

基于成果导向教育的计算机编程开发类课程实践教学改革研究

张 辉

南宁学院 广西南宁 530001

摘 要: 本研究基于成果导向教育(OBE)模式,对计算机编程开发类课程实践实施教学改革。通过系统构建课程目标体系、优化学习成果评估机制、创新项目化教学方案、完善多元评价系统等核心环节,深入比较分析了传统教学与成果导向教育模式的教学成效差异。结合教学数据,验证了成果导向教育模式在计算机编程开发类课程实践质量提升中的优势。成果导向教育模式能够有效调动学生学习积极性,明显增强编程实践水平、工程思维素养和团队协作效能,为培育满足产业需求的应用型人才构建了创新性教学模式。

关键词: 成果导向;程序设计;课程设计;教学改革

引言

在数字化时代快速发展的今天,计算机编程开发能力已成为计算机专业学生必备的核心技能之一。计算机编程开发类课程作为培养学生编程能力的重要课程,其教学效果直接影响着人才培养质量。然而,现有的教学模式往往过于侧重理论知识的传授,实践环节相对薄弱,教学内容与行业实际需求存在一定差距,难以有效满足学生个性化学习和实践能力培养的要求。为此,引入成果导向教育模式对计算机编程开发类课程进行改革,为提升教学质量提供了新的思路和解决方案。

一、OBE在计算机编程开发类课程中的应用

1. 实践过程反向设计

(1) 分层教学目标体系设计

课程构建了阶梯式能力培养框架:基础维度聚焦编程语言核心要素(语法结构、数据类型、程序控制等)的系统掌握;进阶维度突出算法实现与数据结构应用的优化能力;工程维度强化项目全生命周期管理能力;发展维度着重培养创新意识、问题分析与团队协作等综合素质。各层级目标均配套可量化的达成标准,确保教学目标的层次性与可测量性。

(2) 多维评估体系构建

课程创新性地建立了“过程-产出-素养”三维评价机制:过程维度采用代码质量分析、开发文档评审等

动态跟踪方式;产出维度实施项目成果展示、系统功能演示等终结性评价;素养维度开展编程竞赛、算法优化等专项能力测评。同时整合教师评价、企业评估与同伴互评等多元主体,形成立体化的质量监控体系,客观反映学生的综合能力发展水平。

(3) 项目导向教学模式创新

教学实施采用“真实项目+任务链”的工程化设计思路,实现知识点的项目化整合。通过递进式项目序列(如C语言的“图书馆管理系统”开发),结合案例研讨、代码走查、编程竞赛等多样化形式,依托在线编程社区的扩展资源,打造产学研融合的学习环境。重点引入企业实际工程案例,通过真实场景模拟提升学生的工程实践能力。

2. 学生为主导

(1) 差异化教学实施

基于OBE的计算机编程开发类课程采用分层教学模式,根据学生前期能力测评结果实施差异化教学。对于基础薄弱的学生,提供基础语法训练和模块化编程任务;针对中等水平学生设计算法优化和功能实现项目;为能力突出的学生安排系统架构设计和性能调优等挑战性任务。通过建立动态学习档案,实时跟踪每位学生的进步情况,及时调整教学策略。

(2) 项目小组协作机制

课程采用“项目经理制”的小组协作模式,每个项目小组5-6人,明确角色分工(如需求分析员、架构师、开发工程师等)。小组需完成从需求分析到测试部署的完整开发流程,定期进行代码评审和迭代演示。教师作为项目顾问,主要提供方向性指导,将决策权交给学生,培养其自主管理能力。通过Git等协作工具的使用,强化

基金项目: 本文系:广西职业教育教学改革研究项目(GXGZJG2025B253)+《基于“教学工厂”理念的专本衔接计算机编程开发类课程实践教学体系创新研究》

学生的工程实践素养。

(3) 个性化成长支持体系

建立“导师-助教-学长”三级指导网络，为不同层次学生提供针对性支持。设置弹性项目难度系数，允许学生根据自身情况调整任务复杂度。开发在线学习支持平台，提供分层教学视频、调试案例库等资源。实施“学习契约”制度，学生可自主选择感兴趣的项目方向，在教师指导下制定个性化学习计划，定期进行成果展示和反思总结。

