

清华大学计算机系列教材

程序设计基础

清
华
大
学
计
算
机
系
列
教
材

吴文虎 编著



清华大学出版社

清华大学计算机系列教材

程序设计基础

吴文虎 编著



B1282326

清华大学出版社
北京

内 容 简 介

本书重点讲授在 C/C++ 语言环境下,编写程序的思路和方法,涉及计算机语言、数据结构和常用算法等内容。全书内容丰富,强调动手实践,深入浅出地引导读者理性思维和理性实践,教学方法引人入胜,便于自学。

本书可作为大专院校教材,亦可供从事计算机、自动化和相关领域的科研人员参考自学。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

程序设计基础/吴文虎编著. —北京:清华大学出版社,2003

(清华大学计算机系列教材)

ISBN 7-302-07163-2

I. 程… II. 吴… III. C语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2003)第 076348 号

出 版 者: 清华大学出版社

<http://www.tup.com.cn>

社 总 机: 010-62770175

地 址: 北京清华大学学研大厦

邮 编: 100084

客 户 服 务: 010-62776969

策 划 编 辑: 马 瑛 琚

责 任 编 辑: 刘 彤

印 刷 者: 北京市清华园胶印厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印 张: 17.5 字 数: 399 千 字

版 次: 2003 年 9 月 第 1 版 2003 年 9 月 第 1 次 印 刷

书 号: ISBN 7-302-07163-2/TP·5220

印 数: 1~10000

定 价: 28.00 元

作者简介



吴文虎,1955—1961年分别就读于清华大学电机工程系及自动控制系;现为计算机系教授、博士生导师,主要研究方向包括语音识别及语言理解、语音合成、语音信号数字处理等,主讲“语音信号数字处理”和“程序设计基础”课。1990—1997年,担任人机语音通信实验室的负责人。作为项目负责人或主要参加者,承担了多项国家级项目,科研成果多次获国家教委、电子部的科技进步二等奖,曾获863高科技成果精品展金奖。1991—1999年间担任中国计算机学会普及委员会主任。作为总教练和领队,曾连续14年带领中国队参加国际信息学奥林匹克竞赛,中国队累计获金牌29块,银牌16块,铜牌12块。并于1997年获“清华大学优秀教学成果特等奖”,1998年获“全国优秀教师一等奖”,1999年荣获“首都劳动奖章”,2000年获“北京市教学优秀成果一等奖”,2001年获“全国师德先进个人”称号,2002年获信息学奥林匹克国际委员会颁发的“特殊贡献奖”,2003年获中国计算机学会颁发的“特殊贡献奖”。主编、主审和编写了几十本有关计算机的图书。主要兼职有:全国高校计算机基础教育研究会副理事长,教育部现代远程职业教育与成人教育专家组组长,中国计算机学会理事。

序

清华大学计算机系列教材已经出版发行了近 30 种,包括计算机专业的基础数学、专业技术基础和专业等课程的教材,覆盖了计算机专业大学本科和研究生的主要教学内容。这是一批至今发行数量很大并赢得广大读者赞誉的书籍,是近年来出版的大学计算机教材中影响比较大的一批精品。

该系列教材的作者都是我熟悉的教授与同事,他们长期在第一线担任相关课程的教学工作,是一批很受大学生和研究生欢迎的任课教师。编写高质量的大学(研究生)计算机教材,不仅需要作者具备丰富的教学经验和科研实践,还需要对相关领域科技发展前沿的正确把握和了解。正因为该系列教材的作者们具备了这些条件,才有了这批高质量优秀教材的出版。可以说,教材是他们长期辛勤工作的结晶。系列教材出版发行以来,无论从其发行的数量、读者的反映、已经获得的许多国家级与省部级的奖励以及在各个高等院校教学中所发挥的作用上,都可以看出该系列教材所产生的社会影响与效益。

计算机科技发展异常迅速、内容更新很快。作为教材,一方面要反映本领域基础性、普遍性的知识,保持内容的相对稳定性;另一方面,又需要跟踪科技的发展,及时地调整和更新内容。该系列教材都能按照自身的需要及时地做到这一点,如《计算机组成与结构》一书十年中共发行了三版,其他如《数据结构》等也都已发行了第二版,使教材既保持了稳定性,又达到了先进性的要求。该系列教材内容丰富、体系结构严谨、概念清晰、易学易懂,符合学生的认识规律,适合于教学与自学,深受广大读者的欢迎。系列教材中多数配有丰富的习题集和实验,有的还配备多媒体电子教案,便于学生理论联系实际地学习相关课程。

