研究生教材

YANJIUSHENGJIAOCAI

数值分析

SHU ZHI FEN XI

主编 陈晓江

武汉理工大学出版社
WUTP Wuhan University of Technology Press

研究生教材

数值分析

主　编　陈晓江
副主编　王伟沧　陈建业

武汉理工大学出版社

内 容 提 要

本书是作者在20多年讲授研究生数值分析课程的基础上编写而成的。全书共分11章,内容包括:绪论、插值法、拟合与逼近、数值积分与数值微分、线性方程组的直接解法、线性方程组的迭代解法、非线性方程求根的数值解法、常微分方程的数值解法、矩阵特征值问题的数值解法、智能计算初步、数值计算问题的MATLAB实现。本书从实用角度出发,介绍科学与工程计算中常用的数值计算方法和理论,介绍各种方法的MATLAB实现,配有常用的、可运行的程序,配有大量的例题、习题,每章有小结,书后有习题答案。

本书可作为理工科大学非数学专业的研究生或数学专业高年级本科生的教材,也可作为科技工作者的参考书。

图书在版编目(CIP)数据

数值分析/陈晓江主编. —武汉:武汉理工大学出版社,2013.10
ISBN 978-7-5629-4166-8

Ⅰ.①数…
Ⅱ.①陈…
Ⅲ.①数值分析
Ⅳ.①O241
中国版本图书馆CIP数据核字(2013)第222398号

项目负责人:陈军东　彭佳佳　　**责任编辑**:彭佳佳
责任校对:段　智　　**装帧设计**:芳华时代
出版发行:武汉理工大学出版社
社　　址:武汉市洪山区珞狮路122号
邮　　编:430070
网　　址:http://www.techbook.com.cn
经　　销:各地新华书店
印　　刷:荆州市鸿盛印务有限公司
开　　本:787×960　1/16
印　　张:18.25
字　　数:380千字
版　　次:2013年10月第1版
印　　次:2013年10月第1次印刷
印　　数:1—2000册
定　　价:27.50元

凡购本书,如有缺页、倒页、脱页等印装质量问题,请向出版社发行部调换。
本社购书热线电话:027-87523148　87664138　87515798　87165708(传真)

前　言

在科学与工程计算中，怎样选择与使用适当的数值计算方法，怎样估计计算结果的误差，怎样解释计算过程中的异常现象，已成为广大科技工作者迫切需要解决的问题。由于这一原因，现在各高等院校为非数学专业的研究生和数学专业的高年级学生普遍开设“数值分析”课程。

本书从实用的角度出发，通过实际问题引出基本概念，着重讲清原理，突出算法的构造和分析，并通过大量的例题帮助读者解决做题难的问题，每章最后都有小结，并附有适量的习题，书后给出了习题的参考答案。本书前 9 章是基本的授课内容，第 10 章是拓展内容，介绍智能计算的三种主要方法。第 11 章是数值计算问题的 MATLAB 实现，介绍用 MATLAB 解决各类问题的应用实例，配有常用的、可运行的程序，帮助读者自己动手，用 MATLAB 解决具体问题。

读者可根据不同的需要，选择适当的章节进行学习。根据我们的教学实践，本书基本内容可在 64 学时内完成。根据不同专业的需要，略去部分内容，可适用于 40 ~ 64 学时的教学需要。

本书由陈晓江主编，并编写第 1、2、3、4、8、11 章；王伟沧编写第 5、6、9 章；陈建业编写第 7、10 章；陈晓江负责全书的统稿。在本书的编写过程中，王卫华教授认真审阅了书稿，提出了修改意见，在此表示衷心感谢。本书的编写得到了武汉理工大学出版社的大力支持，在此一并表示感谢。

由于作者水平有限，本书的缺点错误在所难免，敬请读者批评指正。

编　者

2013.8.5

目　录

第1章　绪　　论

随着科学技术的发展，科学与工程计算已被推向科学活动的前沿. 科学与工程计算的范围扩大到了所有科学领域，并与科学实验、科学理论三足鼎立，相辅相成，成为人类科学活动的三大方法之一. 因此，熟练地运用计算机进行科学计算，已成为科技工作者的一项基本技能. 这就要求人们去研究和掌握适用于计算机上使用的数值计算方法，而数值分析就是研究用计算机解决数学问题的方法及其有关理论.

