)的 r 进制数 x 称为 r 进制规格化浮点数。

3 计算机中的数系

任一计算机只能用有限的位数来表示浮点数的尾数和阶数.设进位制为 r,阶数 p 满足条件

$$-m\leqslant p\leqslant M, \tag{5}$$

其中 m,M 为正整数，它们主要由计算机用多少位数来表示阶数而决定. 如果尾数的小数位数为 t(t 一般比 m 和 M 的位数大若干倍)，则计算机的数系由一切阶数满足(5)的 t 位 r 进制浮点数的集合 F 组成.

当 $r=10, t=4, m=M=99$ 时,

$$-0.0001\times 10^{-99}, \quad 0.0001\times 10^{-99}$$

是数系 F 中绝对值最小的非零数,而

$$-0.9999\times 10^{99}, \quad 0.9999\times 10^{99}$$

是此数系中的最小数和最大数. 若计算的中间结果超出了上述范围,则称为溢出.

当 $10^{-1}\leqslant s<s+10^{-4}<1, -m\leqslant p\leqslant M$ 时,

$$s\times 10^{p} \quad 和 \quad (s+10^{-4})\times 10^{p}$$

之间的任何数都不属于上述计算机的数系. 一切满足上列条件的二数之差都等于 10^{-4+p}. 由此可见,在计算机数系 F 中,数的个数有限,数系中的每一个数都是有理数. 从整体看,数系中的数分布很不均匀;从局部看,阶数相等的数,又以相等的距离,分布在数轴的某一段上.

在计算机中，常用尾数等于零的数来表示零. 零不能化为规格化浮点数.

使用一定位数的浮点数进行数值计算，可以满足一般实际问题的要求. 所以,虽然计算机数系由一些残缺不全、分布不匀的数组成,但用以解决实际问题却很有效.

§ 3　误差的基本概念

1　误差的来源

用近似方法解决科学技术问题时，一般有误差，其来源有下列四种.

1）数学描述和实际问题之间的误差. 用数学模型描述实际问题时，往往抓住主要因素，略去一些次要因素，将实际问题理想化以后，才进行数学概括. 这种描述，虽然相当好地反映了实际情况，但也有误差.

2）观测误差. 数值问题的原始数据，一般由观测或实验获得. 观测结果和这些数量的实际大小总有误差. 这种误差，称为**观测误差**.

3）截断误差. 实际计算只能用有限次运算来完成，理论上的精确值往往要求用无限的过程才能求出. 例如，已知 $x>0$ 求 e^{-x} 时，由表达式

$$e^{-x}=1-x+\frac{1}{2}x^2-\frac{1}{6}x^3+\cdots \tag{1}$$

取部分和

$$E(x)\triangleq 1-x+\frac{1}{2}x^2-\frac{1}{6}x^3 \tag{2}$$

作为 e^{-x} 的近似值，有

$$e^{-x}-E(x)=\frac{1}{24}e^{-\xi}x^4, \tag{3}$$

其中 ξ 在零和 x 之间. 这样求出的近似值 $E(x)$ 和精确值 e^{-x} 之间，存在着误差. 这种类型的误差，称为**截断误差**.

4）舍入误差. 计算机数系是有限集，不仅无理数 e,π 等不属于计算机数系，一些有理数，如 1/3，也不属于计算机数系. 常常用计算机数系中和它们比较接近的数来表示它们. 由此产生的误

差，称为舍入误差.

观测误差和原始数据的舍入误差，就其来源说，有所不同，就其对计算结果的影响看，完全一样. 数学描述和实际问题之间的误差，往往是计算工作者不能独立解决的，甚至是尚待研究的课题. 基于这些原因，在计算方法课程中所涉及到的误差，一般指舍入误差(包括初始数据的误差)和截断误差. 讨论它们在计算过程中的传播和对计算结果的影响；研究控制它们的影响以保证最终结果有足够的精度；既希望解决数值问题的算法简便而有效，又想使最终结果准确而可靠.

