

浅析计算机教学改革

郭强强

东北大学，沈阳

摘要 | 随着计算机技术和知识的普及，非计算机专业与计算机专业在程序设计方面的差距逐渐缩小，计算机专业的程序设计课程改革势在必行。详细分析了高级语言程序设计课程建设的背景以及课程的主要问题，从教学目标、实验环节、教学内容、考试改革等方面提出了课程改革方案。

关键词 | 程序设计；教学改革；高级语言

Copyright © 2022 by author (s) and SciScan Publishing Limited

This article is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/). <https://creativecommons.org/licenses/by-nc/4.0/>



程序设计是计算机专业学生的基础课程，程序设计能力是计算机专业人员，特别是软件工作者的重要专业基础能力，C 语言是目前计算机专业工作者的第一语言。在计算机技术尚未普及的时候，掌握 C 语言的基本语法和语义，能够使用 C 语言编程，就足以使计算机专业的学生在专业知识和技能方面明显区别于非计算机专业的学生。然而，随着计算机技术和知识的普及，越来越多的非计算机专业的学生也在这一层次和深度上掌握了 C 语言的知识 and 运用能力。与此同时，计算机专业在高级语言程序设计的教学方面依然采用原来的教学内容和教学方法，在程序设计知识的深度和广度以及编程语言的运用能力方面没有大的进展，导致计算机专业学生与非计算机专业学生之间在这方面的专业能力差

作者简介：郭强强，东北大学学生。

文章引用：郭强强. 浅析计算机教学改革 [J]. 现代计算机技术与应用, 2022, 4 (2) : 50-56.

<https://doi.org/10.35534/mcta.0402008c>

距逐渐缩小甚至消失。计算机专业学生所完成的程序在程序结构、程序的可靠性、可维护性、可扩展性、程序的时空效率、以及程序的风格和可读性等方面与非计算机专业学生的程序差别不大，而非计算机专业的学生，由于其或者具有较强的抽象思维能力，或者具有相关应用领域的背景，因而可能在程序的算法选择、易用性、以及功能的完备性等方面具有更好的表现。

1 高级语言程序设计课程建设的背景

随着计算机技术的迅速发展和日益普及，越来越多的高校毕业生涌入计算机技术及应用产业，特别是软件产业。这些人中不仅有计算机专业的毕业生，也有诸如数学、电子、自动化、机械等其他大量的理工科专业的毕业生以及部分文科专业的毕业生。在与这些非计算机专业的学生的竞争中，计算机专业的学生并未表现出明显的专业优势。相反，有些非计算机专业的学生由于具有本身的专业知识和训练，在一些应用软件的研发领域表现出了更强的适应性和竞争力。造成这一现象的原因多种多样，其中重要的一点就是，计算机专业学生所掌握的一些知识和技能，特别是一些基础专业知识和能力，缺乏专业性的深度和广度。从学校教育的角度来看，可以说计算机专业学生所学的一些专业课程和专业基础课程没有为学生提供足够的专业知识和训练，是一个不可忽视的因素。高级语言程序设计课程就是这样一门在专业性方面有待加强的专业基础课程。

2 高级语言程序设计课程的主要问题

造成上述这种情况的原因，从教学方面看主要有5点：①课程定位不明确；②高级语言程序设计课程的教学内容偏窄偏浅；③教学方法不适当；④实验考核手段落后；⑤学生对学习内容和方法不适应。

2.1 教学目标和指导思想

长期以来高级语言程序设计这门课程定位不明确，目前的教材或二级考试中的一些例题和习题，侧重于语法细节，与实际的程序设计能力相距甚远，究竟是教语言还是教编程，是注重知识还是能力的培养？由于对专业能力的培

养重视不够,导致与非计算机专业教学的区别不明显。另外对课程的实践性重视不够,对书本知识过于偏重,实际知识运用的训练较少、较浅。

2.2 课程的教学内容

高级语言程序设计在教学内容上偏重于语言本身,在程序设计方面注意不够。教学内容以语言要素为主组织教学,着重介绍语法、语义、语句以及简单的应用举例,讲解过于强调语言的知识点,如for语句,while语句、switch语句的格式、算术、条件、逻辑表达式的计算、自增自减运算的不同写法、变量的存储类别、变量生存期等。讲解了各种语法规则,但学生抓不住重点,不会将纷繁的知识点融会贯通。由于缺少较深入讨论程序设计的内容,缺少对语言运用能力培养和训练的内容,使得该课程成了一门思维游戏课,学生学到是只是语法,并没有掌握结构化程序设计的思想和方法,没有具备通过编程解决实际问题的能力。