1.1　问题的提出

在高等数学中，计算定积分是一个很平常的事.

由函数 $f(x)=\mathrm{e}^{-x^2}$ 在闭区间[0,1]上连续，可知定积分 $\int_0^1 \mathrm{e}^{-x^2}\,\mathrm{d}x$ 是存在的，但是它不能用牛顿-莱布尼茨(Newton-Leibniz)公式求解，因为不定积分 $\int \mathrm{e}^{-x^2}\,\mathrm{d}x$ 积不出来，被积函数 $f(x)=\mathrm{e}^{-x^2}$ 的原函数不能用初等函数表示，所以定积分 $\int_0^1 \mathrm{e}^{-x^2}\,\mathrm{d}x$ 存在但是求不出精确解，而只能求其满足一定精度的近似解. 像这样的计算问题，数学中还有很多. 如何解决这些计算问题，如何实现数学的实用价值，这就是数值分析这门课的任务(上述计算定积分近似解的问题称为数值积分问题，将在第4章中解决).

对于理工科大学生、研究生和广大科学技术人员来说，学习数学的目的是在实际工作中能综合应用数学方法去认识问题、解决问题. 一般地，用数学方法解决实际问题，包括建立数学模型和求模型的数值解这两个基本过程.

具体来说，数学模型就是为了某种目的，用字母、数字及其他数学符号建立起来的等式或不等式以及图表、图像、框图等描述客观事物的特征及其内在联系的数学结构表达式. 例如，图1.1所示的直角三角形，两条直角边分别为 a 和 b，斜边为 c，由勾股定理知 a、b、c 满足关系式

$$a^2+b^2=c^2 \tag{1.1}$$

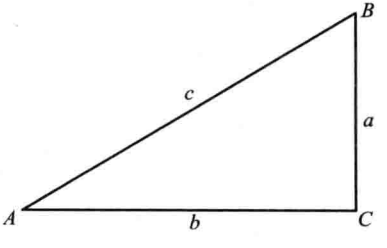

图1.1　直角三角形

式(1.1)就是描述直角三角形边长关系的数学模型.利用这个模型,只要知道其中任意两条边的长,就可以求出第三条边的长.

还是利用图1.1所示的直角三角形,由三角函数的定义,可得

$$\tan\angle A = \frac{a}{b} \tag{1.2}$$

式(1.2)与式(1.1)相比,数学模型稍微复杂一点,可以用来解决更多的问题,比如可以用来求角的度数.

无论是式(1.1)还是式(1.2),如果不用来进行数值计算,仅仅停留在理论上,这些模型对于解决实际问题就起不到应有的作用.然而就是这些简单的模型,求它们的数值解也还是需要一定方法的.下面利用这两个模型来看看相应的数值解问题.

在图1.1所示的直角三角形中,取两条直角边 $a = b = 1$,则由式(1.1)得斜边 $c = \sqrt{2}$.保留四位小数,计算$\sqrt{2}$的近似值.

仅用基本的加、减、乘、除运算,计算$\sqrt{2}$的近似值就有许多不同的算法,我们可用一种特殊的递推公式

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{2}{x_n}\right) \tag{1.3}$$

来计算,取 $x_1 = 1$,由式(1.3),可得 $x_2 = 1.5$、$x_3 = 1.4167$、$x_4 = x_5 = \cdots = 1.4142$,所以$\sqrt{2}$的近似值为1.4142.

这种算法是一种迭代算法,将在第7章中介绍.

同样在图1.1所示的直角三角形中,取两条直角边 $a = b = 1$,则由式(1.2)得 $\tan\angle A = 1$,$\angle A = \frac{\pi}{4}$,可用来计算 π 的近似值.

