

程序设计基础课程中学生编程能力培养的问题与对策

赵靓博

北京金融科技学院 北京 101118

【摘要】：随着信息技术的飞速发展，编程能力已成为现代学生必备的重要技能。然而，当前程序设计基础课程在学生编程能力培养方面仍存在诸多问题，如教学方法单一、实践环节不足以及学习兴趣不高等。本文围绕这些问题展开分析，提出了优化教学模式、强化实践训练和激发学生兴趣等针对性对策，旨在提升学生的编程能力和综合素养，为信息技术人才培养提供理论支持和实践指导。

【关键词】：程序设计基础；编程能力；教学问题；培养对策；信息技术教育

DOI:10.12417/2705-1358.25.17.071

引言

在数字化时代背景下，编程能力逐渐成为推动社会发展的关键技能。程序设计基础课程作为学生学习编程的入门环节，其教学质量直接影响学生的学习效果和未来发展。然而，传统教学模式难以满足学生多样化的学习需求，导致部分学生编程水平提升缓慢。如何有效培养学生的编程能力，成为教育工作者亟需解决的问题。本文将深入探讨程序设计基础课程中存在的问题，并提出切实可行的培养策略，助力提升学生编程能力和创新实践水平。

1 程序设计基础课程中学生编程能力培养存在的主要问题分析

在当前信息技术教育不断深化的背景下，程序设计基础课程作为培养学生编程能力的关键环节，承载着培养创新型人才和提升学生信息素养的重要使命。然而，现实教学过程中暴露出诸多问题，严重制约了学生编程能力的有效提升。教学内容设计过于理论化，缺乏对实际编程技能的有效引导，导致学生难以形成系统的编程思维和解决问题的能力。许多课程依然停留在语言语法和基本算法的讲解，缺少对复杂逻辑结构和实际项目的深入探讨，使得学生的编程知识较为零散，难以应用于实际情境中。此外，课程内容更新滞后，不能紧跟编程技术的快速发展，教学内容与现实技术应用之间存在明显脱节，无法满足新兴产业对编程人才多样化技能的需求。教学资源配置不足也是显著问题之一，硬件环境、软件工具以及实验平台的匮乏，限制了学生动手实践的机会，降低了学习的主动性和探索欲望。部分学校因经费、师资力量等限制，难以提供多样化的教学支持，学生只能依赖传统的课堂讲授，缺乏足够的编程实践和项目训练，导致理论与实践脱节，难以培养出具有实际操作能力的编程人才。

学生自身编程基础薄弱以及学习动机不足也是制约编程能力培养的重要因素。程序设计作为一门逻辑性极强的学科，

对学生的抽象思维、算法设计能力和代码实现能力提出了较高的要求。许多学生入门阶段缺乏必要的计算思维训练，不能迅速理解程序运行机制和调试技巧，导致学习过程中出现较多困惑和挫折感，影响学习积极性。另一方面，编程学习过程中容易遇到复杂错误和调试难题，如果缺乏有效的辅导和及时反馈，学生容易产生畏难情绪，进而影响对课程的兴趣和坚持度。教学过程中对学生个体差异的关注不够，统一的教学进度和内容安排忽视了不同学生的学习基础和接受能力，无法实现差异化教学，导致部分学生跟不上课程节奏，出现学习瓶颈。此外，缺乏系统的学习评价体系，使得学生难以准确了解自身编程能力的真实水平，缺乏明确的学习目标和成长路径，也难以形成持续改进的动力和方法。学生在编程学习中缺乏真实项目驱动，导致学习内容与实际应用脱节，难以形成解决实际问题的能力，限制了综合编程素养的培养。

教学方法单一和实践环节不足也极大影响了学生编程能力的提升。传统程序设计课程往往以教师讲授为主，缺少互动式教学和启发式引导，难以激发学生的学习兴趣 and 自主探索精神。课堂上强调理论知识的传授，而忽视对编程思维的培养和创新能力的训练，学生缺乏解决实际问题的机会和能力。实践教学环境建设滞后，实践课程安排缺乏系统规划和科学设计，学生参与项目开发的机会有限，实践内容多为单一的编程练习，缺乏跨学科综合应用和团队协作经验。实验课程的安排和考核形式未能充分体现实际开发流程和工程管理理念，难以帮助学生形成规范的代码编写习惯和良好的软件工程素养。

2 影响学生编程能力提升的关键因素探讨

编程能力的提升不仅依赖于课程内容的设置，更受多方面因素的综合影响。其中，教学方法和教学设计的科学性直接决定了学生对编程知识的理解深度和掌握程度。当前程序设计基础课程普遍存在以讲授为主、缺乏互动和探究式学习的现象，限制了学生思维的活跃性和问题解决能力的培养。面向对象编程、数据结构与算法等核心内容如果仅停留在理论层面，学生

