- 国家特色专业建设点建设项目
- 广东省高等学校重点专业建设项目
- 广东省本科教学改革立项项目
- 华南师范大学卓越计划项目

数学实验系列教程

系列教程主编/冯伟贞

信息与计算科学实验教程

XINXI YU JISUAN KEXUE SHIYAN JIAOCHENG

主编 杨坦 谢骊玲

华南理工大学出版社
SOUTH CHINA UNIVERSITY OF TECHNOLOGY PRESS

数 学 实 验 系 列 教 程

系列教程主编/冯伟贞

信息与计算科学
实验教程

XINXI YU JISUAN KEXUE SHIYAN JIAOCHENG

主 编 杨 坦 谢骊玲

华南理工大学出版社
SOUTH CHINA UNIVERSITY OF TECHNOLOGY PRESS
·广州·

内容简介

本书不是一本传统意义上的数值分析（或计算方法）课程的实验教材，没有按照定义、定理、算法、函数语法说明和实验题目这样的顺序组织内容。每章分为理论部分和实验部分。理论部分强调算法的来源、各种方法间的联系、每个算法优缺点的讨论以及算法的改进与完善，基本覆盖了目前国内本科计算方法课程中的内容。本书选择 MATLAB 作为数值算法的实现平台，在给出算法实现细节的同时介绍了 MATLAB 中与该算法相关的函数。这样的设计使得本书的难度具有层次性和一定的弹性。工科专业的学生可以理解算法的原理、掌握数值计算工具解决实际工程问题。信息与计算科学专业的学生则要求完成更有深度的算法设计问题。本书有相当多的延伸内容是以实验的形式出现的，以促使学生能够在解决实际问题过程中开拓眼界，学到新的知识。本书的实验内容铺排与一般计算方法课程教学展开同步，使教学双方能够在日常教学中同步融入实验。本书可作为信息与计算科学专业本科数值计算方法课程或其他有数值计算需求的工科专业的相关课程的配套实验教材或参考教材。

图书在版编目（CIP）数据

信息与计算科学实验教程/杨坦，谢骊玲主编. —广州：华南理工大学出版社，2012.7
（数学实验系列教程/冯伟贞主编）
ISBN 978-7-5623-3706-5

Ⅰ. ①信… Ⅱ. ①杨… ②谢… Ⅲ. ①信息技术-实验-高等学校-教材 ②计算机科学-实验-高等学校-教材 Ⅳ. ①G202-33 ②TP3-33

中国版本图书馆 CIP 数据核字（2012）第 171554 号

信息与计算科学实验教程
杨　坦　谢骊玲　主编

出版发行：华南理工大学出版社
（广州五山华南理工大学 17 号楼，邮编 510640）
http://www.scutpress.com.cn　E-mail：scutc13@scut.edu.cn
营销部电话：020-87113487　87111048（传真）
策划编辑：何丽云
责任编辑：兰新文
印 刷 者：广州市穗彩彩印厂
开　　本：787mm×1092mm　1/16　**印张**：12.5　**字数**：259 千
版　　次：2012 年 7 月第 1 版　2012 年 7 月第 1 次印刷
定　　价：26.00 元

《数学实验系列教程》

序　　言

实验是科学家根据确定的认识目的，应用特定的物质手段，对认识对象进行控制，使对象按照自身的意愿发生变化，从而对认识对象进行观察和分析的认识方法。因此，实验是获取认知经验、探求对认识对象的控制手段或技术的重要途径。

数学具有经验与演绎二重性，“数学的源泉就在于思维与经验的反复出现的相互作用”（D. 希尔伯特），“数学有两个侧面，一方面它是欧几里得式的严谨科学，从这个方面看数学像是一门演绎科学；但另一方面，创造过程中的数学看起来却像一门试验性的归纳科学”（G. 波利亚，《数学与猜想》）。

