21st CENTURY
实用规划教材

21世纪全国应用型本科计算机案例型规划教材

赠送 电子课件 www.pup6.com

赠送 教学参考资料 www.pup6.com

实用计算方法

徐亚平　编著

北京大学出版社
PEKING UNIVERSITY PRESS

21 世纪全国应用型本科计算机案例型规划教材

实用计算方法

徐亚平　编著

北京大学出版社
PEKING UNIVERSITY PRESS

内容简介

本书共分8章，内容包括：数值计算基本概念，插值与数据拟合方法，导数应用及近似计算，定积分应用及近似计算，方程求根数值方法，线性方程组数值解法，线性规划问题及解法，矩阵特征值与特征向量。

本书从历史背景、知识回顾、实际应用、求解方法和算法实现(用C语言)5个方面介绍各章的相关内容。

本书阐述简明易懂，注重理论联系实际，可作为大学计算机及有关专业的教材，也适合其他理工科专业计算方法课程使用，还可作为从事与数值分析相关人员的参考工具。

图书在版编目(CIP)数据

实用计算方法/徐亚平编著．—北京：北京大学出版社，2011.3

(21世纪全国应用型本科计算机案例型规划教材)

ISBN 978-7-301-18538-4

Ⅰ.①实… Ⅱ.①徐… Ⅲ.①计算方法—高等学校—教材 Ⅳ.①O241

中国版本图书馆CIP数据核字(2011)第022159号

书　　　名：实用计算方法
著作责任者：徐亚平　编著
责 任 编 辑：郑　双
标 准 书 号：ISBN 978-7-301-18538-4/TP・1154
出　版　者：北京大学出版社
地　　　址：北京市海淀区成府路205号　100871
网　　　址：http://www.pup.cn　http://www.pup6.com
电　　　话：邮购部62752015　发行部62750672　编辑部62750667　出版部62754962
电 子 邮 箱：pup_6@163.com
印　刷　者：三河市富华印装厂
发　行　者：北京大学出版社
经　销　者：新华书店
787毫米×1092毫米　16开本　11.5印张　255千字
2011年3月第1版　2011年3月第1次印刷
定　　　价：24.00元

信息技术的案例型教材建设

(代丛书序)

刘瑞挺

北京大学出版社第六事业部在2005年组织编写了《21世纪全国应用型本科计算机系列实用规划教材》，至今已出版了50多种。这些教材出版后，在全国高校引起热烈反响，可谓初战告捷。这使北京大学出版社的计算机教材市场规模迅速扩大，编辑队伍茁壮成长，经济效益明显增强，与各类高校师生的关系更加密切。

2008年1月北京大学出版社第六事业部在北京召开了“21世纪全国应用型本科计算机案例型教材建设和教学研讨会”。这次会议为编写案例型教材做了深入的探讨和具体的部署，制定了详细的编写目的、丛书特色、内容要求和风格规范。在内容上强调面向应用、能力驱动、精选案例、严把质量；在风格上力求文字精练、脉络清晰、图表明快、版式新颖。这次会议吹响了提高教材质量第二战役的进军号。

案例型教材真能提高教学的质量吗？

是的。著名法国哲学家、数学家勒内·笛卡儿(Rene Descartes，1596—1650)说得好：“由一个例子的考察，我们可以抽出一条规律。(From the consideration of an example we can form a rule.)”事实上，他发明的直角坐标系，正是通过生活实例而得到的灵感。据说是在1619年夏天，笛卡儿因病住进医院。中午他躺在病床上，苦苦思索一个数学问题时，忽然看到天花板上有一只苍蝇飞来飞去。当时天花板是用木条做成正方形的格子。笛卡儿发现，要说出这只苍蝇在天花板上的位置，只需说出苍蝇在天花板上的第几行和第几列。当苍蝇落在第四行、第五列的那个正方形时，可以用(4，5)来表示这个位置……由此他联想到可用类似的办法来描述一个点在平面上的位置。他高兴地跳下床，喊着“我找到了，找到了”，然而不小心把国际象棋撒了一地。当他的目光落到棋盘上时，又兴奋地一拍大腿：“对，对，就是这个图”。笛卡儿锲而不舍的毅力，苦思冥想的钻研，使他开创了解析几何的新纪元。千百年来，代数与几何，井水不犯河水。17世纪后，数学突飞猛进的发展，在很大程度上归功于笛卡儿坐标系和解析几何学的创立。

这个故事，听起来与阿基米德在浴池洗澡而发现浮力原理，牛顿在苹果树下遇到苹果落到头上而发现万有引力定律，确有异曲同工之妙。这就证明，一个好的例子往往能激发灵感，由特殊到一般，联想出普遍的规律，即所谓的“一叶知秋”、“见微知著”的意思。

回顾计算机发明的历史，每一台机器、每一颗芯片、每一种操作系统、每一类编程语言、每一个算法、每一套软件、每一款外部设备，无不像闪光的珍珠串在一起。每个案例都闪烁着智慧的火花，是创新思想不竭的源泉。在计算机科学技术领域，这样的案例就像大海岸边的贝壳，俯拾皆是。