3. 案例分析

(1) 案例教学与OBE的融合机制

在OBE框架下，案例教学采用“目标-案例-实践”的三阶段设计模式。教师首先根据预期学习成果精选行业真实案例，如电商系统、智能客服等典型应用场景。通过案例拆解演示，帮助学生建立从需求分析到系统实现的完整认知框架，将抽象的理论知识转化为具体的工程实践能力。案例库按难度分级，与课程的能力目标矩阵严格对应。

(2) 项目驱动的案例实施路径

学生以小组形式对案例进行深度重构，经历完整的软件开发生命周期。采用“案例模仿-功能扩展-自主创新”的递进式训练路径：初期参照案例实现基础功能，中期增加特色模块，后期自主设计创新应用。每个阶段设置里程碑评审，邀请企业工程师参与案例验收，确保项目成果达到行业实践标准。

(3) 多维度的案例教学评估

建立基于案例的多元评价体系：过程性评估关注需求分析、设计方案等文档质量；成果评估侧重系统功能完整性和代码规范性；能力评估通过答辩展示考察学生的工程表达能力。引入案例对比分析报告等创新考核形式，要求学生将课堂案例与开源项目进行横向比较，培养其技术评估和批判性思维能力。

4. 教学策略

(1) 能力导向的教学策略设计

基于OBE的课程设计聚焦学生实际能力产出，采用“任务驱动+项目引领”的复合教学模式。通过设计阶梯式实践项目序列，将编程语言基础、算法设计与软件工程知识有机整合，使学生在完成具体开发任务的过程中自然掌握核心技能。教师角色转变为学习引导者，主要提供脚手架支持，鼓励学生通过自主探索构建知识体系。

(2) 案例驱动的项目实施路径

采用“模仿-改进-创新”三阶段项目开发模式，

组织学生团队完成案例项目的全流程开发。在基础阶段复现案例核心功能，在提升阶段开发特色功能组件，在创新阶段实现个性化功能拓展。各开发阶段设置关键节点评审，引入企业技术专家参与项目验收，确保开发成果符合工业级质量标准。

(3) 案例教学的多维评价体系

构建“过程-质量-能力”三维评价框架：开发过程评估重点考察需求文档和设计方案的规范性；项目质量评估聚焦系统功能完备性和代码质量指标；综合能力评估通过技术答辩检验工程表达能力。创新性地采用案例对比分析法，要求学生将教学案例与业界同类项目进行技术对比，培养其技术鉴别和系统分析能力。

二、教学实施过程

1. 学习产出的系统化定义

基于OBE的计算机编程开发类课程采用“项目载体-能力矩阵”的产出定义方法。以“通讯录管理系统”为例，该项目被分解为四个能力培养维度：技术能力维度要求掌握文件读写、数据结构应用等核心技能；工程能力维度训练需求分析、模块设计等开发流程；创新能力维度鼓励扩展通讯录智能查询等特色功能；职业素养维度培养代码规范、文档编写等工程习惯。每个维度设置基础、提高、创新三级达成标准，学生可根据自身基础选择适合的挑战层级。同时，项目要求融入当前行业主流技术栈，如使用Git进行版本控制、采用单元测试保证代码质量等，确保学习产出与业界需求同步。教师通过个性化学习合约，允许学生在统一项目框架下自定义功能扩展方向，既保证基础能力达标，又尊重个体发展差异。

2. 教学活动的工程化设计

课程采用“理论精讲-案例示范-项目实战”三阶递进的教学活动设计。在理论环节，聚焦程序设计核心概念的精讲，如通过“联系人结构体设计”讲解内存管理原理；案例环节展示优秀开源通讯录项目的架构设计；项目实战阶段实施“做学合一”的工程训练。教学活动特别强调企业级开发规范的导入，包括：使用Visual Studio进行调试技巧训练、实施代码评审提升软件质量、应用CMMI简化模型管理开发流程等。采用角色扮演法，学生轮流担任项目经理、开发工程师、测试工程师等职位，体验完整软件生命周期。课内重点突破关键技术难点，课外通过在线编程平台补充训练，形成“精讲多练”的教学节奏。同时引入企业真实案例，如将通讯录与数据库连接等进阶需求，保持教学内容的前沿性。