由 $\tan\frac{\pi}{4} = 1$,利用级数展开式,可得

$$\pi = 4\arctan 1 = 4\sum_{n=0}^{\infty}\frac{(-1)^n}{2n+1} \tag{1.4}$$

但是式(1.4)中级数的部分和数列收敛比较慢.要快速算出 π 的近似值,可以用下面公式来计算

$$\begin{aligned}\pi &= 16\arctan\frac{1}{5} - 4\arctan\frac{1}{239}\\ &= 16\sum_{n=0}^{\infty}\frac{(-1)^n}{2n+1}\left(\frac{1}{5}\right)^{2n+1} - 4\sum_{n=0}^{\infty}\frac{(-1)^n}{2n+1}\left(\frac{1}{239}\right)^{2n+1}\end{aligned} \tag{1.5}$$

计算 π 的近似值,也可以用积分式子

$$\pi = 4\int_0^1 \frac{1}{1+x^2}\mathrm{d}x \tag{1.6}$$

利用数值积分的方法来解决.

对于复杂的求近似解问题,我们将在后面各章中详细介绍.

1.2 数值分析的内容与特点

数值分析是计算数学的一个主要部分. 计算数学是数学科学的一个分支,它研究用计算机求解各种数学问题的数值计算方法及其理论与软件实现. 一般地说,用计算机解决科学计算问题,首先需要针对实际问题提炼出相应的数学模型,然后为解决数学模型设计出数值计算方法,经过程序设计之后上机计算,求出数值结果,再由实验来检验. 概括如图 1.2 所示.

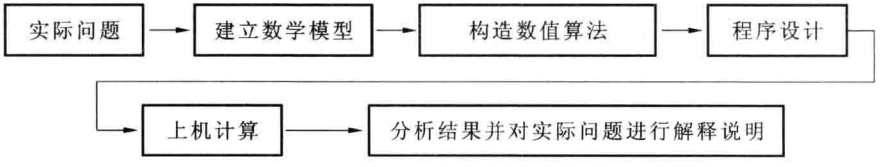

图 1.2 实际问题的求解过程

其中根据数学模型提出求解的数值算法直到编出程序,上机计算求出结果,这一过程是计算数学的任务,也是数值分析研究的对象. 因此,数值分析是寻求数学问题近似解的方法、过程及其理论的一个数学分支. 它以纯数学为基础,但却不完全像纯数学那样只研究数学本身的理论,而是着重研究数学问题求解的数值计算方法以及与此有关的理论,包括方法的收敛性、稳定性及误差分析;还要根据计算机的特点研究计算时间最省(或计算量最少)的计算方法. 有的方法在理论上虽然还不够完善与严密,但通过对比分析、实际计算和实践检验等手段,被证明是行之有效的方法,也可采用. 因此数值分析既有纯数学的高度抽象性与严密科学性的特点,又有应用数学的广泛性与实际试验的高度技术性的特点,是一门与计算机紧密结合的实用性很强的数学课程.

目前,计算机已成为数值计算的主要工具,数值分析的主要任务是研究适合计算机使用、满足精度要求、节省计算时间的有效算法及其相关的理论. 在实现这些算法时往往还要根据计算机的容量、字长、速度等指标,研究具体的求解步骤和程序设计技巧. 数值分析的特点概括起来有四点:

第一,面向计算机,要根据计算机特点提供切实可行的有效算法. 即算法只能包括加、减、乘、除运算和逻辑运算,这些运算是计算机能直接处理的运算.

第二,有可靠的理论分析,能任意逼近并达到精度要求,对近似算法要保证收敛性和数值稳定性,还要对误差进行分析. 这些都建立在相应数学理论的基础上.

第三,要有好的计算复杂性,包括好的时间复杂性(计算时间少)和好的空间复杂性(占用存储单元少). 对很多数值问题使用不同算法,其计算复杂性将会大不一样,这也是数值算法要研究的问题,它关系到算法能否在计算机上实现.

第四，要有数值试验，即任何一个算法除了从理论上要满足上述三点外，还要通过数值试验证明是行之有效的.