数学具有科学与技术两种品质。一般认为科学回答“是什么与为什么”，技术说明“做什么和怎么做”。应用数学大致分成三类。第一类是应用数学的理论基础研究；第二类是数学与别的学科领域的交叉渗透，以解决该学科中重要数学结构有关问题为目的；第三类面向国民经济系统、军事系统和社会发展系统，以解决这三大系统中提出的现实问题为目标。在第二、第三类的数学应用推进过程中，数学建模、数据挖掘、数值计算、统计、仿真等数学技术获得突飞猛进的发展。有众多的实例说明，数学技术已经在多个领域成为直接的生产力。计算机的本质是数学机器，它使数学知识与数学实践活动得到有效联结，并在最近的几十年使数学技术得到空前的发展。发展至今天的数学技术的运用与进一步发展对计算机有强烈的依赖。

数学的经验性及技术品质凸显了数学发展对实验的需求。1988 年，美国 Rossciacr 技术学院正式引入数学实验课。1989 年，美国的 Mount Holyke College 数学系集体编写了第一本专门教材《数学实验室》。新事物的诞生引起了广泛的关注，并在“什么是数学实验”这个问题上形成了百家争鸣的局面，也因此产生了不同定位的数学实验教材。

本系列教材的编者对以下数学实验定义有较多的认同：“数学实验是在一定的数学思想和数学理论指导下，经过某种预先的组织设计，借助于一定的仪器和技术手段，进行数学化操作，包括对客观事物的数量化特性进行观察、抽样、测试、检验、逼近、仿真等，进而解决数学和科学问题的一类科学研究方法。”这段定义强调的是数学实验在数学研究过程中有“经验获取”的重要地位，提出“客观事物的量化特性”为数学实验的对象，“观察、抽样、测试、检验、逼近、仿真”等是数学实验的主要方法，强调数学实验中“数学技术手段”、“实验工具”运用的重要性。这段定义对数学实验的本质作了较好的阐释。

作为数学教育过程中的数学实验活动，G. 波利亚作出如下解释："数学实验活动是在一定的（数学学习、研究、发现）目标的指导下，对具有一定数学意义的实物、模型、事物，以及数字、图形、式子、题目等，进行观察、变换、制作、演示、求解以获取感性认识和数学信息的活动，而这实际上也就是一种数学研究活动（的初级阶段）。"这段解释与前面的数学实验的定义本质上基本吻合，但更强调的是：数学实验，作为数学教育过程中的一种"活动"，是为学生建造的一个直接感知数学、获取经验、形成基本数学技能的重要平台。

"数学实验系列教程"的编写侧重于为大学生建造数学学习中必要的数学实验活动平台。本系列教程以多种背景下的量化模式与结构为实验对象，以计算机为主要实验工具展开论述。教程中的实验设计力图让学生"获取对数学的感性认识和数学信息"，了解数学实验的方法。教程中的实验设计强调计算机技术的运用，并融入了初步的数学技术运用体验。

"数学实验系列教程"包括《数学基础实验教程》、《金融数学与金融工程实验教程》、《信息与计算科学实验教程》及《统计学实验教程》。《数学基础实验教程》的内容与"高等数学"及"线性代数"两门课程相对应，适用于数学专业及非数学专业学生。《金融数学与金融工程实验教程》、《信息与计算科学实验教程》及《统计学实验教程》的内容分别和金融数学与金融工程、信息与计算科学、统计学三个专业（方向）主干课程相对应。《金融数学与金融工程实验教程》也可作为一般经济类专业在强化量化处理方法学习方面的参考用书；《信息与计算科学实验教程》是基本数值计算方法计算机实现的学习用书，对计算机专业及其他一般理工科学生而言也有重要参考价值；《统计学实验教程》可作为"概率论与数理统计"、"统计学"课程教学的参考用书。

本系列教材能兼顾教学上开展同步实验及阶段性实验的要求，也可作为数学各专业方向独立开设实验课程的主教材。各分册编写组在编写过程中对相应数学方向的实验思想、方法及手段作了认真的思考及提炼，并充分考虑实验者的基础及能力作内容铺排，使各分册内容具有相应专业（方向）的特色。

《数学实验系列教程》的编写得到国家特色专业建设点建设项目、广东省高等学校重点专业建设项目、广东省本科教学改革立项项目《数学技术人才培养模式的构建与实践》及华南师范大学卓越计划项目的支持。