事实上，案例研究(Case Study)是现代科学广泛使用的一种方法。Case包含的意义很广：包括Example例子，Instance事例、示例，Actual State实际状况，Circumstance情况、事件、境遇，甚至Project项目、工程等。

我们知道在计算机的科学术语中，很多是直接来自日常生活的。例如Computer一词早在1646年就出现于古代英文字典中，但当时它的意义不是“计算机”而是“计算工人”，

即专门从事简单计算的工人。同理，Printer 当时也是“印刷工人”而不是“打印机”。正是由于这些“计算工人”和“印刷工人”常出现计算错误和印刷错误，才激发查尔斯·巴贝奇(Charles Babbage，1791—1871)设计了差分机和分析机，这是最早的专用计算机和通用计算机。这位英国剑桥大学数学教授、机械设计专家、经济学家和哲学家是国际公认的“计算机之父”。

20 世纪 40 年代，人们还用 Calculator 表示计算机器。到电子计算机出现后，才用 Computer 表示计算机。此外，硬件(Hardware)和软件(Software)来自销售人员。总线(Bus)就是公共汽车或大巴，故障和排除故障源自格瑞斯·霍普(Grace Hopper，1906—1992)发现的“飞蛾子”(Bug)和“抓蛾子”或“抓虫子”(Debug)。其他如鼠标、菜单……不胜枚举。至于哲学家进餐问题，理发师睡觉问题更是操作系统文化中脍炙人口的经典。

以计算机为核心的信息技术，从一开始就与应用紧密结合。例如，ENIAC 用于弹道曲线的计算，ARPANET 用于资源共享以及核战争时的可靠通信。即使是非常抽象的图灵机模型，也受到二战时图灵博士破译纳粹密码工作的影响。

在信息技术中，既有许多成功的案例，也有不少失败的案例；既有先成功而后失败的案例，也有先失败而后成功的案例。好好研究它们的成功经验和失败教训，对于编写案例型教材有重要的意义。

我国正在实现中华民族的伟大复兴，教育是民族振兴的基石。改革开放 30 年来，我国高等教育在数量上、规模上已有相当的发展。当前的重要任务是提高培养人才的质量，必须从学科知识的灌输转变为素质与能力的培养。应当指出，大学课堂在高新技术的武装下，利用 PPT 进行的“高速灌输”、“翻页宣科”有愈演愈烈的趋势，我们不能容忍用“技术”绑架教学，而是让教学工作乘信息技术的东风自由地飞翔。

本系列教材的编写，以学生就业所需的专业知识和操作技能为着眼点，在适度的基础知识与理论体系覆盖下，突出应用型、技能型教学的实用性和可操作性，强化案例教学。本套教材将会有机融入大量最新的示例、实例以及操作性较强的案例，力求提高教材的趣味性和实用性，打破传统教材自身知识框架的封闭性，强化实际操作的训练，使本系列教材做到“教师易教，学生乐学，技能实用”。有了广阔的应用背景，再造计算机案例型教材就有了基础。

我相信北京大学出版社在全国各地高校教师的积极支持下，精心设计，严格把关，一定能够建设出一批符合计算机应用型人才培养模式的、以案例型为创新点和兴奋点的精品教材，并且通过一体化设计、实现多种媒体有机结合的立体化教材，为各门计算机课程配齐电子教案、学习指导、习题解答、课程设计等辅导资料。让我们用锲而不舍的毅力，勤奋好学的钻研，向着共同的目标努力吧！

刘瑞挺教授 本系列教材编写指导委员会主任、全国高等院校计算机基础教育研究会副会长、中国计算机学会普及工作委员会顾问、教育部考试中心全国计算机应用技术证书考试委员会副主任、全国计算机等级考试顾问。曾任教育部理科计算机科学教学指导委员会委员、中国计算机学会教育培训委员会副主任。PC Magazine《个人电脑》总编辑、CHIP《新电脑》总顾问、清华大学《计算机教育》总策划。

前　言

大学应当为社会培养更多的应用型人才已成为共识。选择合适的教学内容，采用适合培养对象的教学方式在培养人才的过程中起着非常重要的作用。

数值计算方法是研究并解决数学问题的数值近似解的方法，是在计算机上使用的解决数学问题的方法，简称计算方法。然而，传统的“计算方法”已不适应培养应用型人才的现状，目前对“计算方法”的改革较多体现为对算法的实现，有选择数学软件(如 Matlab 等)实现算法的，也有给出算法流程图和程序实现的，试图改变传统“计算方法”只重方法不重实现的问题，当然这种改变无疑是正确的。

笔者认为，从培养应用型人才的角度考虑，可以从以下 3 个方面对传统的“计算方法”加以改进。

(1) 增加数学建模的内容，使读者知道相关的知识和方法可用于分析和解决哪一类实际问题。

(2) 提供求解问题的实例(主要是算法的实现)，给出算法流程图和实现程序可能更有助于提高读者的计算机应用能力和应用水平。

(3) 根据读者对象，适当调整内容。

基于上述观点，本书共分 8 章，内容包括：数值计算基本概念，插值与数据拟合方法，导数应用及近似计算，定积分应用及近似计算，方程求根数值方法，线性方程组数值解法，线性规划问题及解法，矩阵特征值与特征向量。本书从历史背景、知识回顾、实际应用、求解方法和算法实现(用 C 语言)5 个方面分别介绍了各章的相关内容。附录介绍了如何在 Visual Studio 6.0 环境下编辑和运行 C 程序。