2.3 教学方法

课程的目标是培养学生分析问题和运用程序设计语言解决实际问题的能力。该课程对编译原理、数据结构、算法分析、软件工程等后续课程的学习有重要的影响,程序设计的逻辑思维直接影响到计算机专业学生在大学学习期间的学习方法。但在课程教学过程中未能充分认识到高级语言程序设计是一门实践性很强的专业基础课,高级程序设计通常混同于一般的理论课教学,因而在教学过程中重视概念的讲解而忽视概念的运用,重视知识的讲授而忽视对能力的培养,实践环节重视不够。

2.4 练习和考核手段

在练习和考试中,由于缺少必要的技术手段,难以提供实际程序设计练习的机会和条件,因此多采用书面方式,缺少对知识灵活运用的练习题,缺少专业性编程基础的练习题,缺少对程序实际运行的检验。这样,一方面使得学生很难把注意力集中到提高实际的程序设计能力上面,另一方面也使得教师在评判学生的练习和考试答卷时面临巨大的工作量和难度,因此只能尽量多出一些选择题、填空题等标准化试题,对于少量的程序设计题,也只能更多地注意学

生答案的基本思路是否正确,是否有明显的错误,而很难判断学生的答案是否可以在计算机上运行,是否可以满足题目的各项要求,产生正确的结果,以及程序在风格和结构上是否合理等。这样也从另一方面引导学生把注意力更多地放在对一些基本概念和语言要素的学习上,而忽视了对程序设计能力的锻炼和培养,因此教学质量难以把握和提高。

2.5 学生方面的问题

从学生学习的角度看,也存在对学习内容和方法不适应的问题。大多数的学生从小学到中学,已经适应了记概念、背知识点、应付书面考试的学习模式,对于灵活运用所学的知识去分析和解决问题,缺乏必要的训练,因此不具备这方面的能力。当学生第1次接触程序设计课程时,依然会沿用原来的思维模式和学习方法。特别是当程序设计教学仍然是以程序设计语言本身为主、以概念为主、以书面教学为主时,就更会强化学生原有的思维模式和学习方法,把这门课程当成以概念和知识点的学习为主的理论型课程,忽视了对各种实际能力的培养。因此,很多学生尽管在书面考试中的成绩尚可,但是在实际的程序设计能力方面普遍偏低,即使是百行以内的简单程序也很难在较短的时间内地完成,更谈不上具有较为专业化的程序设计能力了。

3 课程结构和组织

为使学生通过编程语言的学习掌握程序设计的能力,养成良好的程序设计习惯,并在分析问题和解决问题的能力上得到培养和锻炼,需要在课程的教学内容和侧重点上进行必要的调整,在教学方法上进行必要的改进,在实验和考核的手段上进行必要的革新,在教学过程的组织和安排上与其它相关课程更好地协调。

3.1 以程序设计能力的培养为目标

根据课程的指导思想制订了新的教学大纲和课程安排。新的教学大纲明确了本课程的基本教学目标是培养学生良好的程序设计能力和习惯,掌握运用编程语言的知识和能力,为学生进一步全面掌握专业化的程序设计知识和能力打下基础。

新的教学大纲以程序设计能力的培养为主线，根据循序渐进的原则安排相关的课程内容和进度，根据学生的知识水平和学习过程中的认知规律，把整个教学内容分为2个有机结合又相对独立的阶段。第1阶段教学内容以培养学生的基本程序设计能力为中心，第2阶段的教学重点以培养学生的专业程序设计能力为中心。

3.2 加强实践环节的设置

其次应当认识到，程序设计课程是一门实践性很强的专业技术课程。要鼓励学生多读典型程序，认真思考算法，自己编写并上机调试程序，通过调试程序来发现语法错误，检查逻辑错误，从而提高编程能力。程序设计的知识和能力不是仅仅通过书本知识的学习就可以掌握的。只有加强实践环节，通过理论学习与实际编程练习相结合的方法，让学生在失败和成功的实践中深入体会，通过自己的经验积累才能真正理解书本上的知识，切实掌握实际运用的能力。因此，在课程教学的过程中不仅要重视理论教学，重视教学内容的设置，也要重视实践环节的设置，重视为学生提供具有及时监控和反馈环节的实践环境。这样不仅可以使学生深入理解和掌握程序语言的知识 and 运用程序设计语言的能力，而且可以培养学生良好的理论与实际相结合的习惯，提高学生分析问题和解决问题的能力，提高学生学习的积极性。通过上机练习使学生进一步熟悉C语言的语句和语法规则，不断深化理解各种不同的数据类型、结构化程序设计的基本思想。为此要增加必要的上机时数，以及必要的编程题目数量和难度，做到课堂教学与编程练习相结合。