在编写过程中，编者感受到，我国的数学实验教学还处在一个亟待完善的阶段。本系列教程的编写与使用也是一种积极的探索和体验。本系列教程中难免有疏漏和不足之处，敬请读者批评指正，以使我们能够获得更大进步。

冯伟贞

2012年5月于华南师范大学

前　言

科学计算或数值分析是信息与计算科学专业的核心内容之一，主要解决在计算机上求得科学与工程计算问题的近似解问题，所涉及的主要问题包括函数求值、数值逼近、数值积分、方程求解、特征值问题和数值优化等。本书基本覆盖了目前国内本科计算方法课程中的内容，包括：非线性方程求根、线性方程组求解、矩阵特征值计算、插值与拟合、数值积分和数值微分、常微分方程数值方法以及快速傅里叶变换与 JPEG 编码。

现有的数值分析实验教材中有两种比较典型的情况。一种是按照定义、定理、算法、例题和上机题目的顺序组织内容，基本上是数值分析教科书内容的精缩外加上机题目的形式。另一种则基于某种科学计算软件，如 MATLAB 或 Mathematica，讲述该软件中各种处理数值计算问题的函数的用法，类似于该软件数值计算工具箱的使用手册。第一种类型比较传统，内容抽象，算法实现起来难度比较高。第二种类型适合对数值分析有入门要求的工科专业学生，可以迅速掌握计算工具以解决一些实际问题，但对算法的理解不够深刻。本书力图在这两种“极端”情况中取得一个平衡，既生动有趣，又不失严谨。

本书提供较多的实际背景。很多作者强调理论上的完整性，却忽视了数值分析这一学科的魅力亦来自于科学和工程计算中的实际问题。源自于计算数学专业的信息与计算科学专业 2005 年成为我国理科规模最大的本科专业之一，开设该专业的院校数和进入该专业的学生数大大增加。培养数值计算的应用型人才在专业培养方案中的比例越来越重。该专业众多的本科毕业生不会大量地进入传统的计算数学方向的研究生学习阶段，而是从事相关领域的应用性工作或者攻读其他专业方向的研究生学位。这个大背景要求强调数值分析课程的入门性质和趣味性，吸引更多的学生了解和掌握数值分析这一有力工具，会使用计算机解决实际的科学与工程计算问题。传统的数值分析教材和相应的实验教材，问题的背景往往被高度地做了数学上的抽象。本书引入了较多的实际问题，不少是信息技术领域的现代内容。这一方面使得学习变得更有趣，另一方面是想强调计算数学作为一门应用性的学科，实际的科学与工程计算问题是它不断向前发展的重要推动力。掌握了数值分析的理论，并不意味着形成了解决科学与工程中提出的实际计算问题的能力。而减小两者之间的差距，是本书的写作目的之一。

本书强调问题解决的策略。从数值分析课本中的一个理论，到一个高效、可靠、经得起实践检验的算法，中间有很长的距离。数值计算理论好比软件设计中

的概念模型，相对抽象，而将要运行在计算机上的代码可以类比物理模型。考虑到计算的稳定性、精度、速度和存储空间，以及所要解决的计算问题的特性，同一个概念模型有多种不同的物理实现，其中有些问题可以使用理论进行分析，还有一些问题更强调解决问题的思路和策略。以第 1 章非线性方程求根为例，传统的数值分析课本都会给出二分法、割线法、牛顿法的推导和收敛性、收敛速度的分析。大名鼎鼎的牛顿法在数值计算领域有着非常广泛的应用，但在非线性方程求根方面，牛顿法由于收敛条件不便于验证，丧失了鲁棒性。因为无法保证使用牛顿法一定可以可靠地求得方程在给定初始值附近的根，所以在 MATLAB 中并没有内置一个实现在理论上有非常重要地位的牛顿法的函数。这说明目前很多的数值计算教科书还是偏重理论分析，而并没有让学生了解到它和科学计算所要求的普适性与实用性之间的区别。MATLAB 求解非线性方程根的内置函数 fzero 中并没有使用牛顿法，而是结合了二分法和逆二次插值法的一个自适应算法。解决实际问题时所采用的策略和逻辑，在作者看来同算法的收敛性、稳定性等方面的理论分析同样重要，而这种观点，并不是现有主流实验教材所体现出来的。