难以将抽象概念转化为实际编程技能。同时,教学过程中缺乏针对性和差异化设计,使得不同基础和兴趣的学生无法获得适合自身的学习资源和指导,削弱了个性化学习路径的构建。课程中的项目驱动和案例教学应用不足,导致学生难以在真实或仿真环境中进行编程实践,影响了对复杂问题拆解和系统设计的能力形成。有效的教学设计应注重激发学生的探究精神,提供多样化的学习方式,如协作编程、问题导向学习(PBL)等,以促进编程思维的多维度发展。

学生自身的认知结构和学习态度也是影响编程能力发展的重要因素。编程作为一门逻辑性强、抽象度高的学科,要求学习者具备较强的抽象思维能力和系统分析能力。部分学生在思维方式上尚未完全适应程序设计的逻辑,面对复杂算法和程序结构时容易产生认知负荷,造成理解障碍。此外,缺乏持续的学习动机和目标导向,使得学生在编程学习中表现出易于放弃和缺乏耐心的情况。自我效能感不足会降低学生主动探索和尝试的意愿,进而影响学习效果。学习策略的缺失也是一大障碍,许多学生未能掌握科学的学习方法,如模块化思考、调试技巧及代码重构等,导致编程效率低下。编程能力的提升不仅依赖于技能训练,更需培养学生的元认知能力,使其能够在学习过程中进行自我监控与调整,从而实现更深层次的知识内化和迁移应用。

技术环境和教学资源的供给状况同样对学生编程能力的提升发挥着关键作用。现代程序设计课程离不开高效稳定的编程环境与丰富的教学辅助工具。硬件设备性能不足,尤其是在需要运行复杂算法和图形界面的程序时,直接限制了学生实验的开展。软件平台的选择与更新滞后,也影响了教学内容的现代化和实用性。缺少云端协作平台、代码版本控制工具以及在线编程竞赛资源,使得学生难以获得团队协作和实战演练的机会,影响了综合编程能力的形成。师资力量与教学团队的专业素质构成了影响编程教学质量的另一核心因素。教师对最新编程语言、开发框架以及行业发展趋势的了解与掌握程度,决定了教学内容的前瞻性和实用性。不够及时更新教学资源或缺少对学生个性化辅导的能力,限制了教学效果的提升。教学评价体系单一,未能科学衡量学生的编程思维、代码质量与解决实际问题的能力,也削弱了学生提升自我能力的动力。

3 优化程序设计课程教学模式的策略与实践

程序设计基础课程的教学模式优化是提升学生编程能力的关键环节,切实有效的教学改革不仅能够激发学生的学习热情,还能增强其实践操作能力和创新意识。传统教学模式以教师讲授为主,缺乏互动性和学生主体地位的体现,这在很大程度上制约了学生编程思维的形成和技能的提升。优化教学模式应从多方面入手,构建以学生为中心的教学体系,积极引入项目驱动法、翻转课堂和混合式教学等先进教学理念,将理论学

习与实践操作紧密结合。通过设计真实的编程项目和案例,培养学生的需求分析、算法设计、代码实现及调试能力,促使学生在动手实践中深化对知识的理解与应用。实践环节的强化不仅使学生掌握基本编程技能,更有助于提升其系统设计和解决复杂问题的能力,推动学生从被动学习转向主动探索和创新。

教学资源的合理配置和信息化教学工具的运用,是优化程序设计课程教学的重要支撑。现代教育技术的发展为编程教学提供了丰富的工具和平台,如集成开发环境(IDE)、在线编程竞赛平台、代码托管和协作平台等,这些工具极大地便利了教学与学习过程。通过引入智能化编程辅助工具,学生能够实时获得代码提示和错误反馈,提高编程效率和质量。借助云计算和虚拟实验环境,突破了硬件资源限制,为学生提供更加灵活的学习空间和资源共享渠道。同时,教学平台的搭建使得教师能够及时跟踪学生的学习进度和问题,开展个性化辅导和针对性训练,满足不同层次学生的学习需求。

