

程序设计类数字资源建设与校本化实施的协同机制研究*

郑申海 张晓霞

重庆邮电大学计算机科学与技术学院, 重庆 400065

摘 要 在大语言模型快速进入高等教育场景的背景下, 程序设计类课程的数字教学资源建设正在从静态材料供给转向智能资源服务, 缺少一个能够将“资源建设”与“校本化实施”的整体框架。本文以程序设计类数字教学资源建设与校本化实施为研究对象, 提出从资源仓库到课程智能体的转型方式, 构建内容资源、模型能力、教学实施、育人价值、治理评价等五维协同机制, 提出面向程序设计类课程的校本化实施路径。方法的研究既吸收了已有实践中的可行经验。

关键词 程序设计, 数字资源, 校本化, 五维协同

From Resource Repository to Course Agent: A Five-Dimensional Collaborative Mechanism for the Construction of Digital Teaching Resources and School-Based Implementation of Programming Courses*

Shenhai Zheng Xiaoxia Zhang

School of Computer Science and Technology
Chongqing University of Posts and Telecommunications,
Chongqing 400065, China;

Abstract—Against the backdrop of the rapid integration of large language models into higher education, the development of digital teaching resources for programming courses is shifting from “static material provision” to “intelligent resource services”. However, there remains a lack of an overall framework that can integrate “resource construction” with “school-based implementation”. This paper focuses on the construction of digital teaching resources for programming courses and their school-based implementation. It proposes a transition model from resource repositories to course agents, establishes a five-dimensional collaborative mechanism involving content resources, model capabilities, teaching implementation, educational value, and governance evaluation, and outlines a school-based implementation pathway for programming courses. The methodology incorporates feasible insights from existing practical experiences.

Key words—programming, digital resources, school-based implementation, five-dimensional collaboration education

1 引 言

程序设计类课程是高校理工科和计算机类专业最基础、也最关键的课程类型之一。无论是C语言、C++程序设计, 还是后续的数据结构、操作系统、嵌入式开发和系统软件课程, 其学习质量都高度依赖程序设计基础课程的教学质量[1][2]。近年来, 在教育数字化转型与生成式人工智能迅速发展的双重推动下, 程序设计类课程的教学场景正在发生深刻变化[3][4]。学生学习资源的获取方式, 已从过去的教材、课堂、作业逐步转向平台、资源库、智能体、多模态互动;

教师的教学组织方式, 也从单一讲授转向混合式、数据化、诊断式和协同式; 而课程资源本身, 则正在从静态课件、习题和实验手册, 转向能够即时生成、持续迭代、智能反馈的资源系统[5]。

程序设计类课程长期存在几类突出问题。其一, 学生基础差异明显, 统一进度与统一资源难以适应不同学生的学习需求。李杰指出, C语言课程中基础差异、实践薄弱、反馈滞后和评价单一问题交织, 导致一刀切教学难以实现精准支持[6]。其二, 通用大模型虽有较强的代码生成能力, 但与校本课程大纲、校内题库、专业案例之间往往存在脱节。杨晓贤等明确指出, 通用生成式AI难以与本校习题和课程目标紧密匹配[7]。其三, 教学资源更新维护成本高, 教案、课件、题库、项目和在线答疑需要不断修订, 但靠纯人工维护难以持续。冷祥光等正是基于这一现实困境,

***基金资助:** 本文得到重庆市高等教育教学改革研究项目(基金编号: 234050)和重庆邮电大学教学改革研究重点项目(基金编号: XJG23112)。

***通讯作者:** 郑申海 zhengsh@cqupt.edu.cn。

提出用大模型辅助教案、项目、思政案例和答疑资源建设[8]。其四,大模型还带来新的教学风险,如学生过度依赖生成答案、实验任务被替代、思辨能力被削弱、学术诚信受到挑战。穆玲玲与许岩分别从实验教学和课程思政两个方向,对此提出警醒[9][10]。最后,程序设计课程内容本身也需要升级,不仅要教语法和算法,还要纳入代码安全、工程伦理、责任意识等内容,尹良泽等关于代码安全知识图谱的研究就说明了这一趋势[11]。