笔者力求本书能具有以下特色。

(1) 注重知识和方法的应用。

(2) 可读性强。

(3) 富有启发性。

(4) 易于教师引导和教学。

(5) 易于学生学习掌握。

(6) 可服务于多层次、多专业、多学科的需要。

希望通过本书的学习，使读者分析解决实际问题的能力、程序设计能力和计算机应用水平均能够有较大提升。

由于水平有限，书中难免有疏漏和不妥之处，衷心希望广大读者提出宝贵意见！

徐亚平

2011 年 1 月于常州

目　录

第 1 章　数值计算基本概念 …… 1

1.1　用计算机解决实际问题的过程 …… 1

1.2　误差及其表示 …… 2

1.3　算法及算法分析 …… 4

1.3.1　算法描述 …… 4

1.3.2　算法流程图与算法的结构 …… 4

1.3.3　算法性能分析与度量 …… 6

1.3.4　算法的稳定性 …… 7

习题 1 …… 8

实验 1 …… 10

第 2 章　插值与数据拟合方法 …… 11

2.1　问题举例 …… 11

2.2　插值问题与插值法 …… 13

2.2.1　拉格朗日插值 …… 13

2.2.2　牛顿插值 …… 15

2.2.3　Hermite 插值 …… 17

2.2.4　分段线性插值 …… 19

2.2.5　三次样条插值 …… 20

2.3　数据拟合问题与最小二乘法 …… 24

2.3.1　数据拟合问题 …… 24

2.3.2　最小二乘法 …… 25

2.4　插值与数据拟合方法例程 …… 30

2.4.1　拉格朗日插值例程 …… 30

2.4.2　牛顿插值例程 …… 32

2.4.3　最小二乘曲线拟合例程 …… 33

习题 2 …… 35

实验 2 …… 36

第 3 章　导数应用及近似计算 …… 37

3.1　导数的基本知识 …… 37

3.2　导数在经济领域中的应用 …… 39

3.2.1　经济领域中常用的函数 …… 39

3.2.2　导数在经济分析中的应用举例 …… 39

3.3　导数的近似计算 …… 42

3.4　求导公式例程 …… 44

习题 3 …… 47

实验 3 …… 48

第 4 章　定积分应用及近似计算 …… 49

4.1　定积分的基本知识 …… 49

4.2　定积分应用 …… 51

4.2.1　面积和体积 …… 51

4.2.2　定积分在经济中的应用 …… 52

4.3　定积分的近似计算 …… 53

4.4　复化型求积公式 …… 57

4.4.1　复化梯形公式 …… 57

4.4.2　复化抛物线公式 …… 58

4.5　数值积分例程 …… 63

4.5.1　变步长复化梯形公式例程 …… 63

4.5.2　变步长复化抛物线公式例程 …… 64

习题 4 …… 65

实验 4 …… 67

第 5 章　方程求根数值方法 …… 68

5.1　非线性方程求根问题 …… 68

5.2　二分法 …… 71

5.3　切线法 …… 73

5.4　迭代法 …… 75

5.5　方程求根方法例程 …… 79

5.5.1　二分法例程 …… 79

5.5.2　切线法例程 …… 80

习题 5 …… 81

实验 5 …… 82

第 6 章　线性方程组数值解法 …… 83

6.1　线性方程组的基本知识 …… 83

6.2　线性方程组应用举例 …… 85

6.3　线性方程组的直接解法 …… 88

6.3.1　消元法 …… 88

6.3.2　三角分解法 …… 91

6.4　向量与矩阵的范数 …… 92

6.5　直接解法的误差分析 …… 93

6.6　线性方程组的迭代法解法 …… 94

6.6.1　简单迭代法与 Seidel 迭代法 …… 95

6.6.2　迭代法的收敛性 …… 98

6.7 线性方程组解法例程 …… 99

6.7.1 列主元素消元法例程 …… 99

6.7.2 三角分解法例程 …… 101

6.7.3 Jacobi 迭代法例程 …… 103

6.7.4 Seidel 迭代法例程 …… 105

习题 6 …… 106

实验 6 …… 108

第 7 章 线性规划问题及解法 …… 109

7.1 线性规划问题 …… 109

7.2 线性规划的图解法 …… 112

7.3 线性规划的单纯形法 …… 114

7.3.1 线性规划的标准形式 …… 114

7.3.2 单纯形法的基本步骤 …… 116

7.3.3 人工变量法 …… 120

7.3.4 单纯形法的实现算法 …… 121

7.4 单纯形法例程 …… 123

习题 7 …… 127

实验 7 …… 129

第 8 章 矩阵特征值与特征向量 …… 130

8.1 特征值与特征向量的基本知识 …… 130

8.2 特征值与特征向量应用举例 …… 133

8.3 乘幂法 …… 138

8.4 逆幂法 …… 142

