

# AIGC在《Java程序设计》课程教学中的应用探究

徐栋

常州刘国钧高等职业技术学校

DOI:10.12238/mef.v8i15.16148

**[摘要]** 本文针对传统教学存在的概念抽象、调试困难等痛点,探讨AIGC技术在《Java程序设计》课程教学中的应用,AIGC可为教师提供课程设计与资源生成支持,为学生提供个性化辅导与答疑,并实现过程化教学评估。研究表明,AIGC能有效提升教学质量,构建“人机协同”的教学新生态。

**[关键词]** AIGC; 程序设计; 课程教学

**中图分类号:** G642.3 **文献标识码:** A

## Exploration of the Application of AIGC in the Teaching of "Java Programming"

Dong Xu

Liu Guojun Higher Vocational and Technical School, Changzhou

**[Abstract]** This paper examines the limitations inherent in traditional teaching methodologies—such as the abstraction of core programming concepts and challenges in code debugging—and investigates the integration of Artificial Intelligence Generated Content (AIGC) technology into the instruction of "Java Programming." AIGC serves as a multifaceted tool that supports educators in curriculum design and instructional resource development, provides students with personalized learning assistance and real-time query resolution, and enables continuous, process-oriented assessment. Empirical findings indicate that the incorporation of AIGC significantly improves teaching effectiveness and fosters the development of a novel "human-machine collaborative" pedagogical ecosystem.

**[Key words]** AIGC; Programming; Course Instruction

### 1 引言：研究背景与意义

#### 1.1 《Java程序设计》课程重要性

《Java程序设计》作为高等院校计算机科学、软件工程、物联网工程及相关信息技术专业的核心必修课程,其重要性主要体现在三个方面。首先,在知识体系方面,Java语言具备纯粹的面向对象特性、健壮的语法结构以及“一次编写,到处运行”的跨平台能力,是学生理解面向对象编程思想、掌握现代软件开发范式的理想工具。学生通过类与对象、继承、多态、接口、异常处理等核心概念的学习,能够建立起扎实的编程基础。其次,在应用生态方面,Java在企业级应用、安卓移动开发、大数据框架和分布式系统等领域具有不可动摇的地位,掌握Java相当于获得了进入多个主流技术领域的“通行证”。最后,在能力培养方面,本课程着重训练学生的计算思维、逻辑分析能力、系统设计能力以及解决复杂工程问题的实践能力,这些能力是构成信息技术人才核心竞争力的关键要素。正因如此,《Java程序设计》的教学质量直接影响到专业人才的培养水平。

#### 1.2 《Java程序设计》课程教学现实痛点

尽管《Java程序设计》课程在专业培养中具有重要地位,

但其传统教学模式在实践中长期面临多重挑战:首先,面向对象编程概念较为抽象,初学者对“类与实例的关系”“多态的运行机制”“接口与抽象类的区别”等核心概念难以通过课本和教师的静态讲解形成直观理解;其次,编程练习中调试过程容易带来挫败感,学生在遇到编译错误或逻辑错误时往往难以自主定位问题,而有限的课堂时间与教师资源难以以为所有学生提供即时支持;再者,教学内容偏重语法知识灌输,与实际项目场景脱节,导致学生虽能记忆语法规则,却缺乏开发完整应用程序的能力;此外,学生编程基础与学习能力差异显著,统一的教学进度难以兼顾不同层次学生的需求;最后,教学案例、练习题与项目任务的设计更新需要教师投入大量精力,手工准备效率较低,且难以持续保证资源的多样性、时效性与趣味性。

#### 1.3 AIGC给《Java程序设计》课程教学带来的契机

以ChatGPT、Claude等为代表的AIGC技术,特别是大型语言模型,凭借其在自然语言理解与代码生成方面的卓越能力,为解决上述教学痛点带来了革命性契机。AIGC不仅是一种工具,更代表了一种全新的教学赋能范式。它能够准确理解师生用自然语言表达的指令,并生成高质量的代码、解释说明、教学案例与评

估反馈。具体而言:面对抽象概念理解困难,AIGC可提供多角度、生活化的类比解析,将抽象转化为具体认知;针对调试过程中的困境,它能作为“永不疲倦”的编程助手,实时提供精准的错误诊断与修改建议;对于理论与实践脱节的问题,AIGC可快速生成贴近实际应用的项目场景、代码框架及技术方案,有效连接理论知识与工程实践;针对个性化教学不足,它能为每位学生提供一对一、按需定制的学习支持,切实落实因材施教;面对教学资源更新压力,AIGC可成为教师的“超级助理”,大幅提升备课效率与资源质量。因此,系统研究AIGC在《Java程序设计》课程教学中的应用,既是顺应技术发展的必然要求,也是提升教学质量、培养创新型软件工程人才的迫切需要,具有重要的理论价值与实践意义。