3. 成果验证的多元化实施

构建“过程监控-阶段考核-综合评定”三维学习成果评价体系。过程监控维度通过代码提交质量、缺陷修复效率等动态指标实时追踪学习进展；阶段考核维度实施周期性成果展示，按开发迭代节点进行功能验收；综合评定维度开展毕业项目答辩，引入行业专家参与评审。评价标准严格参照企业技术规范，设置功能完备度（35%）、代码规范性（30%）、技术创新性（20%）、团队贡献度（15%）等核心指标。创新设计“能力发展图谱”，直观展示学生在算法实现、工程素养、创新意识等维度的成长曲线。通过对比入学能力基线评估与毕业成果展示，完整记录学生的能力发展路径。

4. 实施教学评价

（1）多维度的作品质量评估体系

课程采用量化评分机制对软件作品进行全面评估，包含五个核心维度：功能实现（30%）考核需求覆盖率与正确性；性能表现（20%）测试系统响应速度与资源占用；用户体验（20%）评估界面友好度与操作流畅性；代码质量（20%）通过静态分析工具检测规范性；运行效率（10%）进行压力测试与算法复杂度分析。每个维度设置分级评分标准，并引入自动化测试工具辅助评估。

（2）过程与结果并重的综合考评方案

课程考核采用“4:6”的加权模式，平时成绩侧重学习过程（课堂项目80%+考勤20%），课程设计注重成果产出。答辩环节设置多维评价：项目计划书（15%）、系统功能（30%）、项目展示（20%）、个人表现（25%）、学术诚信（10%），实行“一票否决”制，对抄袭等学术不端行为从严扣分。通过动态成长档案记录学生全周期表现，实现发展性评价。

三、教学效果对比

1. 个性化能力培养成效

OBE模式下的计算机编程开发类课程实践教学通过差异化教学策略显著提升了学生的实践能力。根据学习数据分析，采用分层项目任务后，基础薄弱学生的代码通过率提升42%，优秀学生完成拓展项目的比例达78%。通过建立个人能力发展档案，教师可精准把握每位学生的成长轨迹，提供个性化指导。项目答辩显示，92%的学生能够将数据结构知识灵活应用于实际问题解决，团队项目的代码复用率较传统教学模式提高35%。

2. 工程实践能力培养成效

课程实施“真实项目+模拟开发”的实践教学模式，有效提升了学生的工程实践水平。教学数据显示，实践敏捷开发流程的团队需求响应效率提升显著，采用版本控制系统的代码管理质量明显改善。持续集成技术的应用使项目构建稳定性大幅提高，最终有较高比例的学生项目方案获得企业技术认可，教学效果较传统模式优势明显。

3. 学习成效可视化分析

创新构建了能力发展评估体系，采用可视化图表动态呈现学生成长轨迹。教学实践表明，经过系统性训练，学生的代码规范性、测试覆盖率等关键指标均有显著提升。校企联合评估结果证实，改革班级在工程化能力、问题排查效率等职业素养维度表现突出，充分体现了教学改革的成效。

结论

本研究表明，基于成果导向教育的计算机编程开发类课程实践教学改革成效显著。该教学模式通过明确学习产出目标、实施项目化教学实践、构建多元化评估机制，有效提升了学生的编程实践水平和工程素养。实践表明，相较于传统教学方法，基于成果导向教育的模式更能调动学生学习积极性，强化其解决复杂工程问题的综合能力，为培养满足产业需求的软件工程技术人才探索出了一条有效路径。

参考文献

- [1] 杨瑞, 阎敏, 王萍, 等. 基于成果导向教育理念的循环管理体系在临床药师培训中的应用[J]. 中南药学, 2025, 23(10): 3097-3101.
- [2] 尹雪峰. 基于成果导向的针织成形服装综合实践课程改革创新改革实践[J]. 服装设计师, 2025, (10): 138-142.
- [3] 李宁宁. 成果导向教育理念下的设计制图课程教学新思路[J]. 瑞丽家居设计, 2025, (04): 165-167.
- [4] 李凡, 吕嘉, 成丽君, 等. 基于OBE理念的“Python程序设计”教学模式研究[J]. 科技资讯, 2024, 22(15): 207-210.
- [5] 杜少波, 李静. 融入OBE+课程思政的Java程序设计课程教学研究[J]. 高教学刊, 2024, 10(15): 116-119.