本书在引入一个算法之后更注重分析方法之间的联系，引导学生进行思考。例如对 A 算法，它有哪些优缺点？它适合所要解决的实际问题吗？算法有没有关键性的假设？为了克服它的缺点，该从什么方面进行改进？它与 B 算法在解决问题的思路上有哪些联系与区别？与目前常见的直接列出定理、公式和算法的实验教材相比，本书语言生动，加入作者对问题的分析和理解，在比较深入地分析基本算法后，不断地对算法进行改进和完善。

本书对很多实际计算中的重要问题作了比较深入的介绍。在常微分方程数值解法这一部分中，数值方法的误差控制在实际计算时是非常重要的一个问题，它决定了如何确定合适的步长以及得到的数值解能否被采用。传统教材更多关注算法的构造及其稳定性、收敛性、精度方面的分析，而对如何进行误差的估计，以及如何通过调整步长来实现误差的控制重视不足。反观 MATLAB 的内置微分方程求解函数，在这方面下足了工夫。和传统教科书在上机练习中给出龙格 - 库塔方法的计算公式及固定步长要求学生上机计算不同，MATLAB 中的 ode 系列函数提供了丰富的选项和功能，以及复杂的处理逻辑，让我们从一个侧面了解到实际科学与工程计算对算法提出的具体要求。本书详细地讨论了各种算法在处理误差控制问题上的策略。

本书在实验的设计方面，强调算法的实现，给出了比较详细的实现的引导。实验的难度具有一定的弹性，可适用于多种不同的专业要求。作为数值分析课程的实验，自然是遵循算法设计、算法实现、性能分析等内容。对于信息与计算科学专业的学生，算法的设计和理论分析是必不可少的。要理解算法的原理和实现细节，并针对不同情况作应有的改进，学生必须自己动手编写、调试函数和程

序。对于工科专业或其他只需掌握数值计算工具的专业的学生，除了了解算法的基本原理外，更多的是要熟练使用已有的 MATLAB 函数和各种工具，改进和优化求解过程。

本教材选用 MATLAB 作为数值分析课程的实验平台，目前要求使用者具备初步的 MATLAB 应用能力。有关的实验教材主要有两种实验平台，多数使用 MATLAB，少数使用 C、C + + 或 Mathematica。MATLAB 界面友好、易学，有方便和强大的绘图功能，代码的执行使用解释方式，便于交互。这些特点决定了 MATLAB 非常适合作为数值实验的平台。MATLAB 可以说是目前商业上最成功的数值计算软件，提供了丰富的专业函数，它的有效性和可靠性经受了大量工业用户的检验。在某些领域，MATLAB 的工具箱已经成为行业标准的软件包。MATLAB 同时也是一个非常开放的软件系统，允许用户编写自己的函数。互联网上有相当多的有关 MATLAB 的资源，包括各种组织和个人开发的工具箱，以及大量的讨论组。对初学者来说，使用互联网是获得帮助和提高的重要手段。

本书在给出各种算法细节的同时，也介绍了使用对应的 MATLAB 内置函数的实现方法。这样做有 4 个好处。

第一，当我们编写代码实现一个算法后，可以使用 MATLAB 来检验自己结果的正确性。MATLAB 作为商业化的“工业软件”，对一般规模的计算问题，在正确使用的前提下，计算结果可以作为检验算法正确性与性能的一种可靠手段。

第二，MATLAB 中的函数是我们学习的典范。MATLAB 函数不是简单地把数值算法翻译成代码，它在诸如防错性设计结构、多输入多输出、自适应处理、向量化处理等程序设计方面值得我们深入学习。MATLAB 代码体现了简洁、稳定、高质量的工业化软件标准。