综上,在大模型时代,程序设计类数字教学资源如何围绕本校培养方案、课程体系、学生画像、教师共同体与评价制度,将数字教学资源才从资源仓库转变为课程智能体具有重要的意义。

2 核心概念与理论框架

2.1 数字教学资源的内涵

在传统理解中,数字教学资源主要指教学视频、电子讲义、PPT、题库、案例库、实验指导书和在线作业等内容性材料。这种理解在平台化教学阶段具有合理性,但在大模型时代已显不足。因为资源已经不再只是可供访问的材料,而越来越成为可参与教学过程智能对象[5]。程序设计类数字教学资源应被界定为:围绕课程目标而构建、能够在知识供给、任务生成、过程诊断、智能反馈、价值引导和评价支持中发挥作用的数字化与智能化资源体系。

2.2 校本化实施

校本化包含目标校本化、内容校本化、学情校本化和制度校本化四层含义[8]。目标校本化是资源必须围绕本校培养方案、课程目标和毕业要求来配置。内容校本化是资源应吸纳本校历年试卷、作业、题库、实验项目、专业案例和常见问题,形成具有本校特色的知识库。学情校本化是资源应依据本校学生基础、专业背景、学习行为和问题分布,形成分层、差异化支持[12][13]。制度校本化是资源必须与本校平台环境、课程组织方式、考核制度和教师团队能力相匹配。

基于上述概念,本文提出从资源仓库到课程智能体的转型逻辑。资源仓库强调的是存储与供给;课程智能体强调的是协同与服务。资源仓库阶段的资源,主要解决有没有;课程智能体阶段的资源,则更关注能不能在教学中动态发挥作用[5]。前者是静态的、分散的、被动调用的;后者是动态的、联动的、可诊断与可反馈的。

在此基础上,本文构建“五维协同机制”,如图1所示。第一是内容资源,即围绕知识点、案例、题库、实验、安全与思政形成多层资源谱系[11][14][15]。第二是模型能力,即通过校本知识库、

微调模型、RAG 与学习分析,让资源具备生成、匹配、答疑和诊断能力[7][5]。第三是教学实施,即通过课前、课中、课后与实验、考核等环节实现资源运行。第四是育人价值,即把代码安全、责任意识、工程伦理、课程思政融入资源设计[14][15]。第五是治理评价,即建立资源校验、学术诚信、评价证据和质量反馈机制[9][10]。只有这五个维度同时联动,程序设计类数字教学资源建设才可能真正完成校本化实施。

3 程序设计类数字教学资源建设的现实困境

尽管大量高校已开展数字资源建设,但从整体上看,程序设计类课程仍面临明显的结构性困境首先是资源供给碎片化与静态化。许多课程已经拥有教材、PPT、录播视频、在线平台题目、实验指导、答疑群等多种资源,但这些资源往往各自分散在不同平台、不同教师手中,缺乏统一的结构组织与动态调用机制。学生获取资源时常常处于到处找资料的状态,教师也难以基于同一资源体系进行持续迭代和统一反馈。

其次是资源更新高成本与低复用。程序设计类课程不同于许多静态理论课程,它对案例、题库、平台工具、实验环境和项目任务的时效性要求较高。工具链变化、题库迭代、学生错误模式变化,都会要求资源持续更新。冷祥光等明确指出,教案设计、PPT 制作、实践项目、思政案例与在线答疑都需要耗费大量人力物力。若仅依赖教师人工维护,不仅效率低,而且难以形成高频迭代和多版本适配。传统资源建设容易陷入建得起、维护不起的困局。

第三是通用模型与校本课程脱节。通用大模型在代码生成和问答上的确展现出强大能力,但杨晓贤等已经指出,通用模型难以将回答与本校教学大纲、具体习题、历年考试风格、专业背景和教师教学重点紧密结合。程序设计课程尤其强调学会按照具体要求写出可通过测试、符合规范、体现思路的程序,这种要求并非通用知识所能完全覆盖。因此,只使用通用模型,可能带来答案似乎正确但不贴课程、代码能运行但不合要求的问题。

