

现代计算机科学与技术系列教材

高级语言程序设计

实验教程

赵占芳 刘坤起 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

国内各高等院校计算机专业教材

高级语言程序设计

实验教程

张乃宁 刘永明 编



清华大学出版社

现代计算机科学与技术系列教材

工业和信息产业科技与教育专著出版资金资助出版

教育部计算机科学与技术专业综合改革试点（石家庄经济学院）项目资助出版

高级语言程序设计 实验教程

赵占芳 刘坤起 编著

电子工业出版社

Publishing House of Electronics Industry

内 容 简 介

这是一本面向计算机科学与技术类专业及其他专业,全面介绍“高级语言程序设计(含C语言程序设计或Pascal语言程序设计)”实验课程教学要求、教学内容、实验环境及其实施的教材。

本书基于计算机科学与技术一级学科人才培养科学理论,按照计算机科学与技术学科教材系列一体化设计纲要的要求,配合“高级语言程序设计”理论课程的教学,全面介绍了“高级语言程序设计”实验课程的基本实验内容、课程设计及其实验环境——Visual C++ 6.0 和 Delphi 7.0。本书最大特色是结合一些典型实例,系统地介绍了C/C++、Pascal/Delphi 的主要程序调试技术,使学生调试程序从经验走向理性,为大程序的调试奠定了坚实的基础。另外,本书还对 Visual C++ 6.0 和 Delphi 7.0 集成开发环境的配置、使用、程序发布和编译错误信息,以及C/C++、Pascal/Delphi 程序编码规范等内容做了详尽介绍,体现了本书所具有的“工具书”的特点。

本书可作为计算机科学与技术类专业及其他专业“高级语言程序设计”课程的实验教材,也可供高等学校的教师、学生和广大工程技术人员学习高级语言程序设计时参考。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

高级语言程序设计实验教程 / 赵占芳, 刘坤起编著. —北京: 电子工业出版社, 2014.9
现代计算机科学与技术系列教材

ISBN 978-7-121-24337-0

I. ①高… II. ①赵… ②刘… III. ①高级语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 211924 号

策划编辑: 袁 玺

责任编辑: 袁 玺

印 刷: 三河市鑫金马印装有限公司

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 19.75 字数: 505.6 千字

版 次: 2014 年 9 月第 1 版

印 次: 2014 年 9 月第 1 次印刷

定 价: 45.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

“现代计算机科学与技术系列教材” 编审委员会

主 编：顾乃杰、赵致琢（兼执行编委）

学 术 顾 问：陈国良

科学哲学顾问：桂起权、於文辉

委 员（以姓氏拼音排序）：

陈 兵	陈 溧	陈 鸣
陈向群	陈仪香	邓安生
冯志勇	傅育熙	巩志国
管会生	顾乃杰	关 永
何钦铭	李凤华	李陶深
李文军	李勇坚	李元香
林作铨	刘坤起	罗里波
陆汝占	毛晓光	欧阳丹彤
宋方敏	宋 文	孙茂松
徐 涛	闫敬文	张晨曦
张春元	张德富	张功萱
张国印	张明义	张松懋
赵 琛	赵致琢	邹北骥

总 序

民众多好饮酒，中外概莫能外。酒馆和酿酒坊伴随饮酒客而起，人类对酒的喜爱造就了酒文化和一个庞大的产业。好酒能卖好价钱，能使文人诗兴大发，催生佳作，还能解人间百难。于是，酿天下名酒自然成了不少人的毕生追求。

好酒源自粮食，这是众人皆知的常识。中外知名的茅台酒，主要原料系产自贵州赤水河谷的糯红高粱和高寒坡地的大麦、燕麦。为了让茅台酒拥有生态、绿色、有机品质，获得消费者的广泛认同，生产企业和当地农民想了很多办法，一直在搞科学种田。

怎样才能酿出好酒呢？国人的看法不尽相同。崇信洋酒的人主张引进国外的生产工艺，学习洋人的生产和经营理念；而喜欢国酒的人则主张走自己的路，但不排除借鉴国外先进的科学技术和管理经验。这样的争论或许永远不会终结，但外国人重视科学酿酒，值得我们学习和借鉴。

计算机科学教育，如同酿酒工业的生产，科学办学迄今还只是部分学者的一种理想。与国内一样，国外的计算机科学教育并没有像他们的科学酿酒业那样，实现科学办学。也许，科学办学要远比科学酿酒困难得多。譬如，怎么实现科学办学？甚至怎么推出一套科学的系列教材，都是一篇大文章。