随着我国进一步的开放,我们需要扩大国际交流,加强学习国外的先进经验。在大学教材建设上,我们也应该注意学习和引进国外的先进教材。但是,计算机系列教材的出版发行实践以及它所取得的效果告诉我们,在当前形势下,编写符合国情的具有自主版权的高质量教材仍具有重大意义和价值。它与前者不仅不矛盾,而且是相辅相成的。

我希望今后有更多、更好的我国优秀教材的出版。

清华大学计算机系教授,中科院院士

张钹

2002 年 6 月 25 日

前 言

“计算机语言与程序设计”是一门十分重要的基础课程。该课长期沿袭着这样的教学模式：过于注重语句、语法和一些细节，基本上是以高级语言自身的体系为脉络展开的，没有把逻辑与编程解题思路放在主体地位上；对如何分析问题和解决问题讲得不够，对学生编程的能力、上机解题的能力训练不够。这样就给后续课程及研究生阶段的课题研究留下了缺憾。很多学生在学习这门课时感到枯燥难学，学过之后，不能用来解决实际问题。

我个人的经历有些不同，除了学校给我安排的教学和科研任务之外，20年来我一直指导初中学生、高中学生和大学生参加有关计算机的各种比赛，包括国际信息学奥林匹克和ACM世界大学生程序设计竞赛，通过对这些学生成长道路的反复思考和研究，使我感到很有必要改变我们的课程教学模式，用新的教学理念和方法培养一流人才。对这一问题，我和有关领导谈了自己的想法，他们非常支持。

从2001年9月起，我接受了程序设计基础课程的教学任务，并开始对程序设计基础课程教学模式进行改革：以强调动手实践上机编程为切入点；以任务驱动方式，通过实例讲授程序设计的基本概念和基本方法；重点放在思路上，即在C/C++语言的环境下，针对问题进行分析，构建数学模型，理出算法并编程实现。同时，要求学生养成良好的编程习惯；在教学过程中培养学生的思维能力和动手能力，鼓励学生探索、研究和创新。在指导思想上，强调转变观念，以学生为中心，将学生视为教学的主体，安排教学首先考虑培养目标、学生的认知规律和学习特点。在教学的每一个环节，顾及学生的实际情况，多想怎样才能有利于调动学生学习的积极性，引导学生主动学习。具体的改革措施主要针对两个方面：教学模式和对学生学习的评价方式。

对教学模式的改革

提出强化实践。明确告诉学生：程序设计课是高强度的脑力劳动，不是听会的，也不是看会的，而是自己练会的。只有让学生动手，他才会有成就感，进而对课程产生兴趣，学起来才比较从容。因此，我们的基本思想是在理论指导下，让学生动手、动脑，更多地上机实践。学生只有在编写大量程序之后，才能获得真知灼见，感到运用自如。注重学生动手能力的培养是这门课和以往课程最大的不同之处。

提出理性思维和理性实践。按照建构主义的学习理论，学生作为学习的主体在与客观环境（指所学内容）的交互过程中构建自己的知识结构。教师应引导学生在解题编程的实践中探索其中带规律性的认识，将感性认识升华到理性高度，只有这样，学生才能举一反三。

提出授课的原则是要学生“抱西瓜”而不是“捡芝麻”，重点放在思路、算法、编程构思和程序实现上。语句只是表达工具，讲一些最主要的，对细枝末节的东西根本不讲。要求学生在课堂上积极思考，尽量当堂学懂。突出上机训练，在编写程序的过程中，使学生提高利用计算机这个智力工具来分析问题和解决问题的能力。