第三，在 MATLAB 平台下数值实验的难度具有相当的弹性。例如第 3 章中求解矩阵特征值的 QR 迭代方法。工科专业的学生，可能只需要掌握现成的 eig 函数的使用方法，通过一些数值算例了解算法的性能和优化方法即可。如果要再深入探究其中的过程，可以使用 MATLAB 内置的 QR 分解函数 qr，或者更进一步使用命令“gallery('house',x,k)”生成相应的 Householder 变换来实现矩阵的 QR 分解，还可以使用函数 hess 直接得到一个矩阵的上 hessenberg 形式。对于要从事数值算法设计和开发的信息与计算科学专业的学生，则必须自己实现并测试上面所有这些算法。

第四，MATLAB 是一个开放的数值计算平台，允许我们修改已有的代码和编写新的函数。虽然它并不是开源软件，部分函数实现的细节无法查看，但大多数函数是可以查看源代码的。学习 MATLAB 可以沿着读懂程序、修改已有程序到编写自己的程序这样的过程进行。

本书的每个章节分为理论部分和实验部分。理论部分强调算法的来源、各种

方法间的联系、每个算法优缺点的讨论和算法的完善与发展，并不过分强调理论的完整性和严密性，更多的是让学生对相关算法形成一个直观的感受和思想方法上的整体把握。正文中的内容非常基本、易懂，涵盖了数值分析课程的主要内容。对学生而言，更多的时间应该用在实验部分，有相当多的延伸内容是以实验的形式出现的，在解决实际问题过程中学到的知识，会更鲜活和生动。

本书可作为信息与计算科学专业本科数值计算方法课程或其他有数值计算需求的工科专业的计算方法课程的配套实验教材或参考教材。

本书的第 1、第 3、第 6、第 7 章由杨坦编写，第 2、第 4、第 5 章由谢骊玲编写。全书由杨坦总纂定稿。

在本书的编写与出版过程中，华南师范大学数学科学学院和华南理工大学出版社给予了很大的支持和帮助。特别是数学科学学院的冯伟贞教授，从教材的立意到写作风格都给两位作者提出了很多宝贵的意见。感谢所有使得本书能够顺利付梓的同事和家人。

限于编者的经验和水平，书中一定有疏漏和不足之处，敬请读者批评指正。

编　者

2012 年 5 月

目　录

第 1 章　非线性方程求解

求非线性方程 $f(x)=0$ 的解或非线性函数 $f(x)$ 的零点是非常普遍的计算问题. 但遗憾的是，除了少数特殊的方程，如 5 次以下的多项式方程外，大多数非线性方程的解没有解析表达式，只能借助于数值方法求得它们的近似值.

求解非线性方程通常使用迭代算法：从一个粗糙的估计值开始逐步逼近精确解. 在计算机上迭代算法通过循环结构来实现. 对于一个循环而言，有 3 个最重要的问题需注意：①如何提供一个初始值以便开始迭代；②循环的每一步如何改善近似解的精度；③无论迭代是否收敛，为了正常地结束循环，这个逐步求精的过程应该在何时停止，也就是循环的终止条件. 明确这些也就导出了这类问题的求解策略：首先找到根的初始估计，回答根在哪里的问题，然后通过数值算法求出满足精度要求的近似值，回答解是多少的问题. 按照程序设计中逐步求精的策略，尽可能地将这个过程中相对独立的步骤编制成函数，并定义好接口，方便设计更复杂的算法.

特别需要注意的是，在求得数值解 x 后，最好将 x 代回方程中，检查左端项 $f(x)$ 与右端项 0 的接近程度. 有时会发现求到的解实际上是函数 $f(x)$ 的奇点.

本章的主要内容包括非线性方程求根的二分法、割线法、牛顿法、反二次插值法、有理函数逼近法以及综合多种方法的自适应求解算法. 在这一章的实验中还介绍了针对迭代序列的加速方法，以及柯西迭代法. 实验（20）是本章的综合性设计性实验，要求大家在上述算法的基础上，设计并实现一个稳健的自适应求根算法.