科学种田、科学酿酒，科学办学，这些表面上看起来风马牛不相及的事物，其实以科学哲学的观点来看，都有着本质上相同或相似的成分。任何一件复杂的事物，其发展、变化、控制等，都可以通过由外而内，由表及里的方式加以观察和研究，在认识上分解为一系列更小的、基本的事物，通过某种原则、流程、结构、关系、操作、规范等联系在一起，或相互作用，或相互依存，构成一个系统或整体。各种事物发展的背后，有许多带有规律性的、可以重复验证的东西，但多数普通人只关心事物变化、现象的起因和结果，而不太关心其发展、演变的过程，以及事物发生的因果关系及其变化机理。这其中，弄清楚了规律的，可称为科学，而尚未弄清楚规律，又有一些用处的知识，也都保留下来，成了专家的经验。

从经验办学方式转向科学办学方式，从外延发展模式转向内涵发展模式，是高等教育发展的必然要求和规律。然而，科学办学不是一种空洞的口号，而是像科学种田、科学酿酒一样务实的工作，需要通过深入观察、了解其全过程，弄清每一个本质的、核心环节的主要方面和操作细节，把工作做到位，符合客观实际要求，才能从点滴汇聚、涓涓溪流、终成江河，奔向大海。科学办学，内涵发展是高等教育办学理念上的一次重大转变，也是中国高等教育有可能通过努力，迅速赶上西方发达国家的一条途径。毋庸置疑，走上科学办学之路，有许多新事物、新问题，需要人们去学习、思考和解决，用科学方法建设一套系列教材就是其中一个具有挑战性的任务。

这套教材的创作始于教育部面向 21 世纪教育与教学改革 13-22 项目的研究。2000 年，在“13-22 项目”研究工作即将完成之际，一些学者开始认识到面对计算机科学技术的高速发展，我们亟需一套体现科学办学思想，反映内涵发展要求，服务教育与教学改革，参与构建学科

人才培养科学体系的系列教材。强调系列教材是因为那时已经意识到计算机科学教育本质上是一项科学活动，但长期以来教师向学生传授科学技术知识的方式、方法的科学性不强。由于高等教育几百年来一直沿袭经验方式而非科学方式办学，大学教学的方式方法仍然还停留在古代作坊的阶段，尽管教学使用的技术手段今天已经相当先进。在经验办学方式下，无论是研究型大学，还是教学型大学，由于种种原因，教学活动的全过程存在着太多的漏洞和质量的隐患。科学办学是对高等教育界传统办学方式的一种挑战，尽管在认识上，人们不难理解，科学办学是经验办学的最高形式，经验办学应该成为科学办学的有益补充。

“13-22 项目”组积极探索，率先倡导科学办学理念，初步构建了一个体现科学办学思想，反映内涵发展要求的计算机科学与技术一级学科人才培养科学理论体系，为学科专业教育探索新天地，走向科学办学和发展学科系列教材提供了一个认知基础。

长期以来，学术界一直在探索计算机科学与技术专业教育的规律。ACM 和 IEEE/CS 的专家小组在走访了全美 400 多位著名计算机科学家的基础上，以学科方法论作为切入点，开展教学改革理论研究，于 1989 年发表了具有开创性意义的成果，尽管他们并未意识到自己的工作是以学科方法论的研究作为切入点，探讨内涵发展的道路。1990 年前后，在迷宫中探索行走的专家小组，经大师和精英群的指点，实际上已经摸到了走出迷宫的大门，却没有打开并进入一个崭新的天地。这一点从他们在 2000 年网上公布的 CC2001 报告最先删除了 CC1991 报告中有关学科方法论的内容便不难看出（注：后经中国人的提醒又补充写入）。

与此同时，中外教材建设也一直没有停止探索，国内外出版社先后出版了种类繁多的计算机类专业教材。这些教材中不乏精品和上乘之作，但难觅具有鲜明特色、真正一体化设计且符合科学办学要求的系列教材。多数丛书和系列教材基本上还只是出版社对出自作者个人创作的教材，通过冠名“丛书”或“系列”的方法结集出版以求强势效应，仅有少数作者注意到了将几门相近课程组合在一起，进行规划、设计和创作。尽管如此，不少优秀作者和学者理所当然地进入了编审委员会的视线。西方发达国家在计算机学科的领先优势曾使许多人不自觉地将目光转向海外，试图从世界名牌大学使用的教材中去寻找蓝本。遗憾的是与国内一样，经验办学并没有使西方大学在教材建设方面摆脱“各自为政，各行其是”的阴影。此时，我们如梦初醒，毕竟科学办学是前无古人的一项创举。随着学科发展的不断深化，在迈向深蓝知识海洋的今天，外国人未必比中国人在科学办学方面占有更多天时地利的优势。不经意中的发现使我们惊喜和激动，同时深感责任不轻且平添担忧：即使能够写出系列教材的一体化设计，我们是否真能确认这项改革的正确性？真能推出科学的系列教材？可是，除了实践和试验外，我们别无捷径可循。