例如，求解线性方程组 $\boldsymbol{Ax}=\boldsymbol{b}$，若 $\det(\boldsymbol{A})\neq 0$，则可用克莱姆(Cramer)法则来求解. 设 $\boldsymbol{A}$ 为20阶矩阵，计算一个20阶行列式需要的乘法运算量为 $19\times 20!$，需要计算21个20阶的行列式，总的乘法运算量为

$$21\times 19\times 20!\approx 9.71\times 10^{20}$$

若用最新的天河二号超级计算机(峰值 5.49×10^{16} 次/秒、持续 3.39×10^{16} 次/秒)来运算，则1 h可完成的乘法运算量为

$$3.39\times 10^{16}\times 3600\approx 1.22\times 10^{20}$$

求解20阶的线性方程组所需乘法运算的时间为

$$9.71\times 10^{20}\div(1.22\times 10^{20})\approx 7.96(\text{h})$$

即7.96 h，显然这个运算时间对超级计算机来说太长了. 而在实际问题中，例如大型水利工程、天气预报等，需要求解的大型线性方程组的阶数一般都远远大于20，若用上述方法显然无法解决. 这个例子说明求解线性方程组的克莱姆法则在理论上虽然可行，但在实际应用中却是不可行的. 有人可能说，随着计算机的发展，运算速度提高、内存增大以及新结构计算机的出现，以前认为过于复杂而不能求解的问题将会得到解决. 但是，不论计算机如何发展，使用计算机的代价，即计算复杂性，都是需要考虑的.

1.3　计算机机器数系与浮点运算

微积分学的基础是实数系，而数值计算方法的理论则是建立在计算机机器数系的基础上. 为了设计高效、可靠的算法，这里简要介绍计算机机器数系的基本知识.

1.3.1　二进制数与计算机机器数系

在大多数计算机中，实数是以二进制形式表示的，并且在二进制实数系统中进行运算. 这似乎与我们从计算机屏幕上看到的不一样. 事实上，计算机首先将我们输入的十进制数转换为二进制数，然后在二进制实数系统中做运算，最后，再将结果转换为十进制数.

【例1.1】　将 $x=237$ 表示为二进制数.

解　将 x 展开成2的乘幂之和

$$x=237=1\times 2^7+1\times 2^6+1\times 2^5+0\times 2^4+1\times 2^3+1\times 2^2+0\times 2^1+1\times 2^0$$

即 x 的二进制表示为：$x=(11101101)_2$.

【例 1.2】 将 $x=0.65625$ 与 $y=0.7$ 分别表示为二进制数.

解 将 x 展开成 2 的负乘幂之和

$$x=1\times 2^{-1}+0\times 2^{-2}+1\times 2^{-3}+0\times 2^{-4}+1\times 2^{-5}$$

即 x 的二进制表示为:$x=(0.10101)_2$. 用类似的方法可求得 $y=(0.1\,\overline{0110})_2$,这里,$\overline{0110}$ 表示 0110 的循环.

对于一般实数 x,将 x 展开成

$$x=\pm(b_{J-1}\times 2^{J-1}+\cdots+b_1\times 2^1+b_0\times 2^0+b_{-1}\times 2^{-1}+$$
$$b_{-2}\times 2^{-2}+\cdots+b_{-n}\times 2^{-n}+\cdots)$$

这样 x 的二进制表示为:$x=\pm(b_{J-1}\cdots b_1b_0.b_{-1}b_{-2}\cdots b_{-n}\cdots)_2$,$b_j(j=J-1,\cdots,1,0,-1,-2,\cdots,-n,\cdots)$ 是 1 或 0.

例如,$18.25=1\times 2^4+0\times 2^3+0\times 2^2+1\times 2^1+0\times 2^0+0\times 2^{-1}+1\times 2^{-2}=(10010.01)_2$.

上述 x 的二进制表示可以写成与十进制类似的浮点形式

$$x=\pm 0.b_{J-1}\cdots b_1b_0b_{-1}b_{-2}\cdots b_{-n}\cdots\times 2^J$$

小数部分 $\pm 0.b_{J-1}\cdots b_1b_0b_{-1}b_{-2}\cdots b_{-n}\cdots$ 称为尾数,2 的指数 J 称为阶码,是整数. 一般地,一个数可以有不同的浮点表示,例如

$$18.25=0.1001001\times 2^5=0.01001001\times 2^6$$

为了保证唯一性,通常规定非零数的尾数的第一位数字非零,即 $b_{J-1}=1$,在这种规定下的浮点表示,称为规格化的二进制浮点数.