提出要让学生养成良好的编程习惯。我们在与国内一些软件公司的技术人员座谈时了解到，中国软件之所以上不去的原因之一就有“习惯问题”。印度十个人编程，会编出一样的

东西,而我们十个人编程可能会有十种风格。因为我们忽略了一个重要问题,即“顾客”的感受,程序的编写是给别人看的,而不是只给我们自己看的。再者,尽管我们学生模型构思做得很快,但编程的基本功不扎实,往往到了关键的时候,就出问题。鉴于此,在课上我们强调程序的可读性、规范性;要求变量必须加注释;程序构思要有说明;学会如何调试程序;尽量使程序优化;还要求对程序的运行结果做正确与否的判断和分析。

提出“自学、动手、应用、上网”的学习习惯。我认为在本科阶段就应该注意培养学生的自学能力。很多东西完全是可以自学的,尤其是计算机。计算机是实践性极强的学科,所学的内容和要实践的东西是一个整体,因此可以自己动手来学,书上看不懂的在机器上动手试试,往往就弄懂了。上网是指充分利用网络平台,提高获取信息、处理信息和交流信息的能力。

对学生学习评价方式的改革

考试是检验学生学习效果、评价学生学习业绩的重要环节。考试作为“指挥棒”对教学目标、教学过程有着相当大的影响。我一直在思考如何进行考试改革,如何借助考试环节调动和激发学生自主学习的积极性、创造性等问题。

开学之初,我就向学生宣布考试方式——上机解题,判分也是由计算机来完成,对就是对,错就是错,不纸上谈兵,不考笔试,不考死记硬背的东西。我们平时比较注意对学生学习方式的引导,让学生明白:理论很重要,要在理论指导下,动手动脑、有条有理地进行实践。实践才能出真知,动手才能学到真本事。

我们还将一些有较好程序设计基础的学生组织起来,因材施教,引导他们进行探索式的研究性学习,让他们继续提高。同时让他们在班上担任“小教员”,帮助同学学习。

我们这样做行不行呢?经过两年的教学实践,这门课取得了很好的教学效果,学生给以很高的评价。学生点评为:“授课方式独特新颖,深入浅出,启发式教学,激发学生兴趣,调动学生的积极性,有助于学生独立思考能力的提高。”(引自清华大学2001年下半年教学评估结果查询)参加“小教员”工作的学生,提高了责任感,培养了敬业精神。他们的水平和能力也有相应的提高,其中三名同学代表清华大学参加了2002年ACM世界大学生程序设计竞赛的分区赛和总决赛,取得了世界总排名第四的好成绩(2300支队伍参加区域赛,60支队伍参加总决赛)。

2002年5月,在北京市高校计算机基础教育研讨会上,我曾应约就此课程的教学改革作了专题报告,受到了与会专家和老师们的好评,他们认为“这是非常好的新的教学范例”。

改革是没有止境的。经过两年的实践,我感到在一些方面还要进一步努力,还有许多工作要做:要进一步加大学生训练环节的力度;要加强对基础较差学生的辅导;要建立一个因材施教的机制,创造条件,让学生能有更广阔的发展空间;要建立平时的督促机制,让每一个学生真正落实动手实践;要考虑与后续课程的衔接。

现在大家看到的这本教材就是在上述的背景下,整理了课上的教案,补充了一些内容写出来的。在教材成文的过程中,我的同事和学生(博士生)起了很大作用。他们提出了很好的建议,对一些算法进行了研究和整理,特别是对全书整体上的结构进行了缜密的推敲。

从一种体系转变为现在的体系是有相当大的难度的,也有风险,学生爱不爱这样学,能不能学到真本事,是不是能够达到预期的教学目标,都会存在问题。但我以为,要改革就要知难而进,不付诸努力就收到良好的教学效果是不可能的。

目前这本教材可能存在很多不足,但是我们有这种思想准备,在教学实践中,多听取学生的反馈意见,不断修改,使之日臻完善。我们相信,恒心与虚心能够成就一本好的教材。

参加本书研究、撰写工作的还有徐明星(参加本书总体策划与章节编排)、邬晓钧(撰写第9、10、13章及附录)和李净(进行教案整理、图文设计),此外赵强工程师和杨非同学也做了大量的书稿整理和成文工作,吴根清、孙辉、刘建、刘林泉、邓菁、陈德锋、侯启明等同学看了本书的第一稿,提出了宝贵的修改意见。在此一并感谢他们所付出的劳动。