令人欣慰的是，从 2004 年起，厦门大学、仰恩大学、石家庄经济学院三校先后在常态下，启动、实施了计算机专业科学办学改革试验，在困难的条件下，用实证方法和实验数据，有力地验证了科学办学理念和学科人才培养理论体系的可行性、科学性和先进性，引起了教育部的重视，并且在与流行的人才培养模式的对比试验中表现出明显的优势，为下一阶段在更大范围内推进科学办学，建设新一代系列教材积累了丰富的经验，奠定了坚实的基础。

任何重大变革，不可能一帆风顺，一蹴而就，尤其是当专业办学长期错位运行，积弊甚多而难以顺利转型时。总结历史经验和教训，我们应该清醒地意识到，任何重大变革和科学创新发现，真理最初永远只可能掌握在少数人手中。在科学探索的道路上，热衷于迎合主流观点，人云亦云，只会更多地让机会与自己擦肩而过。值得一提的是，2012 年 9 月，教育部理工处邀请了 100 多所高校的 168 名专家学者，主持召开了“关于仰恩大学计算机科学教学改革案例通信研讨会”。会上，厦门大学、仰恩大学计算机专业的教学改革工作得到一大批专

家学者的积极评价。仰恩作为一所私立大学，在常态化和非常困难的办学条件下，引进和采纳厦门大学创立的一级学科内涵式人才培养模式，在计算机科学系开展科学办学试验，经过学校上下的共同努力，培养了一批质量较高的本科专门人才，二届试点班毕业生整体就业率均达到 95%~100%，在专业技术研发岗位工作的比例均达到 70% 以上。仰恩的改革意义重大，其实践清楚地表明：仰恩大学能够承受高素质创新人才培养模式，一大批公立大学没有理由办不好计算机专业。

从 20 个世纪的 50 年代起，我国几代学者苦苦追赶了西方发达国家半个世纪，靠引进、学习、消化、跟踪、改进、创新的高新技术发展的思维定式，曾使我们付出了高昂的学费和沉重的代价。固然，在高新技术领域，依靠“引进”和“舶来”的次等技术和产品，我们取得了长足的发展和进步，填补了不少国内的“空白”，但在水准上始终与发达国家保持着一段差距，一种在行业内部看来时长时短，难于逾越的差距。这种差距主要表现在对高、精、尖学科的发展中，我们缺乏思想、概念、理论、方法、技术、制度、规范和设计的原始创新和发展模式的全面创新，研究工作总是跟在别人后面亦步亦趋。我们缺乏在发展中另辟蹊径，走自己道路的机制和氛围，迷信洋人，盲目追随西方学术发展道路的习惯性思维方式，几乎导致国人丧失了创新的机能，这是一个国家和民族发展高新技术学科和产业致命的硬伤。

最先进的高新技术，永远不可能从竞争者手中花钱买到。高新技术领域竞争的成败，关键取决于人才与文化。现代科学技术的创新，已不单纯是一个学术问题，还是一个与文化、人文密切相关的问题。科学教育求真求是，技术教育求实求精，人文教育求灵求善，艺术教育求美求新。没有科学技术知识，人的认识和生活难免停留在原始社会，而没有人文精神和艺术的陶冶，科学技术的创新必然失去力量的源泉。可见，走自己的道路，发展中国的科技创新体系，在某种程度和意义上，成败的关键在于大学能否真正培养一大批高素质的人才。高等学校要实现培养大量高素质计算机科学技术人才的目标，需要在前进中不断地进行系统的、科学的总结和深刻的反省，需要对遇到的问题进行科学的分析和判断，作出正确的决策。

工欲善其事，必先利其器。倘若教师不能在思想上摆脱陈旧的思维定式，用先进的理念武装头脑，勇于探索前人没有走过的发展道路，那么，即使采用了世界一流大学的全套教材，恐怕也难于培养出一流的人才。中西文化、人文传统之间的差异之大，中外教育思想、基础教育之间的差异之大，使得中国教育的现代化绝不是一个通过引进和模仿就可以轻易解决的问题。教师的职业不是贩卖知识。授业、传道、启蒙、解惑技能的高低，不仅取决于教师知识的广博和深厚，更重要的在于远见、卓识、探索、创新、敬业、求真的本领和身先垂范。