8.5 实对称矩阵特征值的计算 …… 146

8.5.1 化实对称矩阵为三对角矩阵 …… 146

8.5.2 求实对称三对角矩阵特征值的对分法 …… 149

8.6 QR 方法 …… 151

8.7 矩阵特征值及特征向量计算例程 …… 156

8.7.1 乘幂法例程 …… 156

8.7.2 化实对称矩阵为三对角矩阵例程 …… 159

8.7.3 对分法计算实对称三对角矩阵特征值例程 …… 161

习题 8 …… 163

实验 8 …… 164

附录 Visual Studio 6.0 环境下建立和运行程序简介 …… 165

参考文献 …… 169

第1章 数值计算基本概念

计算问题是现代社会各个领域普遍存在的问题，工业、农业、交通运输、医疗卫生、文化教育、航空航天、科学技术研究等，哪一行哪一业都有许多数据需要计算，例如，发射一颗探测宇宙奥秘的卫星，科学家和工程技术人员就要对卫星的总体、部件以及选用的火箭进行全面的设计，这里面就有许许多多的数据要进行准确的计算。发射和回收的时候，又有关于发射角度、轨道、遥控、回收下落角度等需要进行精确的计算。又如，在高能加速器里进行高能物理实验，研究具有很高能量的基本粒子的性质、它们之间的相互作用和转化规律，这里面也有大量的数据计算问题。

计算机具有能存储数据、可完成四则运算和进行逻辑判断等基本功能。计算机具有存储容量大、运算速度快和计算精度高等特点。为了使计算机的使用方式更加简单方便，也为了提高计算机的使用效率，人们发明了操作系统。各种计算机语言的出现则为拓展计算机的"能力"奠定了良好的基础。然而，借助计算机解决问题的前提则是事先已设计出符合计算机特点和便于在计算机上实现的用于解决问题的方法。

计算数学是一门研究计算问题解决方法和相关理论的科学。计算数学属于应用数学的范畴，主要研究用计算机解决数学和逻辑问题的理论和方法。

计算方法也称数值分析是用计算机解决诸如积分、微分、非线性方程(组)、线性方程组、微分方程等数学问题的基本理论和方法。

实用计算方法是在介绍计算方法基本内容的同时，尽可能多地引入建立数学模型的实例和对方法实现的引导。

在设计用于解决具体问题的计算方法时，要依据计算机的特点即求解过程只能包括加、减、乘、除运算和逻辑运算。所设计的算法要便于在计算机上实现并且计算结果能达到精度要求。

尽管计算机的存储容量大且运算速度快，但评价一个算法优劣的重要标志依然是该算法的时间效率和所需存储量的大小。

1.1 用计算机解决实际问题的过程

应用计算机解决实际问题的过程分以下5步完成。

第一步：建立问题的数学模型；

第二步：构造(或选择)求解数学模型的算法(或称计算方法)；

第三步：编写程序(即用计算机语言描述算法)；

第四步：编辑、调试、编译和运行程序，获得计算结果；

第五步：分析计算结果。

【例1.1】 某公司计划生产Ⅰ、Ⅱ两种家电产品。已知生产一件产品需占用设备A、B的时数及需要的调试时间、每天可用于生产这两种家电的设备的时数及调试时间和出售一件产品的获利情况(见表1-1)。问该公司每天生产两种家电各多少件时获利最大?

表 1-1　生产信息表

项　　目	占用设备 A/h	占用设备 B/h	需调试时间/h	利润/元
生产Ⅰ1 件	0	6	1	2
生产Ⅱ1 件	5	2	1	1
每天可用时数	15	24	5	

解：第一步：建立数学模型。

设制造Ⅰ、Ⅱ产品数量分别为 x_1、x_2，则利润 $z=2x_1+x_2$。

问题：在现有设备、调试时数的限制下，如何确定产量 x_1、x_2，可使利润最大？可用数学语言表述如下。

目标函数：$\max z=2x_1+x_2$

$$
\text{约束条件}\begin{cases}
5x_2\leqslant 15 & \text{设备 A 的限制}\\
6x_1+2x_2\leqslant 24 & \text{设备 B 的限制}\\
x_1+x_2\leqslant 5 & \text{调试时间限制}\\
x_1,\ x_2\geqslant 0 & \text{非负约束}
\end{cases}
$$

以上数学描述即问题的数学模型称为线性规划。

第二步：构造解线性规划的算法。

构造解线性规划的算法即构造可获得既满足所有“约束条件”又使“目标函数”的值取到最大的 x_1 和 x_2 的方法(具体方法在第 7 章中介绍)。

1.2　误差及其表示

解决实际问题的过程通常表现为对数据的采集和处理过程。对于采集数据而言，多数情况下数据是通过观测获得的。由于受观测环境和观测工具精度的影响，采集到的数据往往是精确数据的近似值，即采集到的数据与实际数据之间存在一定的差距。由于采集到的数据不一定精确，加之计算机不能将其可表示范围内的任意实数都表示出来，从而导致处理的结果与实际结果之间也会存在一定的差距(有时差距也可能很大)，即所得结果仅为实际结果的近似。为了描述和表示这种差距，需要引入相应的概念。