2 绝对误差和相对误差

假设某一数 x 的近似值是 x^*(经常在一数的右上角加星号"*"表示此数的近似值). 由

$$\varepsilon(x) \triangleq x^* - x \tag{4}$$

定义的数 $\varepsilon(x)$ 称为近似数 x^* 的绝对误差，简称误差.

准确值 x 一般是未知的，因而 $\varepsilon(x)$ 也是未知的. 但往往可以估计出绝对误差的上限，即可以求出一正数 η，使

$$|\varepsilon(x)| \leqslant \eta. \tag{5}$$

满足上式的 η 称为 x^* 的绝对误差限.

例如，用(2)表示的 $E(x)$ 作为 e^{-x} 的近似值，当 $0 \leqslant x \leqslant 1$ 时，由(3)式可得

$$|E(x) - e^{-x}| \leqslant 1/24.$$

有时也用

$$x = x^* \pm \eta \tag{6}$$

来表示(5)式. 例如 $x = 0.3106 \pm 0.0014$ 指的是

$$0.3092 \leqslant x \leqslant 0.3120.$$

绝对误差不足以刻画近似数的精确程度. 例如，测量飞机机翼长度时，发生一毫米的误差，和测量飞机机翼厚度时，发生一毫

米的误差，大不一样．要决定一个近似值的精确程度，除了绝对误差以外，还必须考虑此数本身的大小．这就需要引进相对误差的概念．

绝对误差和准确值的比，即

$$\varepsilon_r(x) \triangleq \frac{\varepsilon(x)}{x} = \frac{x^* - x}{x}, \tag{7}$$

称为 x^* 的相对误差．上式描述了绝对误差和相对误差的关系．

准确值一般是未知的，因而 $\varepsilon_r(x)$ 一般也是未知的．但往往可以估计出相对误差的上限，即可以求出一正数 δ，使得

$$|\varepsilon_r(x)| \leqslant \delta. \tag{8}$$

满足上式的 δ 称为 x^* 的相对误差限．

绝对误差和用 x 表示物理量时使用的单位有关，而相对误差则和单位无关．

例如，以

$$x^* \triangleq 0.8727 \times 10^{-90}, \quad y^* \triangleq 0.8727 \times 10^{90}$$

分别作为

$$x \triangleq 0.8726 \times 10^{-90}, \quad y \triangleq 0.8726 \times 10^{90}$$

的近似值，其绝对误差和相对误差分别为

$$\varepsilon(x) = 0.0001 \times 10^{-90}, \quad \varepsilon(y) = 0.0001 \times 10^{90};$$

$$\varepsilon_r(x) = \frac{0.0001}{0.8726}, \qquad \varepsilon_r(y) = \frac{0.0001}{0.8726}.$$

3 近似数的算术运算

现在讨论进行加、减、乘、除等运算时，原始数据的相对误差和计算结果的相对误差之间的关系．

1）乘法和除法．设正数 x 的近似值为 $x+\Delta x$，绝对误差 Δx 近似地等于 x 的微分，即 $\Delta x \approx dx$．相对误差是

$$\varepsilon_r(x) = \frac{\Delta x}{x} \approx \frac{dx}{x} = d\ln x, \tag{9}$$

即 x 的相对误差近似地等于 $\ln x$ 的微分.

下面用这种方法讨论积与商的相对误差. 由

$$\varepsilon_r(x_1x_2) \approx d\ln(x_1x_2) = d\ln x_1 + d\ln x_2 \tag{10}$$

可知, 乘积的相对误差是各乘数相对误差的和. 同理可得商的相对误差是被除数与除数相对误差的差,即

$$\varepsilon_r\left(\frac{x_1}{x_2}\right) \approx d\ln\frac{x_1}{x_2} = d\ln x_1 - d\ln x_2\,. \tag{11}$$

一般地, 任意多次连乘连除所得结果的相对误差限可取为各乘数和各除数相对误差限的和.