身处 21 世纪，面对国家的期望，处在科学技术发展浪潮之巅的计算机科学系的教师，任重道远。我们就像茫茫林海中的探险者，环顾苍翠的群山，犹如身陷迷宫一般。计算机科学技术教育，敢问路在何方？其实，我们的出路或许只有一条，那就是系统总结前人的经验和教训，设法努力登上山峰，居高眺望，探寻走出林海的希望之路。

曾有一些人对于一级学科人才培养科学理论体系的可行性表示怀疑。带着这个问题，在中国科学院和国内部分高等学校一大批知名学者的支持下，从 1999 年夏天起，连续六年在贵阳举办了“计算机科学与技术高级研讨班”，向（博士）研究生和中青年教师陆续开设了研究生核心学位课程“高等计算机体系结构”、“并行算法设计基础”、“分布式算法设计基础”、“高等逻辑”、“形式语义学基础”、“可计算性与计算复杂性”、“形式语言与自动机理论”，后来又进一步开设了本科生重点课程“算法设计与分析”、“数理逻辑基础”、“信息安全技术”、“密码学原理及其应用”等一系列课程。六年里，高级研讨班受到同行广泛关注、响应和支持，先后吸引了全国几十所大学四百多人次的师生参加听讲和学习，后发展到由教育部批准资助、

16 所大学联合主办的高级研讨班。高级研讨班上先后产生了一批在科研中取得创新成果并在权威刊物发表论文的学者。实践证明,高级研讨班为中国高校计算机科学技术教学改革和教育质量的提高,发挥了独特的作用,得到国内外一大批学者的充分肯定和好评。高级研讨班正在成为按照一级学科办学和教学改革要求,对计算机科学系教师进行高起点研究生学位课程和本科重点课程进行师资培训的一个模式,有可能对未来计算机科学技术教育产生深远的影响。试想,如果高校教师和培养的研究生,普遍具有高级研讨班所开设的 3~4 门学位课程的基础,不仅科学办学面临的主要困难迎刃而解,各大学科研学术队伍的素质也将得到显著提高。令人遗憾的是,在近年来全国高校科研论文“大跃进”,中国大陆发表论文数高居世界第一的时候,环顾计算机科学技术领域发表的论文,有多少百分比的论文工作基础是建立在上述研究生学位课程基础之上的?准确回答这一问题,也许就能实事求是地认清我们与国际同行在科研水平上的真正差距。因为,这些研究生学位课程知识基础之上的科研成果,总体上代表着整个学科发展的主流、水平和未来发展的趋势。

一些学者对高起点研究生学位课程的必要性提出疑问:是否这些课程都要学习?我们认为,应该看到,在高等教育界从来就存在着两种不同的教育观:一种是专才教育观,另一种是通才教育观。持这两种教育观的人尽管都主张基础知识的重要性,但在对学以致用原则的理解和解释方面存在差异。一般地说,专才教育观主张在一定的基础上,通过深入钻研某一方向的学问,逐步扩展和加深自己的知识,缺什么基础补什么知识,学以致用,逐步成长为一个学科的专家。通才教育观则不同,它不主张在具备一定的基础后,就匆忙沿着某一方向钻研学问,单线独进,而是主张在一级学科的范围内,通过尽可能系统地掌握从事本学科各个重要方向的研究所需要的基础知识,能够站在学科的若干个制高点上,沿着学科的一个方向,以单线独进、多线并进或整体推进的观点,逐步扩展和加深一级学科的知识,融会贯通,学以致用,逐步成长为一个学科的专家。两种教育观都有其代表人物。迄今为止,高等教育中研究生教育主要以培养专才为主,专才教育观是主流。但是,两种不同的教育观各有其特点。一般地说,当一个学科的发展处于早期时,专才教育比较容易跟上学科的发展步伐,比较容易出成果,也比较容易迅速地达到较深的学术层面。而当一个学科的发展比较成熟、发展速度比较平稳时,通才教育的优势就比较明显。因为,通才教育培养的人才可以在一级学科的范围内比较容易地向任何一个方向转向。特别是在胜任高难度重大创新人才的培养方面,在出综合性的大成果方面,在创立一套科学理论和开辟一个研究方向方面,通才教育的多种优势往往是专才教育所不具备的。此时,专才教育培养的人才要继续深入开展创新研究工作是比较困难的,往往会选择边缘跨学科研究或退出。当然,两种教育观谁优谁劣迄今并无定论,根据两种教育观的特点和现实情况,选择哪一种教育观实际仅反映了师生的一种选择策略。不过,实践经验告诉我们,尽管通才教育观的操作实现比较困难,但作为师资补充的来源,通才教育培养的人才更容易适应大学教学与科研的双重要求,理应更多地受到研究型大学的青睐。在科学技术日益深化、高度分化又高度综合的今天,放眼未来,在高、精、尖学科中,通才教育观无疑有着更为宽广而美好的发展前景。