误差：设 x^* 是精确值 x 的一个近似值，记 $e=x^*-x$，则称 e 为 x^* 的误差，简称为误差。

绝对误差：称误差的绝对值即 $|e|$ 为绝对误差。

绝对误差限：若绝对误差 $|e|\leqslant\varepsilon$，则称 ε 为绝对误差限。

显然，误差、绝对误差和绝对误差限可用于表示一个近似数与精确数之间的差距。由于精确值 x 往往是未知的，所以 e 也无法计算，但可根据情况估计出 ε。

【例 1.2】　设一个物体的重量为 x，当称其重量时，重量界于量器的两个刻度 6 和 7 之间，若计其重量为 6.5 并作为 x 的近似值，则 $|x-6.5|\leqslant 1/2$。

虽然误差、绝对误差和绝对误差限可以表明一个近似数与精确数之间的差距，但还不足以表明一个近似数的精确程度。

【例 1.3】　设有精确数 $x=100$，$y=10$。而 $x^*=99$，$y^*=9$ 分别是 x 和 y 的近似值，

则 $|x-x^*|=|y-y^*|=1$，即 x^* 和 y^* 有相同的绝对误差。然而相对而言，x^* 的精确程度明显要比 y^* 的精确程度高。

相对误差：设 x^* 是精确值 x 的一个近似值，e 是 x^* 的误差，记 $e^*=|e/x|$，则称 e^* 为近似值 x^* 的相对误差，简称为相对误差。

相对误差限：若 $e^*\leqslant\varepsilon^*$，则称 ε^* 为相对误差限。

由相对误差的概念可知，例 1.3 中 x^* 和 y^* 的相对误差分别为 0.01 和 0.1。

类似于绝对误差，当精确值 x 未知时，无法计算出 e^*，但可根据情况估计出 ε^*。

应用计算机解决实际问题的过程中的第五步：分析计算结果，就是对计算结果的误差进行分析。一般情况下，在设计算法时就要对计算结果的误差进行分析，以保证计算能获得满足精度要求的结果。

反映一个近似数准确程度的另一个常用概念是**有效数字**。计算的结果是否可靠，前提是参加运算的数据的每一位数字是否可靠。可靠的数字越多，这个近似数就越精确。在计算机上不可能取无限多位数字。具有多位乃至无穷位的准确值 x，需要用前有限位来近似时，是按照四舍五入规则进行截取和进位的，按此规则可保证其绝对误差最小。

【例 1.4】 圆周率的真值 $x=\pi=3.141\,592\,6\cdots$，若取 3 位数字，让 $\pi\approx\tilde{x}=3.14$，则 3.14 是 π 的所有有 3 位数字的近似值中绝对误差最小的；而 $\pi\approx x=3.1416$ 是所有 5 位数中绝对误差最小的。它们均是按照四舍五入规则进行截取和舍入的。可以看到，由四舍五入规则得到的两个近似值的绝对误差均不超过末位的半个单位，即：

$$|\pi-3.14|\leqslant 0.005=\frac{1}{2}\times10^{-2}=\frac{1}{2}\times10^{1-3}$$

$$|\pi-3.1416|\leqslant 0.000\,05=\frac{1}{2}\times10^{-4}=\frac{1}{2}\times10^{1-5}$$

定义 1.1 设准确数 x 的近似值 $\tilde{x}$ 表示成十进制浮点制形式：

$$\tilde{x}=\pm0.a_1a_2\cdots a_n\times10^m=\pm(a_1\times10^{-1}+a_2\times10^{-2}+\cdots+a_n\times10^{-n})\times10^m$$

其中 m 为整数，$a_i(i=1,\ 2,\ \cdots,\ n)$ 为 0 到 9 中的一个数字，$a_1\neq0$。如果 $\tilde{x}$ 的绝对误差不超过末位的半个单位，即

$$|x-\tilde{x}|\leqslant\frac{1}{2}\times10^{m-n}$$

则称近似值 $\tilde{x}$ 具有 n 位有效数字。

由定义，如果近似值 $\tilde{x}$ 的误差界是某一位的半个单位，该位到 $\tilde{x}$ 的第一位非零数字共计有 n 位，那么此 $\tilde{x}$ 就有 n 位有效数字。显然，有效数字的位数多少与小数点的位置无关。近似数的有效数字不仅给出了近似值的大小，而且还指出了它的绝对误差界。一个近似数的有效数字越多，其绝对误差和相对误差都越小。在计算中保护好有效数字，使之尽量不损失或少损失，是一项很重要的基本原则。

【例 1.5】 用圆周率的疏率 $\frac{22}{7}$ 来近似 π，讨论它的有效数字和误差界。

解：$\tilde{x}=\frac{22}{7}=3.142\,85\cdots=0.314\,285\cdots\times10^1$，即 $m=1$。

$$|\pi-\tilde{x}|=0.001\,26\cdots=0.126\cdots\times10^{-2}\leqslant0.5\times10^{-2}=\frac{1}{2}\times10^{-2}=\frac{1}{2}\times10^{1-3}$$