2) 加法和减法. 加法和减法的运算结果是数的代数和. 设 x_1, x_2 的近似值分别为 $x_1+\Delta x_1, x_2+\Delta x_2$,则

$$\begin{aligned}\varepsilon_r(x_1+x_2) &= \frac{\Delta(x_1+x_2)}{x_1+x_2} = \frac{\Delta x_1+\Delta x_2}{x_1+x_2}\\ &= \frac{x_1}{x_1+x_2}\cdot\frac{\Delta x_1}{x_1} + \frac{x_2}{x_1+x_2}\cdot\frac{\Delta x_2}{x_2}\\ &= \frac{x_1}{x_1+x_2}\cdot\varepsilon_r(x_1) + \frac{x_2}{x_1+x_2}\cdot\varepsilon_r(x_2).\end{aligned} \tag{12}$$

若 x_1 和 x_2 同号,则上式右端 $\varepsilon_r(x_1)$ 和 $\varepsilon_r(x_2)$ 的系数

$$\frac{x_1}{x_1+x_2}, \qquad \frac{x_2}{x_1+x_2} \tag{13}$$

都在零和 1 之间,并且它们的和等于 1. 这时,由(12) 可得

$$\begin{aligned}|\varepsilon_r(x_1+x_2)| &\leqslant \frac{x_1}{x_1+x_2}\max(|\varepsilon_r(x_1)|, |\varepsilon_r(x_2)|)\\ &\quad + \frac{x_2}{x_1+x_2}\max(|\varepsilon_r(x_1)|, |\varepsilon_r(x_2)|)\\ &\leqslant \max(|\varepsilon_r(x_1)|, |\varepsilon_r(x_2)|).\end{aligned} \tag{14}$$

所以,当一些数的符号相同时,它们和的相对误差限小于诸数相对误差限中的最大者.

若 x_1 和 x_2 满足条件 $|x_1| \gg |x_2|$ (表示 $|x_1|$ 远大于 $|x_2|$), 则

(13)中前一数近似等于1，后一数绝对值相当小，这时，由(12)可得

$$\varepsilon_r(x_1+x_2)\approx\varepsilon_r(x_1). \tag{15}$$

所以，当二数的绝对值相差很大时，这二数代数和的相对误差近似地等于绝对值较大者的相对误差.

若 x_1 和 x_2 异号，则(13) 中两个数的绝对值至少有一个大于1. 如果这时 x_1 和 $-x_2$ 相当接近，则(13)中两个数的绝对值都可能很大. 由(12)式可以看出，在这种情况下，原始数据的误差会对计算结果产生相当大的影响.

4 数的近似表示

大家都知道用“四舍五入”或“只舍不入”的办法求实数的近似值. 例如

$$\pi=3.14159265\cdots,$$

在四位浮点数系中，通常取

$$\pi_R=0.3142\times10^1(\text{四舍五入时}),$$
$$\pi_C=0.3141\times10^1(\text{只舍不入时})$$

作为 π 的近似值.

下面讨论数的近似表示. 绝对值相等的正负数仅差一个符号，所以我们只讨论正数的近似表示，并且假设此数属于计算机数系. 设正数 x 的十进制规格化浮点表示为

$$x\triangleq s\times10^p,\quad 10^{-1}\leqslant s<1, \tag{16}$$

s 的小数位数可能无限. 若 s 的十进制小数表示为

$$s\triangleq0.d_1d_2\cdots d_td_{t+1}\cdots, \tag{17}$$

其中$0<d_1\leqslant9,0\leqslant d_i\leqslant9$. 定义 t 位十进制数 s_R 如下:

$$s_R=\begin{cases}0.d_1d_2\cdots d_t, & \text{当}\ 0\leqslant d_{t+1}<\dfrac{1}{2}\times10\ \text{时;}\\ 0.d_1d_2\cdots d_t+10^{-t}, & \text{当}\ \dfrac{1}{2}\times10\leqslant d_{t+1}\ \text{时.}\end{cases}$$