## 2 AIGC在《Java程序设计》课程教学中的具体应用

### 2.1 教师端:作为全方位的课程设计与课堂管理助手

在《Java程序设计》课程教学中,AIGC为教师提供了全方位的教学支持,显著提升了教学效率与质量。在课程设计环节,教师可借助AIGC快速生成丰富的教学资源。例如,输入“请设计一个包含继承、多态等面向对象特性的Java教学案例,要求包含完整的类结构和详细的代码注释”,AIGC即可生成结构清晰、注释详实的案例,这些案例既可直接用于教学,也可按需调整优化,极大减轻了备课负担。AIGC还能根据教学进度和学生基础,自动生成分层练习题和项目任务。教师只需提出基本要求,如“针对Java异常处理设计5道难度递进的编程题,并附参考解答和解析”,AIGC便能快速输出相应习题集,助力构建系统化的练习体系。在课堂教学中,AIGC可作为教师的实时辅助工具,用于代码演示,如比较不同排序算法的效率差异,或展示代码重构过程。通过指令“请展示冒泡排序与快速排序在相同数据集上的性能对比,并提供可视化结果”,AIGC能生成直观的对代码与性能分析,使抽象概念具体可感。此外,它还能帮助设计互动环节,如生成含典型错误的代码供学生调试,或创设模拟面试场景以辅助就业准备。在课程评估方面,AIGC可协助教师制定科学评价标准,自动生成评分细则,并提供个性化教学反馈建议。通过分析学生学习数据,AIGC还能识别教学中的薄弱环节,支持教师及时调整策略,实现教学过程的持续优化。综上所述,AIGC使教师得以将更多精力聚焦于教学设计与学生指导,从而显著提升教学质量。

### 2.2 学生端:作为个性化编程伙伴与项目顾问

AIGC作为学生的个性化学习伙伴,为《Java程序设计》课程提供了多维度、全天候支持。在概念理解阶段,学生可随时向AIGC提问并获得个性化解答。例如,当学生对“Java接口与抽象类的区别”产生困惑时,AIGC不仅能提供通俗易懂的阐释,还会辅以具体代码示例,帮助学生从多角度理解抽象概念。这种即时、精准的答疑方式,有效弥补了传统课堂统一教学的局限性。在编程实践环节,AIGC的价值更为突出。学生遇到编译错误或逻辑错误时,只需将错误信息和相关代码提交给AIGC,即可获得快速的问题定位与详细的解决方案。以典型的

“NullPointerException”为例,AIGC不仅能准确指出错误成因,还会深入解释异常产生机制,并提供多种避免此类问题的最佳实践方案。在项目开发阶段,AIGC可承担项目顾问的职责。从项目构思、技术选型到具体实现,AIGC都能提供专业建议。当学生提出“为图书馆管理系统设计技术架构,并给出核心模块的代码框架”等需求时,AIGC能够提供包含类设计、数据库设计及关键算法实现的完整解决方案。此外,AIGC还能基于学生的学习进度和知识掌握程度,智能推荐适合的学习资源与练习题,实现真正的个性化学习路径规划。通过持续记录和分析学习数据,AIGC能够识别学生的知识薄弱环节,提供针对性的强化训练,助力学生构建完整的知识体系。这种智能化的学习陪伴,不仅显著提升了学习效率,更有助于培养学生独立解决问题的能力。

### 2.3 评估端:作为过程化与自动化的评估辅助工具

AIGC在教学评估中的应用正推动《Java程序设计》课程实现从结果导向到过程导向的评估转型。在作业批改方面,AIGC能够协助教师完成基础性评估工作:通过分析学生提交的代码,自动检查代码规范、语法正确性与基础逻辑,并生成个性化改进建议。例如,针对一段Java代码,AIGC可准确指出“变量命名不规范、缺少必要异常处理、算法效率需优化”等问题,同时提供具体修改方案。这种即时、精准的反馈机制有效帮助学生及时发现并修正问题,显著提升了学习成效。在过程性评价层面,AIGC展现出更显著的优势。通过集成至开发环境,AIGC可持续追踪学生的编程过程,完整记录代码修改历史、调试轨迹及问题解决路径。这些过程性数据为教师评估学生的学习态度、问题解决能力与编程思维提供了客观依据。教师可通过指令“分析学生本学期项目中的代码演进过程,评估其面向对象设计能力的提升情况”,获取包含设计模式运用、代码重构改进等维度的详细分析报告。在项目评估中,AIGC能协助教师构建多维度评价体系。除代码质量外,还可评估文档完整性、测试覆盖率、架构设计合理性等指标。借助“基于Java项目最佳实践,制定包含代码质量、架构设计、文档规范等维度的评分标准”等指令,AIGC能够提供专业评估框架。此外,通过对比学生不同阶段的作品,AIGC还能协助教师进行学习成效分析,为教学改进提供数据支撑。这种智能化的评估体系不仅有效减轻了教师工作负担,更重要的是实现了对学习过程的全面、客观评价,推动教学评价体系向科学化方向发展。