由于时间仓促,作者水平有限,书中难免有纰漏,欢迎读者多提宝贵意见。

吴文虎

2003年7月

MISSY/02

目 录

第 1 章 绪论	1
第 2 章 编程准备	4
2.1 程序编写	4
2.2 程序代码及说明	7
2.3 输出流对象 cout	8
2.4 输入流对象 cin	9
2.5 程序注释	9
2.6 算术运算符	10
2.7 数学函数	10
2.8 小结	11
习题	12
第 3 章 变量、代数与计算机解题	13
3.1 程序的基本结构	13
3.2 变量与数据类型	15
3.2.1 变量的基本概念	15
3.2.2 数据类型	15
3.3 定义变量和赋初值	16
3.4 变量赋值	16
3.4.1 赋值符号与赋值表达式	16
3.4.2 变量赋值的 5 要素	16
3.5 小结	17
习题	17
第 4 章 逻辑思维与计算机解题	18
4.1 关系运算和关系表达式	19
4.1.1 关系运算符	19
4.1.2 关系表达式的一般格式	19
4.1.3 将“是”“否”写成关系表达式	20
4.2 枚举法的思路	21
4.3 循环结构	22
4.3.1 使用循环结构的部分程序	22
4.3.2 for 语句的格式和执行过程	22

4.3.3	使用 for 循环解题实例	23
4.3.4	for 循环的程序框图	25
4.4	分支结构	26
4.4.1	if 语句的格式	26
4.4.2	分支结构的实例	27
4.5	任务 4.1 的程序框图	28
4.6	任务 4.1 的参考程序	29
4.7	逻辑问题及其解法	31
4.7.1	逻辑运算符与逻辑表达式	31
4.7.2	逻辑问题的解题思路	32
4.7.3	任务 4.2 的参考程序	37
4.8	小结	40
	习题	41
第 5 章	数据组织、筛选与排序问题的解题思路	42
5.1	数组	42
5.1.1	一维数组的定义	44
5.1.2	数组初始化	44
5.2	筛法	45
5.3	冒泡排序法	48
5.4	结构与结构数组	51
5.4.1	结构体类型的定义	51
5.4.2	结构体变量的定义和引用	51
5.4.3	结构体变量的初始化	53
5.4.4	结构数组	54
5.5	二维数组	56
5.5.1	二维数组的定义	57
5.5.2	二维数组的初始化	57
5.5.3	二维数组中的元素存放顺序	57
5.6	小结	60
	习题	60
第 6 章	函数、递推、递归	62
6.1	函数	62
6.1.1	函数的说明	63
6.1.2	函数的定义	63
6.1.3	函数的返回值	64
6.1.4	函数的调用	64

6.1.5	形式参数和实在参数	65
6.1.6	调用和返回	65
6.1.7	带自定义函数的程序设计	66
6.1.8	编程实例	68
6.2	递推	69
6.2.1	递推数列的定义	71
6.2.2	递推算法的程序实现	72
6.3	递归及其实现	73
6.4	递归算法举例	80
6.4.1	计算组合数 $C(m, n)$	80
6.4.2	青蛙过河	81
6.4.3	快速排序	85
6.4.4	分书问题	90
6.4.5	下楼问题	93
6.4.6	八皇后问题	96
6.4.7	跳马问题	99
6.4.8	数字旋转方阵	101
6.5	小结	105
	习题	105
第7章	指针	111
7.1	指针的概念	111
7.1.1	指针定义与初始化	112
7.1.2	指针赋值	112
7.1.3	在赋值语句中使用间接访问运算符	114
7.2	指针与数组	115
7.2.1	用数组名给指针赋值	115
7.2.2	数组名是一个常量指针	116
7.2.3	编程实例	117
7.2.4	指针数组	119
7.3	字符串及其处理	122
7.3.1	字符数组的定义和初始化	122
7.3.2	字符数组元素的赋值	124
7.3.3	字符判断函数	125
7.3.4	字符串相关函数	125
7.3.5	字符串应用举例	126
7.4	指针与函数	128