在计算机中,一个非零数通常被表示为如下二进制浮点形式

$$\pm 0.b_1b_2\cdots b_t\times 2^m$$

其中 $b_j(j=2,3,\cdots,t)$ 是 1 或 0,$b_1=1$;t 称为计算机的字长;阶码 m 有固定的上、下限,即 $L\leqslant m\leqslant U$,L、U 和 t 随计算机而异. 上述形式的数称为机器数. 由于机器数的字长与阶码有限,因此计算机中的数是有限的. 事实上,计算机中共有

$$2^t(U-L+1)+1$$

个机器数. 把计算机中的全体机器数组成的集合记为 F 或 $F(2,t,L,U)$,称为计算机机器数系. 机器数系 F 不是连续统,它是一个有限的、离散的、分布不均匀的集合. 不难验证,F 中任意非零数 y 满足

$$2^{L-1}\leqslant|y|\leqslant 2^U(1-2^{-t})$$

机器数有单精度与双精度之分,字长 t 的值规定了机器数的精度. 一般地,单精度数 $t=23$,约为十进制的 7 位有效数字. 双精度数 $t=52$,约为十进制的 15 位有效数字. 字长越大,机器数的精度越高. 阶码 m 的值规定了机器数的绝对值范围,单精度数阶码 m 的范围为 $-127\leqslant m\leqslant 128$,其绝对值范围为 $2^{-128}\sim 2^{128}$,即 $10^{-38}\sim 10^{38}$. 双精度数阶码 m 的范围为 $-1023\leqslant m\leqslant 1024$,其绝对值范围为 $2^{-1024}\sim 2^{1024}$,即 $10^{-308}\sim 10^{308}$.

在计算中,当数据的绝对值不在上述范围之内时,称为产生溢出:小于机器数

下限时称为产生下溢出，此时，对应机器数被取为零；大于机器数上限时称为产生上溢出，此时，对应机器数被取作无穷大，程序停止执行.

1.3.2　数据的表示与浮点运算

无论怎样的计算机，其机器数系 $F(2,t,L,U)$ 都是一个有限的集合，它所表示的实数只是实数系的一小部分. 绝大多数实数输入计算机时，要转换为有限字长的二进制机器数，总要经“舍”或“入”，而由一个与之相近的机器数代替. 实数 x 对应的机器数记为 $fl(x)$.

一般地，设 $x=\pm 0.b_1b_2\cdots b_t\cdots\times 2^m$，且 $2^{L-1}\leqslant|x|\leqslant 2^U(1-2^{-t})$，则

$$fl(x)=\operatorname{sgn}(x)\,\bar{a}\times 2^m \tag{1.7}$$

其中

$$\bar{a}=\begin{cases}0.b_1b_2\cdots b_t, & \text{若 } b_{t+1}=0\\ 0.b_1b_2\cdots b_t+2^{-t}, & \text{若 } b_{t+1}=1\end{cases} \tag{1.8}$$

这种获取机器数的方法称为舍入法；另一种获取机器数的方法称为截断法. 此时，对应上述 x 的机器数为

$$fl(x)=\operatorname{sgn}(x)0.b_1b_2\cdots b_t\times 2^m$$

【例 1.3】　将实数 $x=2.65625$ 与 $y=0.1$ 分别表示为 $F(2,8,-19,19)$ 中的机器数.