教师队伍建设是保障教学模式优化实施效果的核心环节。高素质的教师不仅具备扎实的专业知识和丰富的编程实践经验,更应具备先进的教学理念和教学方法。加强教师的信息技术培训,提升其对新兴编程语言、开发工具和教学技术的掌握,是提高教学质量的重要途径。教师应成为学生学习的引导者和协助者,注重培养学生的创新思维和解决问题的能力,鼓励学生开展自主学习和团队合作。通过开展教学研讨、教学观摩和校企合作,促进教师之间的经验交流和教学资源共享,形成良好的教学研究氛围。此外,建立科学合理的教学评价体系,结合过程性评价和终结性评价,全面评估学生编程能力的多方面表现,激励教师持续改进教学内容和教学方法。教师素质的不断提升和教学理念的更新,将为程序设计基础课程的教学模式优化提供坚实保障,最终实现学生编程能力的有效提升和综合素质的全面发展。

4 提升学生编程能力培养效果的总结与展望

提升学生编程能力培养效果的总结与展望,意味着在系统分析现有教学状况和实践成效的基础上,提出持续优化的路径,推动编程教育不断迈向高质量发展。程序设计基础课程作为培养计算思维和编程技能的关键环节,承载着学生未来信息技术素养的培育重任。经过教学模式的改革与教学资源的投入,学生的编程实践能力和问题解决能力已得到明显提升,课堂互动性和自主学习意识也有所增强。教学中项目驱动法和翻转课堂的实施,使学生从被动接受转向主动探索,促进了理论知识与实际操作的有效结合。多样化的评价体系不仅考察了学生的代码质量和程序效率,更关注其创新能力和团队协作精神,从而推动学生综合能力的全面发展。实践证明,只有将理论与实践有机融合,强化创新意识和工程思维的培养,才能真正提升学生的编程能力和适应未来技术发展的竞争力。

在总结阶段,不可忽视的是编程教育仍面临的多重挑战,未来需进一步优化教学内容与方法。现有课程体系虽然已涵盖核心编程知识,但对于新兴技术如人工智能、云计算及大数据的引入尚显不足,难以满足行业快速发展的需求。课程内容更新速度滞后,限制了学生对前沿技术的理解和应用能力的培养。教师队伍专业化建设仍需加强,部分教师对新技术掌握有限,影响了教学效果和学生学习兴趣。学生个性化发展需求日益突出,传统“一刀切”的教学方式难以兼顾不同基础和兴趣层次的学生。教学资源分配不均,部分院校硬件设施和软件环境落后,制约了编程实验和项目实践的广度与深度。学生学习动力不足和自我管理能力欠缺,也对编程能力的稳步提升构成障碍。综合来看,提升编程能力培养效果是一项系统工程,需从课程内容、教学方法、师资建设、资源保障和学生培养等多方面协同推进。

面向未来,提升学生编程能力的培养效果必须注重教育理念的革新和技术手段的融合。引入智能化教学辅助工具和大数据分析技术,能够精准评估学生学习状况,提供个性化学习方案,有效弥补传统教学中个别化辅导的不足。构建开放式学习平台,整合线上线下资源,推动跨校、跨区域的教育协同发展,

促进优质教育资源共享。同时,深化产教融合,积极引入企业真实项目和行业专家参与课程设计与教学,实现学科知识与职业技能的无缝对接,提升学生的实践能力和就业竞争力。强化学生的计算思维培养,将算法设计、程序优化与软件工程理念贯穿教学全过程,促进学生形成系统的技术视野和创新思维。师资培养方面,需建立持续的专业发展机制,推动教师不断学习新技术和教学理念,提高教学适应性和创新能力。通过多方协同与持续改进,努力打造符合新时代信息技术教育需求的高效教学体系,推动学生编程能力培养迈向更高水平,为信息社会培养具有创新能力和实践能力的高素质人才奠定坚实基础。

5 结语

提升学生编程能力培养效果是程序设计基础课程改革的核心目标。通过优化教学模式、完善评价体系和强化实践环节,可以有效激发学生学习兴趣,提升其综合编程素养。面对信息技术的快速发展,课程内容和教学方法需不断创新,融合智能化手段与跨学科资源,满足多样化的学习需求。未来,应加强校企合作与产教融合,拓展实践平台,促进学生能力的全面提升。

参考文献:

- [1] 王明华.程序设计基础课程教学模式改革研究[J].现代教育技术,2021,31(12):45-50.
- [2] 李晓红.基于项目驱动的编程能力培养实践探索[J].计算机教育,2020,38(8):72-77.
- [3] 陈伟东.高校学生编程能力影响因素分析[J].计算机与教育,2019,27(6):88-93.
- [4] 刘婷婷,赵强.翻转课堂在程序设计课程中的应用研究[J].电化教育研究,2022,43(4):101-107.
- [5] 何建华.信息技术背景下高校程序设计课程教学创新[J].现代远程教育研究,2021,33(3):56-62.