故疏率有 3 位有效数字，即 3.14 是有效的，从小数点后第 3 位起已不可靠。

1.3 算法及算法分析

如何设计和表示各种数据处理的方法，如何比较不同方法的优劣等是本课程研究的重要内容。算法是对数据处理方法的详细表述，下面介绍算法的特性、算法的描述、算法的性能分析与度量等。

算法是求解问题的方法的步骤序列。一个算法应该具有以下几个特性。

(1) 有穷性：一个算法必须在有限步之内结束。

(2) 确定性：算法的每一步必须有确切的定义。

(3) 可行性：算法中的每一步都可以通过基本运算得以实现。

(4) 输入：一个算法具有零个或多个输入。

(5) 输出：一个算法具有一个或多个输出，输出同输入之间存在某种特定的关系。

用程序设计语言描述的算法就是“程序”。作为一个程序，通常要满足以下要求。

(1) 正确：程序的执行结果应当满足预先规定的功能和性能要求。

(2) 可读：一个程序应当思路清晰、层次分明、简单明了、易读易懂。

(3) 健壮：当输入不合法数据时，应能作适当处理，不至引起严重后果。

(4) 高效：有效使用存储空间和有较高的时间效率。

1.3.1 算法描述

可以使用各种不同的方法来描述算法。现介绍如下。

(1) 最简单的方法是使用自然语言。用自然语言描述算法的优点是便于人们对算法的阅读和理解。缺点是不够严谨。

(2) 可以使用算法流程图、N－S图等算法描述工具。其特点是描述过程简捷明了。

(3) 可以直接使用某种程序设计语言来描述算法，前提是对程序设计语言非常熟悉，其特点是接近于程序但不直观，需要借助于注释加以说明。

(4) 可以使用一种称为伪码语言的描述方法来描述算法。伪码语言介于高级程序设计语言和自然语言之间，它比程序设计语言更容易描述和被人理解，而比自然语言更接近程序设计语言。它虽然不能直接执行但很容易被转换成高级语言。

为使读者把注意力集中于对算法的理解而不受描述工具的影响，本书主要选择自然语言和算法流程图来描述算法，算法的实现采用C语言。

1.3.2 算法流程图与算法的结构

算法流程图是用图形表示的算法。用图形表示算法具有直观、易于理解等特点。常用的流程图符号如图1.1所示。

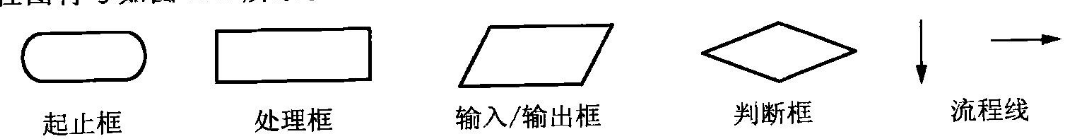


图 1.1 常用的流程图符号

【例 1.6】 图 1.2 是计算："1+1/2+1/3+1/4+…+1/100"的算法流程图。

【例 1.7】 图 1.3 是"在 N 个数中找出一个最大的"这一过程的算法流程图，其中 N_i 表示第 i 个数。

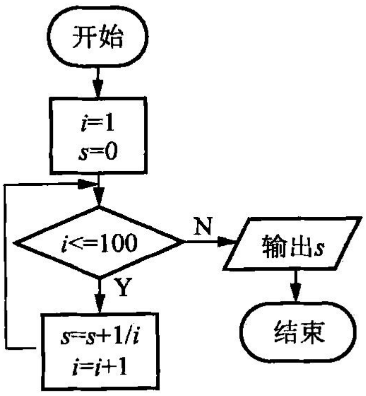


图 1.2 例 1.6 的算法流程图

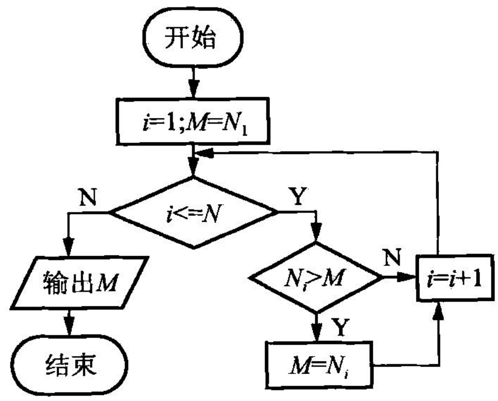


图 1.3 例 1.7 的算法流程图

【例 1.8】 图 1.4 是问题"输入 50 个人的学号和成绩，输出其中成绩在 80 分以上者的学号和成绩。"的处理算法的算法流程图，其中 N_i 表示第 i 个人的学号，G_i 表示第 i 个人的成绩。