解　因为 $x=2.65625=0.1010101\times 2^2\in F$，所以，

$$fl(x)=x=0.1010101\times 2^2$$

而 $y=0.1=(0.\overline{0011})_2=0.\overline{1100}\times 2^{-3}\notin F$，但 $2^{-20}\leqslant|y|<2^{20}$，按舍入法

$$fl(y)=0.11001101\times 2^{-3}=0.100097656$$

按截断法

$$fl(y)=0.11001100\times 2^{-3}=0.099609375$$

以上介绍了二进制机器数系，机器数系不仅可以是二进制，还可以是 β 进制，例如八进制、十六进制、十进制等. β 进制机器数系可表为 $F(\beta,t,L,U)$，F 中任意非零数 y 可表示为

$$y=\pm 0.b_1b_2\cdots b_t\times\beta^m$$

其中 $0\leqslant b_j\leqslant\beta-1,(j=1,2,\cdots,t),b_1\neq 0$；$t$ 称为机器数的字长；阶码 m 满足 $L\leqslant m\leqslant U$. 特别地，十进制机器数系为 $F(10,t,L,U)$. 在下面的讨论中，为适应人们的习惯，采用十进制机器数系. 类似于二进制机器数系，十进制机器数系 $F(10,t,L,U)$ 也按两种方式获取机器数：舍入式或截断式. 前者是按四舍五入原则截取 x 尾数的前 t 位数，后者是直接截取 x 尾数的前 t 位作为机器数的尾数.

【例 1.4】　将实数 π 表示为 $F(10,5,-19,19)$ 中的机器数.

解 $fl(\pi)=0.31416\times 10$ （舍入式）

$fl(\pi)=0.31415\times 10$ （截断式）

下面讨论计算机中浮点数的运算. 如前所述，计算机只能做加、减、乘、除四则运算，而且机器数系对四则运算并不封闭，也就是说 F 中任意两数的和、差、积、商不一定都在 F 中. 此时，计算机自动将计算结果用 F 中的机器数表示出来.

设 x 和 y 都是机器数，即 $x,y\in F(10,t,L,U)$，它们的算术运算符合下述规则：

(1) 加减法：先对阶（靠高阶），后运算，再舍入；

(2) 乘除法：先运算，再舍入.

在运算中，不妨假定计算机具有双精度累加寄存器，即在运算时先保留 $2t$ 位，最后再把第 $t+1$ 位的数进行四舍五入. 下面举例说明.

【例 1.5】 设 $x=0.50556128\times 10^{-3}$，$y=0.23162743\times 10^{2}$，$z=-0.23162132\times 10^{2}$，在 $F(10,8,-29,29)$ 中，按舍入式，分别计算和 $x+y+z$ 与乘积 xy.

解 按两种方法求和：

(1) $fl(x+y+z)=fl(fl(x+y)+z)$

$=fl(fl(0.0000050556128\times 10^{2}+0.23162743\times 10^{2})-0.23162132\times 10^{2})$

（对阶，靠高阶）

$=fl(0.23163249\times 10^{2}-0.23162132\times 10^{2})=0.11170000\times 10^{-2}$

(2) $fl(x+y+z)=fl(x+fl(y+z))$

$=fl(0.50556128\times 10^{-3}+fl(0.23162743\times 10^{2}-0.23162132\times 10^{2}))$

$=fl(0.50556128\times 10^{-3}+0.61100000\times 10^{-3})=0.11165613\times 10^{-2}$

精确结果为 $x+y+z=0.111656128\times 10^{-2}$，显然，方法(2) 的结果较准确.

$fl(xy)=fl(0.50556128\times 10^{-3}\times 0.23162743\times 10^{2})=0.11710186\times 10^{-1}$

由上例可以看出，在计算机机器数系中，人们所熟悉的加减法的交换律与结合律是不成立的，特别在某些加法运算中，运算顺序对计算结果有很大影响. 关于这个问题，本书在后面的误差分析中还会谈到.

1.4 数值计算的误差

用数值计算方法来解决实际问题，不可避免地会产生误差. 数值分析的任务之一是将误差控制在一定的容许范围内或者至少对误差有所估计.

1.4.1 误差的来源与分类

1. 模型误差

数学模型与实际问题之间的误差称为模型误差.

一般来说，生产和科研中遇到的实际问题是比较复杂的，要用数学模型来描述，需要进行必要的简化，忽略一些次要的因素，这样建立起来的数学模型与实际问题之间一定有误差. 它们之间的误差就是模型误差.

2. 观测误差

实验或观测得到的数据与实际数据之间的误差称为观测误差或数据误差.