1.1　根的隔离

这里的算法从一个估计的包含解的区间开始. 确定隔根区间就是把用户指定的原始区间划分成若干个小的子区间，每个子区间中都包含至少一个 $f(x)$ 的根. 这样的每个小区间称作隔根区间. 如果是通过人工交互来确定，最简单的办法是使用 plot 绘图命令绘制出函数在给定区间的图像，通过观察来确定隔根区间. 现在想在计算机上通过程序来自动确定，首先得把“某区间包含 $f(x)$ 的一个根”这件事用表达式表示出来. 如果一个子区间内包含一个根，直观的理解是函数值在子区间的两个端点处取不同的符号（注意这在逻辑上是有漏洞的!），这很容易通过例如 $f(a)*f(b)<0$ 这样的比较表达式来进行判断.

下面给出确定隔根区间的算法描述.

算法 1.1

输入：$f(x)$，$x_{\min}$，$x_{\min}$，n

$\delta=(x_{\max}-x_{\min})/n$

$a=x_{\min}$

for $m=0:n-1$

  $b=a+\delta$

  if $f(x)$在$[a,b]$上改变符号

    保存$[a,b]$

    $a=b$

  end

end

输出：保存有根的小区间的端点为数组，小区间的两个端点分别保存在同一列的两行中；若没有隔根区间，则返回空数组.

对于不是 MATLAB 内置函数的 $f(x)$，通常使用 m 文件函数或匿名函数的形式来定义. $x_{\max}$和 $x_{\min}$为初始区间的两个端点，n 为对初始区间的等分份数. n 的取值大小对结果的影响较大，如太小，可能会漏掉一些解，但也不宜取得太大.

在计算机上实现算法 1.1，还有很多细节需要分析. “$f(x)$在$[a,b]$上改变符号”当然可以使用

```
if  f(a) * f(b)<0
……
```

来判断，但这样处理有两个潜在的问题. 第一，在区间两个端点处函数值符号不同只适用于过零点类型的根，也就是在根的左右函数图像分布在 x 轴的两侧. 对于某些偶次重根，例如 $f(x)=x^2$，这种判断方法就不再适用. 所以上述的算法是有前提假设的，在实际问题的计算中，首先要分析函数 $f(x)$的性质，例如作图观察，生搬硬套算法往往会得到错误的结果. 另外一个问题涉及数值在计算机中运算的问题，如果 $f(a)$和 $f(b)$的绝对值都非常小，两者的乘积可能产生下溢从而导致符号判断的失误. 使用 MATLAB 内置函数 sign 进行符号的判断是一种稳妥的实现方式，无论 $f(a)$和 $f(b)$的绝对值有多小，

```
sign(f(a))~=sign(f(b))
```

总能给出正确的结果.

对于算法的输出，为了和后面的程序衔接，需将求到的所有隔根区间的两个端点保存在一个二维数组中. 如在用户指定的区间内没有根，算法的输出值是一个空数组. 这可以通过下述代码实现，首先生成一个空数组：

```
X = [ ]
```

如果在某次循环中，小区间通过了算法的验证，则将小区间的两个端点值添加到数组 X 中. 实际上是在原有数据的右侧插入 2 行 1 列共 2 个数据，分别是隔根区间的左、右端点值. 在 MATLAB 中可以使用下述方式动态地扩充数组维数：

```
X = [X,[a,b]']
```

$[a,b]$后的撇号“ ′ ”表示转置，所以$[a,b]'$表示一个列向量.

1.2　二分法

得到一个包含根的小区间后，二分法可以逐步逼近准确解. 二分法的思想非常简单，如果函数 $f(x)$连续，且在区间的两个端点 a、b 上函数值 $f(a)$与 $f(b)$异号，则在区间$[a,b]$内必有一个过零点类型的根. 当使用算法 1.1 求出一个包含根的区间后，如果对该区间做平分，根要么出现在分点上，要么在等分的两个子区间内. 后一种情况下再次根据 $f(x)$的符号保留包含根的那个子区间，继续对该子区间做平分. 重复上述过程，将得到一个区间序列. 每操作一次所得的子区间长度减半，算法保证根始终包含在这个区间序列内. 如果在实数域内处理，这样的操作很可能会无限进行下去，区间的极限是一个点，而这个点就是方程的根.