7.4.1	函数返回指针	128
7.4.2	指向函数的指针	130
7.5	指针和结构	133
7.6	指针与结构数组	135
7.7	引用的概念及应用	138
7.8	利用引用来传递参数	139
7.9	几种参数传递方式的比较	140
7.10	小结	144
	习题	145
第 8 章	蒙特卡罗法	150
8.1	伪随机数的产生	150
8.1.1	产生随机整数	150
8.1.2	产生随机小数	151
8.2	伪随机数的应用	152
8.2.1	求 π 的近似值	152
8.2.2	计算图形面积	153
8.3	小结	155
	习题	155
第 9 章	贪心法	156
9.1	贪心法解题的一般步骤	156
9.1.1	事件序列问题	156
9.1.2	区间覆盖问题	159
9.1.3	贪心法解题的一般步骤	162
9.2	贪心法相关理论	163
9.2.1	多阶段决策问题、无后向性与最优化原理	163
9.2.2	有向图最短路径的 Dijkstra 算法	163
9.2.3	贪心法解题的注意事项	167
9.3	小结	169
	习题	169
第 10 章	动态规划	170
10.1	石子合并问题	170
10.1.1	问题描述	170
10.1.2	分析与解题	171
10.1.3	动态规划思想	177
10.2	用动态规划方法解题	177

10.2.1	适用条件和相关概念	177
10.2.2	背包问题	178
10.2.3	生产计划问题	180
10.2.4	最长公共子序列	183
10.2.5	最佳旅行路线	186
10.3	小结	191
	习题	191
第 11 章	链表	193
11.1	举例说明链表的概念	193
11.2	建立链表的过程	194
11.3	链表结点的插入与删除	198
11.3.1	链表结点的插入	198
11.3.2	链表结点的删除	204
11.4	循环链表	207
11.5	小结	210
	习题	211
第 12 章	二叉树	213
12.1	基本概念	213
12.2	二叉树的遍历	214
12.3	二叉树的建立	215
12.4	小结	218
	习题	218
第 13 章	流与文件	219
13.1	流	219
13.1.1	流的概念与输入输出格式	219
13.1.2	改变整数的进制	220
13.1.3	设置浮点数的精度	221
13.1.4	设置输入输出宽度	222
13.1.5	设置对齐方式和填充字符	223
13.1.6	其他设置	224
13.2	文件	225
13.2.1	将数据保存到文件	225
13.2.2	从文件中读取数据	227
13.3	小结	231
	习题	231

附录A 程序调试	233
A.1 计分程序的调试	233
A.1.1 编译时的调试	235
A.1.2 运行时的调试	236
A.1.3 其他调试相关知识	242
A.2 跳马程序的调试	243
附录B 库函数	250
B.1 数学函数	250
B.2 字符判断函数	252
B.3 字符串相关函数	255
参考文献	263

第1章 绪 论

计算机语言及程序设计是信息学的一门基础课程,旨在为学生进行程序设计打基础。有的学者认为:

程序设计=计算机编程语言+数据结构+算法

程序设计基础的核心内容包括:程序设计基本结构、算法与问题求解、基本数据结构、递归和事件驱动程序设计等。

在程序设计基础的学习过程中,须深刻理解计算机是“人类通用智力工具”。要用好这个工具,人是主动的,是第一位的,必须发挥人的能动性。要让计算机帮助运算,就要知道计算机是如何工作的,它能够做什么,擅长做什么。人要用计算机能懂的语言驾驭计算机,让它成为驯服的工具。学习程序设计的目标是利用计算机这个智力工具来分析问题和解决问题。因此,编程能力的培养是这门课的首要任务。按照可持续发展的教育观,程序设计基础课应处理好知识、能力和素质三者的辩证关系。一个具有较强能力和良好素质的人必须掌握丰富的知识。程序设计基础领域的知识是由程序设计的基本概念和程序设计艺术(技巧)组成的,要掌握这些基本概念和设计艺术,必须立足于理性化的学习和实践。能力是技能化的知识,是知识的综合体现。程序设计能力的培养不是纸上谈兵就能做到的,需要强调动手实践,不动手是绝对学不会的。但是这门课也有别于那种简单的、以人操作为主的实验课。程序设计需要以扎实的理论基础、科学方法,以及思维能力和思维方法来指导实践。简而言之,就是学习程序设计既要动手又要动脑,即所谓进行“理性”的思维和“理性”的实践。