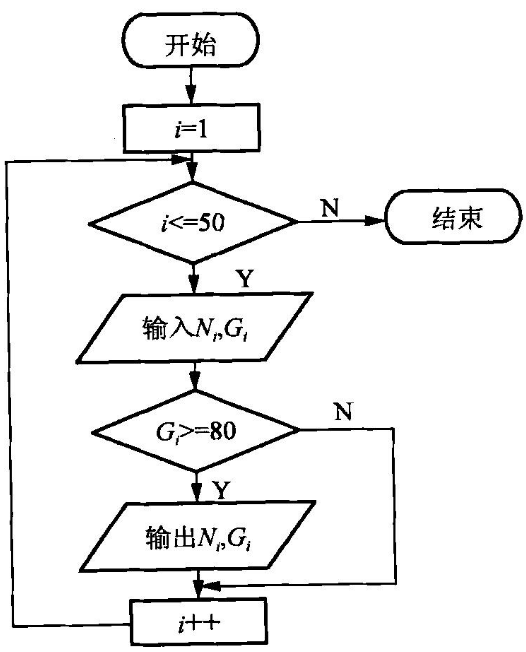


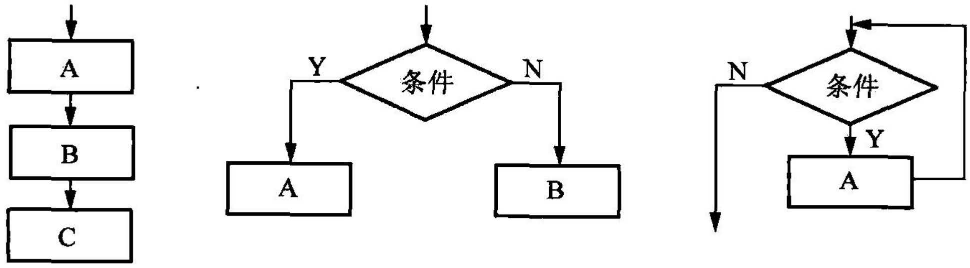
图 1.4 例 1.8 的算法流程图

显然，用算法流程图描述算法会使算法的逻辑结构更加清晰。反之，能正确地画出一个算法的算法流程图则说明对该算法的逻辑结构有了清晰的了解。

准确理解并能正确描述算法是程序设计者的基本素质。

算法不同其对应的算法流程图自然不同，然而任一算法都是若干基本结构的不同组合，就像一个城市里的楼房一样，虽然其高低、大小、样子可以互不相同，但(式样)结构均为几种基本几何图形的不同组合。

算法的基本结构有以下 3 种，如图 1.5 所示。



(a)顺序结构　　(b)选择结构　　(c)循环结构

图 1.5　算法的几种基本结构的流程图

(1) 顺序结构，即构成算法的各部分有严格的先后顺序(例如，把大象放进冰箱里要分 3 步：第一步把冰箱门打开；第二步把大象放进去；第三步把冰箱门关上，3 步的顺序不能改变)。

(2) 选择结构，即已知可能出现的结果为两种或两种以上，结果不同对应的下一步的操作也不同(例如，明天上午有课吗？可能的结果有两种，即有和没有)。

(3) 循环结构，即在一定的条件下重复执行某些操作(例如，计算 100 个数的和的过程就是 99 次加法运算的重复)。

1.3.3　算法性能分析与度量

一个算法在计算机上执行时所需要的时间取决于下列因素：①硬件的速度；②书写程序的语言；③编译程序所生成目标代码的质量；④问题的规模。

显然，在各种与计算机相关的软、硬件因素都确定的情况下，一个特定算法的运行工作量的大小就只依赖于问题的规模(通常用正整数 n 表示)，或者说它是问题规模的函数。因而可以从一个算法的**时间复杂度**与**空间复杂度**来评价算法的优劣。

一个算法是由控制结构和原操作构成的，其执行时间取决于两者的综合效果。为了便于比较同一问题的不同的算法，通常以该算法执行的原操作的次数作为算法的时间度量。一般情况下，算法中原操作重复执行的次数是规模 n 的某个函数 $T(n)$。而多数情况下精确地计算 $T(n)$是困难的，只能给出它的一个估计。

定义 1.2　设问题的规模为 n，算法执行的原操作的次数为 $g(n)$，如果存在 $f(n)$和正常数 c，使得当 n 趋向于无穷大时，$f(n)/g(n)$的极限为 c。则称 $T(n)=cf(n)$为算法的时间复杂度，记为：

$$T(n)=O(f(n))$$

读为：算法的时间复杂度与 $f(n)$是同阶的。

例如，完成两个 $n\times n$ 阶矩阵相乘需要进行的运算 $g(n)=n^3$(次乘法运算)$+n^2(n-1)$(次加减法运算)。取 $f(n)=n^3$，则当 n 趋向于无穷大时，$f(n)/g(n)$的极限为 1，所以 $T(n)=O(n^3)$，即时间复杂度与 n^3是同阶的。

通常用 $O(1)$ 表示常数计算时间。且有：

$$O(1)<O(\log_2 n)<O(n)<O(n\log_2 n)<O(n^2)<O(n^3)<O(2^n)$$

算法的空间复杂度与算法的时间复杂度类似，算法的空间复杂度是指在实现算法时所需的存储量。同算法的时间复杂度一样是评价一个算法好坏的重要标志。

1.3.4 算法的稳定性

由于获取的原始数据有可能是近似的，而且在每一步的计算过程中都可能产生舍入误差，所以，完全有可能出现这样一种情况，即在计算的过程中，前一步产生的误差不断被放大，以至于最终算出错误的结果。