3.3 教学内容详略得当、重点突出

在高级语言程序设计课程的教学内容和侧重点上需要进行必要的调整，注重知识点的取舍，讲授的角度和深度符合学生的知识水平和认知规律，与编程紧密结合，优化教学内容。把教学的重点从编程语言本身转移到程序设计的思路和方法、以及运用编程语言进行程序设计上来，把讲授语言的概念转变为讲授语言运用，把对概念的讲授由抽象的分析和讲解转变为通过实例和经验对概念的学习。以往常规的程序设计教材和教学方法主要是以编程语言教学为中心，

围绕着编程语言本身进行。课程以变量、表达式、语句、程序结构、数组、指针、结构等编程语言要素为单元逐一讲解相关的知识点，而只把实际的程序作为说明编程语言要素的示例。为培养学生的程序设计能力，教学内容应当围绕程序设计的思想和方法进行组织，把编程语言作为讲解程序设计的工具，在讲解程序设计的思想、方法和步骤的过程中，根据程序描述的需要，本着由简到繁，由浅入深的原则，逐步讲解编程语言中的知识点，将编程语言要素贯穿于结构化程序设计的顺序、分支、循环3种结构之中，通过地址将数组、指针、函数3者有机联系起来，使得纷繁的知识点形成一个完整、统一的知识体系。例如在初学阶段对于一些不常用或非关键的内容如编译预处理、输出格式涉及的数制转换、宏替换、指针函数、二级指针、变量的存储类别与生存期等都可以不讲，今后在计算机组成原理、数据结构的课程中会学到数制转换、位运算、链表等复杂的数据结构。通过与其他课程的衔接，在多门课程的反复学习过程中编程水平会呈螺旋式上升。

3.4 考试改革

在课程的考核环节，应该以检验学生的实际程序设计能力为主，关键概念为辅。采用笔试+上机考试+课程设计三位一体的方式。对于语言的语法细节和使用方式，则不应单独作为考核的题目，而应该与对程序设计能力的考核有机地融和在一起。例如应当避免表达式求值、 $*p++$ ， $(*p)++$ ， $*p--$ ， $(*p)--$ 自增自减运算、判断题、改错题、概念问答题这类关注语言细节的题目，上机考要求学生按指定的题目编写程序、调试、运行程序得到正确结果，课程设计由学生自选一些有实用性的题目，分组完成，或组织参加一些程序设计竞赛，这样才能够引导和鼓励学生真正把学习的注意力集中到程序设计的能力培养方面，从而改变读书、背概念、应付考试的学习模式。这种考核方式最大程度上发挥了学生的自学能力和主动性，是实现创新性人材培养的途径。

4 小结

由于高级语言程序的重要性，通过该课程培养学生良好的程序设计的知识

和能力,理论联系实际的习惯,自学能力,分析和解决实际问题的能力,为后续的专业课程奠定了基础。

参考文献

- [1] 尹宝林. C 程序设计思想与方法 [M]. 北京: 机械工业出版社, 2009.
- [2] 张基温. 组织学生参加计算机统一考试该不该叫停 [J]. 计算机教育, 2004 (9): 268-269.
- [3] 苏丹丹, 孟祥锐, 褚丽娟. 高校非计算机专业 C 语言程序设计课程四位一体教学模式研究 [J]. 长春师范学院学报 (自然科学版), 2008, 27 (5): 129-132.

Analysis of Computer Teaching Reform

Guo Qiangqiang

Northeastern University, Shenyang

Abstract: With the popularization of computer technology and knowledge, the gap between non-computer majors and computer majors in programming is gradually narrowing. The program design curriculum reform of computer majors is imperative. This paper analyzes the background and main problems of the course construction of advanced language programming in detail, and puts forward a curriculum reform plan from the aspects of teaching objectives, experimental links, teaching contents and examination reform.

Key words: Program design; Teaching reform; A high-level language