数学模型中通常包含一些由观测(实验)得到的数据，例如求书桌桌面的面积，需要测量书桌的长和宽，若用钢卷尺去测量(最小刻度为 mm)，测量出的数值与实际数值是有出入的(小于 $0.5mm$). 它们之间的误差就是观测误差.

3. 截断误差

数学模型的精确解与数值方法得到的数值解之间的误差称为方法误差或截断误差.

例如，由泰勒(Taylor) 公式得

$$e^x = 1 + x + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!} + R_n(x)$$

用 $p_n(x) = 1 + x + \frac{x^2}{2!} + \cdots + \frac{x^n}{n!}$ 近似代替 e^x，这时的误差就是截断误差，为

$$R_n(x) = \frac{e^{\xi}}{(n+1)!}x^{n+1}, \quad \xi \text{介于 } 0 \text{ 与 } x \text{ 之间}$$

4. 舍入误差

计算中遇到的数据可能位数很多或是无穷小数，如$\sqrt{2} = 1.41421356\cdots$，受机器字长的限制，无穷小数和位数很多的数必须舍入成一定的位数(符合机器字长). 舍入方法如下：

(1) 舍入法，如将 $1.41421356\cdots$ 四舍五入为 1.4142136；

(2) 截断法，如$\sqrt{2}$ 在 8 位字长的截断机里取成 1.4142135.

这样产生的误差称为舍入误差. 少量的舍入误差是微不足道的，但是在计算机作了成千上万次运算后，舍入误差的累积有时可能是十分惊人的. 它取决于算法的稳定性：如果算法能够累积大量的误差，这种算法是不稳定的，反之则是稳定算法.

研究计算结果的误差是否满足精度要求就是误差估计问题，本书主要讨论算法的截断误差与舍入误差. 其中，截断误差将结合具体算法讨论. 为分析数值运算的舍入误差，先要对误差的基本概念作简单介绍.

1.4.2　绝对误差与相对误差

定义 1.1　设 x 为准确值，x^* 为 x 的一个近似值，称 $e^* = x^* - x$ 为近似值的绝对误差，简称误差.

注意，这样定义的误差 e^* 可正可负，当绝对误差为正时近似值偏大，称为强近似值；当绝对误差为负时近似值偏小，称为弱近似值.

通常我们不能算出准确值 x，当然也不能算出误差 e^* 的准确值，只能根据测量工具或计算情况估计出误差的绝对值不超过某个正数 ε^*，也就是误差绝对值的一个上界. ε^* 称为近似值的误差限，它总是正数.

一般情形 $|x^* - x| \leqslant \varepsilon^*$，即 $x^* - \varepsilon^* \leqslant x \leqslant x^* + \varepsilon^*$. 这个不等式有时也表示为 $x = x^* \pm \varepsilon^*$.

例如用卡尺测得一个圆杆的直径为 $x^* = 350\text{mm}$，它是圆杆直径 x 的近似值，由卡尺的精度知道这个近似值的误差不会超过半个毫米，则有

$$|x^* - x| = |350 - x| \leqslant 0.5(\text{mm})$$

于是该圆杆的直径为

$$x = 350 \pm 0.5(\text{mm})$$

用 $x = x^* \pm \varepsilon^*$ 表示准确值可以反映它的准确程度，但不能说明近似值的好坏. 例如，测量一根 10cm 长的圆钢时发生了 0.5cm 的误差，和测量一根 10m 长的圆钢时发生了 0.5cm 的误差，其绝对误差都是 0.5cm，但是，后者的测量结果显然比前者要准确得多. 这说明决定一个量的近似值的好坏，除了要考虑绝对误差的大小，还要考虑准确值本身的大小，这就需要引入相对误差的概念.

定义 1.2 设 x 为准确值，x^* 为 x 的一个近似值，近似值 x^* 的误差 e^* 与准确值 x 的比值 $\dfrac{e^*}{x} = \dfrac{x^* - x}{x}$ 称为近似值 x^* 的相对误差，记为 e_r^*.