冬去春来,年复一年。当我们终于从跟踪、学习、盲从西方大学教育的发展模式中走出时,感受到了一种从未有过的释然与激动,一种走自己的发展道路,独立自主的自豪与喜悦。这条道路虽然艰难,但前景光明。连续六年在贵阳举办的全国计算机科学与技术高级研讨班的成功实践,更进一步坚定了对内涵发展模式与科学办学方式的认识与追求。

伴随着学科教学改革理论研究与实践探索的推进,社会热切地期待着一套与教学改革方案相配套的高质量系列教材问世。总结过去教材建设成功和失败的经验、教训,使我们清楚

地认识到：教材建设必须建立在科学研究基础之上，按照科学的运作程序，动员在第一线从事科学研究、功底深厚、学有所长、能够在权威刊物发表论文，或在重大教学改革实践中做出显著成绩的优秀教师，参与到教材的创作中来，才有可能推出高质量并符合学科发展要求的系列教材。我们的主张是：“让大学中的科学家来创作教材！”

2014 年年初，电子工业出版社“现代计算机科学与技术系列教材”编审委员会正式成立，计算机科学与技术一级学科系列教材一体化设计报告也即将完成重新修订。编审委员会为系列教材的出版制定了严格、详细的操作程序，选准作者，并在体制创新方面设立学术编审人，跟踪编审教材的创作内容，力求教材的尽善尽美。可以预期，“现代计算机科学与技术系列教材”将是基于计算机科学一级学科人才培养科学理论体系，体现科学办学思想，反映内涵发展要求，开展系列教材一体化设计和建设的一个尝试。然而，就像任何新生事物一样，她难免存在缺点和不足，诚恳地希望关心和使用本套系列教材的师生、读者，在使用中将批评或建议留下来，帮助我们改进教材建设工作，修正存在的错误。

今天，经过编审委员会、作者和出版社的共同努力，“现代计算机科学与技术系列教材”终于开始陆续出版发行。在 21 世纪里，愿“现代计算机科学与技术系列教材”的出版，能够为新一代的莘莘学子攀登现代科学技术的高峰成就未来。

“现代计算机科学与技术系列教材”编审委员会

2014 年 3 月

前 言

为计算机科学与技术专业的学生创作一本有特别实用价值的《高级语言程序设计实验教程》一书，是多年来想做但一直没有做好的事情。原因是目前高等学校教师的工作业绩评价中，编写教材远远比不上搞科研重要。于是，在高等学校中，放下科研，专心于教材创作的教师实属凤毛麟角，尽管教材的创作工作对于人才培养来说是一件非常重要的基础性工作。就这样，我们的编写工作一推再推，直到 2010 年 9 月我们对计算机科学与技术专业进行综合教学改革之时。

其实，在我校计算机科学与技术专业综合教学改革启动之前，我们曾多次讨论“高级语言程序设计实验”课程教学改革的问题。在多年的教学实践中我们发现，相当一部分学生并没有通过该实验课程的教学，真正掌握该门课程的基本实验技能，为后续课程的学习打下坚实的基础，致使一部分学生难以顺利学习后续课程，甚至失去了对专业的学习兴趣，放弃了本专业的学习，其教训十分惨痛！因此，如何大幅度地提高该门实验课程的教学质量，是一个值得探讨的大问题。那么，“高级语言程序设计实验”课程的基本实验技能是什么？究竟如何来开展该门课程的教学呢？

认真分析，仔细想来，从“授人以鱼不如授人以渔”的古训中使我们认识到：由于高级语言的程序调试技术是这门课程的最基本的实验技术，而对这一技术的熟练掌握是计算机科学与技术专业学生必须练就的基本功之一。因此应该把它作为这门实验课程教学的重点。很遗憾，由于高等学校从事计算机科学与技术专业教学的教师中，系统而熟练掌握这一技术的人并不多，加上国内外已经出版的有关教材对程序调试技术的介绍基本上回避或轻描淡写，相应地，在该门课程的教学中学生基本上还是凭着经验调试程序，根本不知道还有一套程序调试技术。因此，计算机科学与技术专业的绝大多数学生普遍没有系统而熟练地掌握程序调试技术，已是一个不争的事实。面对这一现实，为了改变现状，我们提出了“高级语言程序设计实验”课程的教学改革，要围绕着系统介绍主要程序调试技术而展开的思想，并付于实施。经过三年来的教学改革实践，检验了这一改革思想的正确性。