【例 1.9】 计算积分：

$$E_n=\int_0^1 x^n e^{x-1}\mathrm{d}x,\ n=1,\ 2,\ \cdots,\ 9$$

解：利用分部积分法可得：

$$E_n=x^n e^{x-1}\Big|_0^1-n\int_0^1 x^{n-1}e^{x-1}\mathrm{d}x$$

从而有递推公式：

$$\begin{aligned}&E_n=1-nE_{n-1}\ n=2,\ 3,\ \cdots,\ 9\\&E_1=1/\mathrm{e}\end{aligned}\qquad(1-1)$$

取 $E_1^*=0.367\ 879$，利用式(1-1)计算，得计算结果见表 1-2。

表 1-2 取 $E_1^*=0.367\ 879$，用式(1-1)计算所得结果

n	E_n	n	E_n
1	0.367 879	6	0.127 120
2	0.264 242	7	0.110 160
3	0.207 274	8	0.118 720
4	0.170 904	9	−0.068 480
5	0.145 480		

由表 1-2 知，当 $n=9$ 时，计算结果 $E_9=-0.068\ 480$，这与积分性质：被积函数非负则积分非负相矛盾，即计算结果是错误的。

算法分析：

由于 e 是无理数，所以，$E_1=1/e$ 只能取 E_1 的近似值 E_1^*。设 $E_1^*-E_1=\varepsilon$，则有 $E_1^*=E_1+\varepsilon$。代入递推式(1-1)计算 E_2 有计算结果：

$$E_2^*=1-2E_1^*=1-2(E_1+\varepsilon)=1-2E_1-2\varepsilon=E_2-2\varepsilon$$

把 E_2^* 代入递推式(1-1)计算 E_3 有计算结果：

$$E_3^*=1-3E_2^*=1-3(E_2-2\varepsilon)=1-3E_2+6\varepsilon=E_3+6\varepsilon$$

同理有：

$$E_4^*=E_4-4!\ \varepsilon$$

当 $n=9$ 时，有 $E_9^*=E_9+9!\ \varepsilon$，即 $E_9^*-E_9=9!\ \varepsilon=362\ 880\varepsilon$。

上述分析说明：用递推式(1-1)计算 E_n，当算到 E_9 时，第一步产生的误差已被放大了 362 880 倍!

如果采用新的算法，把上述递推关系改写成

$$E_{n-1}=\frac{1-E_n}{n},\ n=\cdots,\ 3,\ 2\qquad(1-2)$$

从后向前计算，则 E_{n-1} 的误差是 E_n 的 $1/n$ 倍。所以，若取 n 足够大，误差逐步减小，其影响愈来愈小。为了得到出发值，可考虑关系：

$$E_n = \int_0^1 x^n e^{x-1} dx \leqslant \int_0^1 x^n dx$$
$$= 1/(n+1)$$

当取 $n=20$ 且 $E_{20}=0.0$ 时，按式(1-2)计算的结果见表1-3。

表1-3　取 $n=20$，$E_{20}=0.0$，按式(1-2)计算的结果

n	E_n	n	E_n
20	0.000 000 0	14	0.062 732 2
19	0.050 000 0	13	0.066 947 7
18	0.051 000 0	12	0.071 773 3
17	0.052 777 8	11	0.077 352 3
16	0.055 719 0	10	0.083 877 1
15	0.059 017 6	9	0.091 612 3

算法式(1-1)与式(1-2)的区别在于，式(1-1)在计算的过程中，初始产生的误差不断被放大，而式(1-2)则没有。

对于一个算法而言，如果像算法式(1-1)一样，在计算的过程中，初始产生的误差不断被放大，则称该算法是不稳定的，否则称该算法是稳定的(像算法式(1-2)就是稳定的)。

由例1.9知，一个不稳定的算法可以导致错误的结果。所以，在设计算法时，要分析算法的稳定性，只有稳定的算法才是可靠的。

习　题　1

1.1　举例说明用计算机解决实际问题的过程。

1.2　指出下列各数具有几位有效数字：

2.000 4　　−0.002 00　　9 000.00

1.3　一个算法步骤如下。

第一步：令 S 的值为0，i 的值为5；

第二步：如果 $i \leqslant 8$ 则执行第三步，否则执行第六步；

第三步：计算 $S+i$ 的值，并将结果代替 S 的值；

第四步：用 $i+2$ 的值代替i；

第五步：转去执行第二步；

第六步：输出 S。

执行以上算法，输出的结果是什么？

1.4　已知一个三角形的三边长分别是 a，b，c，它的面积可用海伦-秦九韶公式计算。

$$S=\sqrt{p(p-a)(p-b)(p-c)}，其中\ p=\frac{a+b+c}{2}。$$

设计一个算法，输入三角形的三条边长 a，b，c，输出三角形的面积 S。

1.5　设计一个输入一个自变量 x 的值，求出下列分段函数的函数值的算法。

$$f(x)=\begin{cases} x+2, & x \geqslant 0 \\ x^2, & x<0 \end{cases}$$