在实际计算中，由于真值 x 总是不知道的，通常取 $e_r^* = \dfrac{e^*}{x^*} = \dfrac{x^* - x}{x^*}$ 作为 x^* 的相对误差，条件是 $e_r^* = \dfrac{e^*}{x^*}$ 较小，此时

$$\frac{e^*}{x} - \frac{e^*}{x^*} = \frac{e^*(x^* - x)}{x^* x} = \frac{(e^*)^2}{x^*(x^* - e^*)} = \frac{(e^*/x^*)^2}{1 - e^*/x^*}$$

是 e_r^* 的平方项级，故可忽略不计.

相对误差也可正可负，它的绝对值上界称为相对误差限，记为 ε_r^*，即 $\varepsilon_r^* = \dfrac{\varepsilon^*}{|x^*|}$.

在上面的例子中，前者的相对误差是 $\dfrac{0.5}{10} = 0.05$，而后者的相对误差是 $\dfrac{0.5}{1000} = 0.0005$. 一般来说，相对误差越小，表明近似程度越好.

1.4.3 有效数字

有效数字是近似值的一种表示法，它既能表示近似值的大小，又能表示其精确

程度.

定义 1.3　若近似值 x^* 的误差限是某一位的半个单位，该位到 x^* 的第一位非零数字共有 n 位，则称近似值 x^* 有 n 位有效数字.

在科学记数法中，将近似值 x^* 写成规范化形式为

$$x = \pm 0.a_1a_2\cdots a_n\cdots \times 10^m \tag{1.9}$$

其中 m 为整数，$a_1 \neq 0$，$a_i(i=1,2,\cdots)$ 为 $0\sim 9$ 之间的整数. 按照定义 1.3，近似值 x^* 有 n 位有效数字当且仅当

$$|x^* - x| \leqslant \frac{1}{2}\times 10^{m-n} \tag{1.10}$$

因此在 m 相同的情形下，n 越大则误差越小，亦即一个近似值的有效位数越多其误差限越小.

【例 1.6】　已知 $\pi = 3.1415926\cdots$，若取近似值为 $x_1^* = 3.14$ 和 $x_2^* = 3.1416$，则 x_1^* 与 x_2^* 有几位有效数字？

解　对 $x=\pi$ 取前 3 位，$x_1^* = 3.14$，$\varepsilon_1^* \leqslant 0.002$；取前 5 位，$x_2^* = 3.1416$，$\varepsilon_2^* \leqslant 0.00001$，它们的误差限都不超过近似值 x_1^* 与 x_2^* 末位数的半个单位，即

$$|\pi - 3.14| \leqslant \frac{1}{2}\times 10^{-2} = \frac{1}{2}\times 10^{1-3}$$

$$|\pi - 3.1416| \leqslant \frac{1}{2}\times 10^{-4} = \frac{1}{2}\times 10^{1-5}$$

所以，用 $x_1^* = 3.14$ 近似 π 有 3 位有效数字，用 $x_2^* = 3.1416$ 近似 π 有 5 位有效数字. 一般地，在 x 有多位数字时，若取前面有限位数的数字作近似值，都是采用四舍五入的原则.

【例 1.7】　按四舍五入原则写出下列各数具有 5 位有效数字的近似值：

$$187.9325, 0.03785551, 8.000033, 2.7182818$$

解　按定义，上述各数具有 5 位有效数字的近似值分别是

$$187.93, 0.037856, 8.0000, 2.7183$$

注意，$x = 8.000033$ 的 5 位有效数字的近似值是 8.0000 而不是 8，因为 8 只有 1 位有效数字.

【例 1.8】　重力加速度 g，如果以 m/s^2 为单位，$g \approx 0.980\times 10^1 m/s^2$；若以 km/s^2 为单位，$g \approx 0.980\times 10^{-2} km/s^2$，它们都具有 3 位有效数字.

解　按第一种写法

$$|g - 9.80| \leqslant \frac{1}{2}\times 10^{-2} = \frac{1}{2}\times 10^{1-3}$$

按第二种写法

$$|g - 0.00980| \leqslant \frac{1}{2}\times 10^{-5} = \frac{1}{2}\times 10^{-2-3}$$