素质是知识和能力的升华。高的素质可以使知识和能力更好地发挥作用,还可以推动知识不断扩展,能力进一步增强。素质教育强调尊重学生的主体作用和主动精神,注重开发学生的潜能,以形成健全人格为根本特征。从程序设计基础课而言,要在内容中融入科学的世界观和方法论,在课程的学习中学会理性思维和理性实践;同时还需要养成良好的编程习惯,以及与人协同的精诚合作精神。

知识、能力和素质是创新的基础。超凡的想象力、扎实的基本功和丰富的实践经验是创新的源泉。对学生创新能力的培养可以从点滴做起;就程序设计课而言,应该逐步向“研究性学习”发展,引导学生就一些新的算法和问题解法进行探索,给他们创造“攀登顶峰”的体验机会,鼓励他们大胆提出新思路和新方法。

以上是本书所反映的教学思想的表述,下面是对教学的设想。

1. 教学对象

理工科大学生,程序设计初学者。

2. 教学目标

- (1) 程序设计的重要性;
- (2) 程序设计的基本概念与基本方法;
- (3) 编程解题的思路与典型方法;
- (4) 算法及算法步骤;

(5) 程序结构与相应语句;

(6) 编码与上机调试。

3. 教学重点

(1) 掌握程序设计的基本概念、基本方法;

(2) 在 C/C++ 语言的环境下,学会如何针对问题进行分析、构建数学模型,寻找算法并编程实现;

(3) 有条有理有根有据地编程实践;

(4) 养成良好的编程风格与习惯;

(5) 重在思维方法的学习,鼓励创新。

4. 指导思想

(1) 立足改革

人的认识要随着时代的前进而不断深化。在新的形势和环境下,教学要突破传统观念和传统模式,要追求高效和完美,以培养高素质有创造精神的人才作为教学目标。

(2) 以学生为中心

- 学生是教学的主体,安排教学首先必须考虑培养目标、学生的认知规律和学习特点。

- 教学的每一个环节都要顾及学生的实际情况,要有利于调动学生学习的积极性,引导学生主动学习。

(3) 强化实践

程序设计是高强度的脑力劳动,实践性极强,不是听会的,也不是看会的,而是练会的。要让学生充分上机动手编程。这可能是与以往的教学安排最大的不同之处。

(4) 鼓励和引导探索式的学习

按照建构主义的学习理论,学生(作为学习的主体)是在与客观环境(所学内容)的交互过程中构建自己的知识结构的。要引导学生在解题编程的实践中探索其中带规律性的认识,并将感性认识升华到理性的高度。

(5) 突出重点

重点放在思路、算法、编程构思和程序实现上,语句只是表达工具,强调主次分明,“抱西瓜,不捡芝麻”。重在训练利用计算机编程手段分析问题和解决问题的能力。

(6) 养成良好的编程习惯

- 程序构思要有说明。
- 强调可读性。
- 变量要加注释。
- 学会如何调试程序。
- 对运行结果要做正确与否的分析。

(7) 考试方法

原则是考学生真正的编程能力。

- 不纸上谈兵,不搞笔试,不考死记硬背的东西。
- 上机解题,自动测试。
- 考核试题与平时训练难度相当,注重平时训练,有相应练习平台支持。

(8) 学习方法

- 动手动脑,在理论指导下实践。
- 注重编程思路的学习和总结。
- 提倡做学问,要又学又问。
- 进行大量练习,以求熟能生巧,运用自如。

(9) 学习心态

- 提倡“自立、自信、自尊、自强”。
- 知难而进。
- 充满信心。

5. 教学内容安排

- (1) 绪论:程序设计的基本概念与基本方法,这门课的学习方法;
- (2) 编程准备与程序的基本结构;
- (3) 变量与问题求解;
- (4) 逻辑思维与问题求解;
- (5) 数据的组织方式一:数组和结构;
- (6) 模块化:函数与函数调用;
- (7) 内存的地址与内容:指针和引用;
- (8) 算法介绍:蒙特卡罗法;
- (9) 算法介绍:贪心法;
- (10) 算法介绍:动态规划;
- (11) 数据的组织方式二:链表;
- (12) 数据的组织方式三:二叉树;
- (13) 输入输出操作:流和文件;
- (14) 程序调试方法。