第四是实验任务短链化与标准答案化。在大模型出现之前,很多实验任务本就偏向验证性、短路径和结果导向;在大模型出现之后,这类任务更容易被一键生成答案。穆玲玲指出,若任务只是让学生输出一个正确结果,大模型很容易直接完成,学生则可能失去必要的技能训练机会。也就是说,数字教学资源若仍按照旧逻辑组织实验材料,就会在新技术环境下迅速失效。资源建设若不能同步转向长逻辑链、开放型和跨平台,就无法支持学生能力发展。

4 校本化数字教学资源建设的总体架构

围绕上述问题,程序设计类数字教学资源建设需要形成一个面向课程目标、教学过程与学习结果的一体化架构。这个架构的第一原则是以培养目标统摄资

源建设。资源建设不能以便于制作或平台支持什么为中心,而必须以课程目标为中心。程序设计类课程的目标至少应包括:掌握程序设计基本知识,形成问题分析与算法设计能力,具备规范编程与调试能力,发展系统思维与工程意识,并逐步形成安全意识、责任意识和创新意识。数字资源的分类与组织必须围绕这些目标展开。总体架构如图1所示。

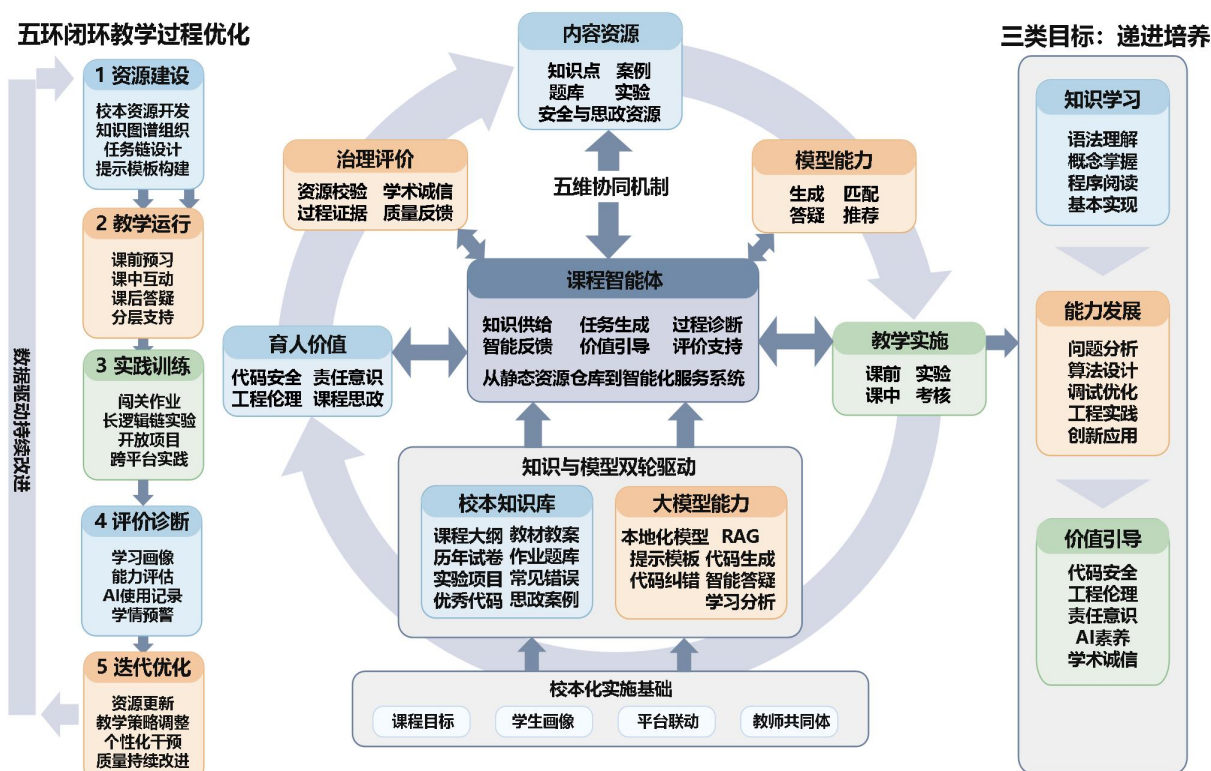


图1 程序设计类数字教学资源建设与校本化实施的总体架构

校本资源体系可分为五类。第一类是基础知识资源,包括微课、电子讲义、例题讲解、核心概念图、易错点提示、预习测验等。第二类是实践项目资源,包括在线作业题、程序改错题、功能实现题、综合设计题、实验任务、跨学科案例和开放项目。第三类是智能问答与诊断资源,包括校本知识库、FAQ、常见错误库、学习画像、代码纠错规则和反馈模板。第四类是价值引导资源,包括代码安全案例、思政案例、工程伦理素材、技术史与行业案例等。第五类是评价治理资源,包括评分量表、过程记录模板、项目互评规则、AI使用规范、学术诚信说明和风险校验流程。