## 3 AIGC在《Java程序设计》课程教学中的应用成效

将AIGC深度融入《Java程序设计》课程教学,能够从以下四个方面产生显著的应用成效:

### 3.1 显著降低学习门槛,有效增强学习信心与兴趣

AIGC通过提供即时、精准的答疑与调试支持,如同一位随时在线的私人导师,能迅速帮助学生扫除学习障碍。在传统教学模式中,学生遇到问题时往往需要等待教师解答,这种等待不仅会中断学习进程,还会逐渐积累挫败感。相比之下,AIGC具备24小时不间断的服务能力,确保学生在问题产生之初就能获得及时帮助。这种即时反馈机制使编程初学者能够更快地获得正向激

励,从而从“害怕出错”逐步转变为“敢于试错”。实践表明,在AIGC辅助学习的环境下,学生的代码尝试次数显著增加,探索复杂编程概念的积极性也明显提升。这种学习态度的积极转变对编程入门阶段的培养尤为关键,为后续深入学习奠定了坚实的心理基础。

### 3.2 深化对抽象编程概念与设计思想的理解

AIGC能够针对学生的不同认知水平,提供多样化且情景化的概念阐释。在传统教学模式下,教师通常以统一的方式讲解抽象概念,难以适应不同认知风格学生的学习需求。而AIGC则能通过生成具体代码示例、绘制UML图示、提供生活化类比等多种途径,帮助学生在不同情境中理解同一概念。以多态概念的讲解为例,AIGC既可呈现“动物发出叫声”这类经典案例,也能结合GUI编程中的事件处理等实际应用场景展开说明。这种多角度、多层次的解释方式,使学生能够从最适合自己的认知入口掌握知识,构建起更为立体和稳固的知识体系,最终实现对面向对象编程思想、设计模式等深层知识的融会贯通。

### 3.3 提升编程效率与工程化素养,对接产业实践

AIGC能够自动完成大量重复性、模式化的编码工作,包括生成Getter/Setter方法、数据库链接代码、单元测试用例等任务。这使得学生可以将有限的认知资源投入到更核心的系统架构设计、算法实现和复杂问题求解中。在传统教学模式下,学生往往需要耗费大量时间编写基础代码,难以真正体验完整的软件开发流程。而通过AIGC的辅助,学生能在相同时间内完成更具复杂度的项目,并接触到更贴近工业界的开发场景。同时,通过观察和学习AIGC生成的规范代码,学生能够在潜移默化中掌握良好的编码习惯与工程规范。这种与产业实践紧密结合的学习体验,显著提升了学生的工程素养和就业竞争力。

### 3.4 真正实现规模化教育与个性化培养的有机统一

AIGC有效化解了传统教学中“优生吃不饱、差生跟不上”的矛盾,显著提升了整体教学效果。这种突破源于其对传统课堂时空局限的超越,能够为每位学生提供真正个性化的学习支持。在传统大班授课模式下,教师受限于教学资源,往往采用“一刀

切”的教学方式,难以顾及学生的个体差异。而AIGC则能针对不同学生的具体问题提供精准指导:对学有余力的学生,引导其探索微服务架构、响应式编程等前沿技术;对基础薄弱的学生,则提供基础知识的系统讲解和充分练习。这种因材施教的支持模式,确保了不同层次的学生都能在现有基础上获得最大程度的提升。

## 4 结论

综上所述,AIGC技术在《Java程序设计》课程教学中的应用代表着一次深刻而积极的教学范式革新。通过赋能教师教学、变革学生学习方式和革新教学评估体系,AIGC全方位提升了教学效率与质量。实践表明,其在减轻学习挫败感、深化概念理解、增强工程实践能力及实现个性化教学等方面均取得了显著成效。然而,我们需要清醒认识到,AIGC本质上是一种强大的“增效工具”而非“替代品”。它既不能取代教师在情感交流、价值引导和人格塑造方面的独特作用,也无法替代学生通过自主思考和刻苦实践所获得的思维训练与能力提升。毕竟,AIGC生成的代码与解决方案,最终仍需由具有主观能动性的师生来理解、验证、批判和实际应用。

因此,推进《Java程序设计》课程教学未来发展的关键,在于构建一个以教师为主导、学生为主体、AIGC为赋能工具的“人机协同”新型教学生态。这一生态将充分发挥各方优势,实现教学效果的最优化。

### [参考文献]

- [1]韩璇,袁中果,王宏.生成式人工智能在教育领域的应用与展望[J].中国电化教育,2023(5):71-78.
- [2]徐晓飞,徐晓东.人工智能赋能教育:智能教育的框架与路径[J].现代教育技术,2022,32(10):5-14.
- [3]王颖,李艳,张欣.智能时代编程教育模式的变革与路径探索——基于AIGC的视角[J].远程教育杂志,2024,42(1):12-21.

### 作者简介:

徐栋(1981—),男,汉族,江苏常州人,硕士,讲师,计算机科学与技术。