为了顺利实施“高级语言程序设计实验”课程教学改革的思想，我们将主要程序调试技术融入到教材中，利用一年的时间完成了这本教材的初稿（内部讲义），后经三年的使用和修改，形成了如下内容体系。

第 1 章是引言，介绍课程在(学科)专业教育中的地位，课程教学的基本指导思想和理念，课程的内容组织与安排，课程的教学目的和要求，课程实验的分类和文档要求；

第 2 章是程序调试简介，主要介绍程序测试与调试的基本概念，程序调试技术概述，程序错误分类，程序错误定位的方法和学习程序调试技术的意义；

第 3 章是 Visual C++ 6.0 集成开发环境及调试器介绍，主要介绍 Visual C++ 6.0 集成开发环境和调试器的使用；

第4章是C语言程序调试实例。主要介绍在 Visual C++ 6.0 集成开发环境下,结合一些 C 语言的典型实例,如何使用有关的调试技术来调试 C 语言程序。它们是全书的核心部分之一;

第5章是 Delphi 7.0 集成开发环境及调试器介绍,主要介绍 Delphi 7.0 集成开发环境和调试器的使用;

第6章是 Pascal 语言程序调试实例。主要介绍 Pascal 语言的实验内容,以及在 Delphi 7.0 集成开发环境下,结合一些 Pascal 语言的典型实例,如何使用有关的调试技术来调试 Pascal 语言程序。它们也是全书的核心部分之一;

第7章是综合课程设计,主要介绍模块化软件开发方法,软件测试的步骤及技术测试用例设计技术和课程设计的内容和要求。

通过三年来的教学试用和总结,我们提出以下几点意见供使用者参考:

(1) 为了顺利通过“熟练掌握高级语言程序设计的基本技术和主要程序调试技术”这道坎,必须在教学课时上给予保障。宁可牺牲一些没必要的专业课学时,也要保证基础课和专业基础课的学时的做法是完全正确的。值得说明的是,本课程的课内学时为 48 学时是最基本的,实现教学目标学生还需要付出更多的课外时间,至于时间的长短则取决于每一个学生学习的具体情况。课程设计应放在课外时间完成。另外,必须加强实验课程的辅导,要求任课教师每节课必须到场。我们认为,让研究生作为实验课程的教师的做法是不妥的,原因在于绝大多数学生还不具备作为实验课程的教师的基本素质。

(2) 为了培养学生独立解决问题的能力,在程序调试的教学中,一定以介绍调试技术和解决问题的思想方法为主,切不可更多地代替学生调试程序。另外,实验教师在教学中必须贯彻“因材施教”的原则,对于动手能力较强的学生,在回答他们提出的问题时要以引导和启发为主,而对于动手能力较差的学生,在回答他们提出的问题时则要回答得具体些,而且鼓励他们在解决了具体问题后要自觉进行总结。当然,我们不赞成将程序调试技术讲得详之又详,面面俱到,不给学生留下任何思考的余地和悬念,试图在课堂上解决“所有”问题,生怕学生有学不会的内容的做法。而应该鼓励学生面对待解决的问题,独立思考,勇于探索,通过查阅文献,大胆试验,最终解决问题。只有这样,才能使学生在启发式教育下,积极、主动地思考问题,通过艰苦的查阅文献、阅读文献、思考、试验、归纳、总结,才能真正培养学生的自学和动手能力,独立解决问题的能力,探索精神、创新意识和能力。而这种意识和能力的培养正是中国大学实验教学改革所要努力的方向。

尽管本书是“高级语言程序设计”课程的配套教材,专门为其实验课程的教学而编写的,供学生一学期使用,但是由于书中的内容极为丰富,其中包含了计算机科学与技术专业后续软件实验类课程要用到的最基础的内容,因此本书可供学生学习所有软件类实验课程时参考,它也是每一个学生今后走向工作岗位后从事软件开发的一个实验手册,是每个学生调试程序时应该经常翻阅的一本参考书。

在教材即将付梓之际,我们不应该忘记为了本书的写作、修改付出过辛劳的所有老师、学生和友人。非常感谢厦门大学计算机科学与技术系教授赵致琢博士,尽管他没有直接参与本书的创作,但从本书讲义的编写到今天的公开出版,始终得到了他的关心和指导。赵致琢教授对计算机科学与技术教育事业倾注了大量心血,提出了“科学办学,内涵发展”的高等教育办学思想和计算机科学与技术一级学科人才培养的科学理论体系,亲自指导了我校计算机科学与技术专业的综合教学改革,对我国计算机科学与技术教育事业的发展可谓贡献良