在计算机上实现算法时，我们更关心这个过程是否能在有限步中结束，而不是陷入死循环. 幸运的是，在计算机浮点数系中，上述过程一定是有限的（见实验（4））. 具体地说，如果使用双精度浮点数进行运算，最多经过 52 次二分后所得子区间的长度就会变为0，迭代终止. 和后面将要介绍的算法相比，二分法的收敛速度很慢，在这个算法中虽然每一步求出一个新的函数值，但二分过程只使用函数值的符号信息（可以理解为如果只知道函数 $f(x)$在区间的两个端点异号，那么只好近似地估计解在区间的中点）. 二分法的优点是非常可靠，只要函数连续、初始区间满足要求，二分法一定会得到区间内使用浮点数所能表示的最接近根的近似值.

算法 1.2　二分法

初始化：给定初始区间$[a,b]$

for　$n=1,2,\cdots$

　　$x_n=a+(b-a)/2$

　　if　$f(x_n)==0$

　　　　跳出循环,返回 x_n 的值

end

if $f(x_n)==0$ 的符号与 $f(a)$ 的符号相同

$a=x_n$

else $b=x_n$

end

if x_n 收敛

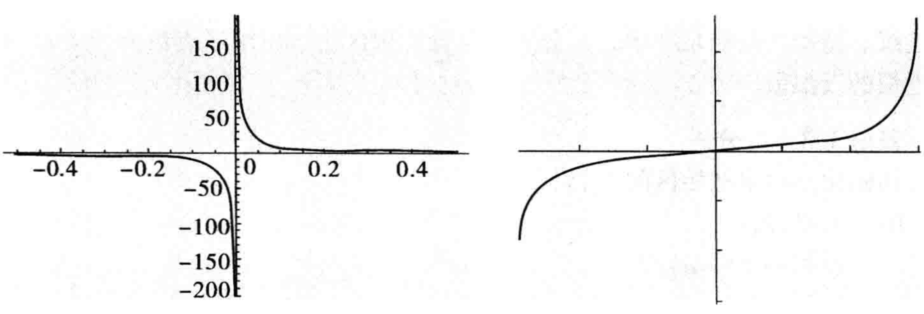
退出循环

end

end

循环内的第一项任务是对区间做二分，求出区间的中点. 你可能会不假思索地使用 $x_n=(a+b)/2$ 来实现. 如果熟悉计算机数系的话，更好的方式是令 $x_n=a+(b-a)/2$. 这两种表达式在实数域上是等价的，但在计算机所使用的浮点数系上却有较大差别. 第一，$(a+b)/2$ 的值可能并不在 a 和 b 之间. 例如在有 3 位有效数字的机器上，$(72.9+72.8)/2$ 的结果是 73.0. 第二，当 a 和 b 足够大时，$a+b$ 的值可能会超出计算机所能表示的最大值，产生所谓的“上溢”.

如何表达算法中的“x_n 收敛”，即循环的终止条件也是一个重要的问题. 在数值计算中，并不会使用数值解序列 $\{x_n\}$ 在极限意义下收敛这样的概念. 数值计算是有限精度和有限步骤的，它不追求得到方程根的精确值（一些求精确解的任务可以使用符号计算实现）. 根据实际问题的意义，只要数值解达到给定的有效数字位数或精度就可以了. 理论上可以给定一个称为容差的值 ε，如果第 n 步得到的数值解 x_n 与精确解 x^* 之间满足 $|x_n-x^*|<\varepsilon$，则结束循环. 困难在于精确解 x^* 是未知的，实践中不能直接使用上述准则. 二分算法较为特殊，它提供了一种内在的误差估计. 由于确保二分过程中根始终包含在区间序列中，二分法每一步输出的近似值 x_n 是区间的中点，那么 x_n 和精确解 x^* 的距离不会超过区间长度的一半. 只要迭代足够多次，子区间的长度就足够小，x_n 就充分靠近 x^*. 因此二分法可以利用子区间长度来估计和控制容差.



(a)在0两侧距离非常近的两个x，对应的y相差巨大

(b) 在原点附近y随着x的增加非常缓慢的变化

图 1.1 收敛准则的误判情况