多。他对教育的挚爱，矢志不移、勇往直前、追求真理的精神尤为作者感动，这种可贵精神必将化为对我们的永远激励。

我们还要感谢石家庄经济学院 2010 级、2011 级和 2012 级计算机科学与技术专业教学改革试点班上的所有学生和参与课程教学改革的年轻教师，是他们在我们边创作、边讲授，边修改、边使用的情况下，积极踊跃地提出问题，发现并更正了讲义中的错误，使讲义得以完善，课程的教学质量逐年提高。而我们从师生的共同讨论中对“教学相长”这一成语的内涵有了最直接的感受，同时也体会到作为传道者因师生共同成长而带来的愉悦。

感谢贵州民族大学的杨承中教授和仰恩大学的陈发强教授，正是他们对我们的信任，在讲义尚未公开出版的情况下，依然将我们的内部讲义作为其各自 2012 级计算机科学与技术专业教学改革试点班的实验教材，并在使用中提出了许多宝贵意见，为讲义的进一步修改做出了贡献。两位教授给予我们的信任和建议，为本书的编写提供了重要参考，也使拙作增色不少，感激之情，溢于言表。

令人欣慰的是，作者受赵致琢教授之邀，以本书的内部讲义为教材，在刚刚结束的“暑期全国计算机科学与技术科学办学与师资培训高级研讨班（贵阳·花溪）”上，为来自全国几十所高等学校的几十名教师主讲了 20 学时的“程序调试技术”课程。我们从这些同行的认真、积极而热烈的课堂讨论中深受启发，促使我们对本教材出版前做了最后一次大的修订，使本教材的适用面进一步拓宽——它不仅适用于“Pascal 语言程序设计”实验课程的教学，而且也同样适用于“C 语言程序设计”实验课程的教学。谢谢这些来自全国几十所高等学校的老师们！

当学生们打开此书时，从其极为丰富的内容和大量细致的工作中不难看出其写作背后作者所付出的辛勤劳动。四年来，作者在没有鲜花和掌声，也没有物质上的奖励和金钱的诱惑下，几乎牺牲了所有课余闲暇，一心扑在教材的写作上，只是真诚地希望学生们通过该课程的学习，能够打下程序设计的坚实基础和培养出强大的调试程序的能力，使学生们顺利踏上学习计算机科学与技术专业之路。如果这本书能够伴随着学生的学习，使他们能够感受到程序设计与调试程序的魅力，激发他们学习计算机科学与技术专业的兴趣和热情，那将是作者最高兴的事情，也是对作者艰辛劳动的最好回报。

尽管本书的创作历时四载，但是由于我们业务水平的限制和工作中的疏忽，书中仍有许多地方需要进一步推敲、修改和完善，恳请读者把对这本书的意见和建议反馈给我们，帮助我们改进工作，完善这本实验教程。第一作者的电子邮件是：zzf_sjz@126.com。

作 者

目 录

第 1 章 引言	1
1.1 高级语言程序设计实验教学在计算机科学与技术专业教学中的地位	1
1.2 高级语言程序设计实验教学的基本指导思想和理念	1
1.3 高级语言程序设计实验教学的内容组织与安排	2
1.4 高级语言程序设计实验教学的目的和要求	2
1.5 高级语言程序设计实验的分类	3
1.6 高级语言程序设计实验文档	3
第 2 章 程序调试简介	4
2.1 计算的正确性问题	4
2.2 程序测试与程序调试的关系	4
2.3 程序调试的概念及其技术概述	4
2.3.1 程序调试定义	5
2.3.2 程序调试基本过程	5
2.3.3 程序调试分类	5
2.3.4 程序调试技术概览	6
2.4 程序错误的分类	7
2.5 程序错误定位的方法	8
2.5.1 增量调试法	8
2.5.2 流程观察的程序插装法	8
2.5.3 数据透视法	8
2.5.4 分离法	8
2.5.5 屏蔽法	9
2.5.6 回溯法	9
2.5.7 二分法	9
2.6 学习程序调试技术的意义	9
第 3 章 Visual C++ 6.0 集成开发环境及调试器介绍	10
3.1 Visual C++ 6.0 集成开发环境	10
3.1.1 Visual C++ 6.0 开发环境	10
3.1.2 主窗口	10
3.2 控制台应用程序	16
3.2.1 创建控制台应用程序	16

3.2.2	编辑并保存文件	19
3.2.3	编译并运行项目	19
3.2.4	控制台应用程序项目的组成	20
3.3	定制控制台应用程序的开发环境	20
3.3.1	菜单与工具栏的定制	20
3.3.2	项目配置	21
3.3.3	开发环境定制	25
3.4	Visual C++ 6.0 的使用技巧	26
3.4.1	Visual C++ 6.0 的一些实用技巧	26
3.4.2	Visual C++ 6.0 开发环境设置技巧	27
3.4.3	使用 Visual C++ 6.0 的 MSDN	27
3.5	程序的编译、链接和运行	28
3.6	调试环境的配置	29
3.7	使用集成调试器进行程序调试	30
3.7.1	控制程序的执行	30
3.7.2	断点的使用	32
3.7.3	使用调试窗口查看或修改变量的值	47
3.8	其他的调试手段	51
3.8.1	使用函数或宏进行程序调试	51
3.8.2	远程调试	52
3.8.3	实时调试	52
3.8.4	编辑并继续调试	52
3.8.5	其他的调试功能	52
3.9	程序的调试版本和发布版本	52
3.9.1	生成调试版本和发布版本	53
3.9.2	调试版本与发布版本的区别	53
3.9.3	调试发布版本	54
3.10	小结	56
第 4 章	C 语言程序调试实例	57
4.1	基础知识程序调试实例	57
4.2	三种基本结构程序调试实例	63
4.3	基于数组的程序调试实例	73
4.4	函数的程序调试实例	80
4.5	结构体的程序调试实例	88
4.6	基于指针的程序调试实例	92
第 5 章	Delphi 7.0 集成开发环境及调试器介绍	105
5.1	Delphi 7.0 集成开发环境	105
5.1.1	Delphi 7.0 的开发环境	105
5.1.2	主窗口	106

5.2	控制台应用程序	109
5.2.1	创建控制台应用程序	109
5.2.2	编辑并保存工程文件	110
5.2.3	编译并运行工程	111
5.2.4	控制台应用程序工程的组成	111
5.3	定制控制台应用程序的开发环境	112
5.3.1	窗口和工具栏的定制	112
5.3.2	编程环境的定制	113
5.3.3	代码编辑器环境的定制	114
5.4	Delphi 7.0 的使用技巧	116
5.4.1	代码编辑器使用技巧	116
5.4.2	调试环境配置技巧	117
5.4.3	帮助的使用	117
5.5	程序的编译	117
5.6	Delphi 7.0 调试环境的配置	118
5.7	使用集成调试器进行程序调试	121
5.7.1	控制程序的执行	121
5.7.2	断点的使用	121
5.7.3	查看变量的值	124
5.7.4	其他调试方法	126
5.8	其他的调试工具	127
5.9	程序的发布	127
第 6 章	Pascal 语言程序调试实例	129
6.1	第一单元——顺序程序设计和分支程序设计	129
6.1.1	简单 Pascal 程序的上机过程	129
6.1.2	顺序程序设计	135
6.1.3	分支程序设计	143
6.2	第二单元——循环程序设计	155
6.3	第三单元——构造类型	166
6.3.1	枚举、子界与数组	166
6.3.2	集合与记录	179
6.4	第四单元——分程序	189
6.5	第五单元——递归技术	208
6.6	第六单元——指针与动态数据类型	215
6.7	第七单元——文件	233
第 7 章	综合课程设计	247
7.1	模块化软件开发方法	247
7.1.1	模块化设计方法	247
7.1.2	程序的分块开发	248

7.1.3	工程文件	250
7.2	软件测试的步骤	250
7.2.1	单元测试	251
7.2.2	集成测试	252
7.2.3	确认测试	252
7.2.4	系统测试	252
7.2.5	验收测试	253
7.3	软件测试技术	253
7.4	测试用例设计技术	253
7.4.1	逻辑覆盖法	254
7.4.2	等价划分法	254
7.4.3	边值分析法	255
7.4.4	错误猜测法	255
7.5	课程设计	255
7.5.1	课程设计实施要求	255
7.5.2	课程设计的功能设计要求	256
7.5.3	课程设计的总体设计要求	258
7.5.4	课程设计报告文档要求	258
附录 1	Delphi 7.0 集成开发环境配置	260
附录 2	Delphi 编译错误信息一览表	262
附录 3	Pascal/Delphi 程序编码规范	268
附录 4	Visual C++编译、链接错误信息	280
附录 5	C/C++程序编码规范	287
参考文献		297