从技术底座看,校本化实施不能缺少三层支持。第一层是校本知识库。它应汇集课程大纲、教材要点、教师教案、历年作业、历年试卷、平台题目、常见错误、优秀代码样例、实验指导以及专业特色案例。第二层是校本模型能力。通过定向微调可以显著提升模型对课程问题的理解和回答质量;而 RAG 可进一步增

强知识密集型任务中的匹配能力。第三层是学习分析能力。只有把预习时长、作业提交、课堂练习、考试得分、提问内容等过程数据整合起来,资源系统才可能从内容分发走向精准干预。也就是说,校本资源体系的真正价值,不是让学生多一个问答工具,而是让教师和学生都获得更强的课程运行支持。

在组织机制上,校本资源建设还需要课程组共同体支持。程序设计课程往往具有教师人数多、平行班多、学科基础性强的特点,因此资源建设不能依赖个别教师“单兵作战”。校本化实施必须形成课程统筹、教师共建、平台联动、学生反馈、持续迭代的组织结构。没有这种组织基础,再先进的模型和资源也很难长期稳定运行。

5 校本化实施的教学运行机制

程序设计类数字教学资源建设只有进入真实教学过程,才能体现其价值。因此,校本化实施的核心,不只是建资源,更是让资源在课程全过程中运行起来。

从教学运行逻辑看,可以构建课前、课中、课后、实验—考核五环联动机制。

5.1 课前基于校本资源的精准预习与分层导学

课前环节的核心,不是让学生被动看视频,而是让学生在进入课堂之前形成结构化预期。线上预习包括微课视频、电子讲义、编程案例演示与预习测验题,目的在于帮助学生提前熟悉课程内容、明确重难点。AI 助教可以根据课程大纲为学生生成预习材料,并依据不同水平提供快速入门或项目驱动的个性化路径。许岩强调,在课前环节,教师还应借助大模型整合准确、精炼、具有价值引导功能的预习资料,避免学生直接面对开放通用模型时出现反馈过多看不懂或内容误导难辨别的问题。由此可见,校本化课前资源至少应同时具备知识铺垫、水平识别、价值引导三方面的能力。

在实施上,可以建立“校本预习包”机制。每一课次的预习包由课程组统一生成,包括核心知识图、微课视频、2—3 个基础例题、1—2 个易错点警示、一个与生活或专业相关的小案例,以及一个短测。学生完成预习后,系统自动采集其学习轨迹、测验结果和提问内容,并将这些信息反馈给教师。这样一来,教师就不再是凭经验估计学生掌握情况,而是可以在课前就看到某一知识点的理解障碍、某一类型学生的困惑分布。李杰关于个性化分层诊断的研究表明,这样的过程数据可以支撑知识点掌握、编程能力和学习习惯三个维度的画像构建。校本化预习资源因此不只是提前学,而是课堂精准化的前提。

5.2 课中教师主导下的人机协同与差异化教学

程序设计课程的课堂不应因为大模型的加入而削弱教师角色,相反,教师角色会更加关键。杨晓贤提出的“AI+教师”双轨制恰恰说明,AI 助教的真正价值,在于为教师提供个性化学习顾问、即时互动伙伴、智能编程助手和作业智审助手的支持,而课堂的方向感、重点突破与价值引导仍然必须由教师掌握。许岩也强调,教师在大模型时代的核心任务,是引导学生正确、高效地利用模型,而不是让模型取代教学。

在这种意义上,课中实施应当是教师主导下的人机协同。具体而言,教师根据课前画像结果确定本次课的重难点,把更多时间投入到共性问题讲解、关键概念辨析、典型错误剖析和高阶问题引导上;AI 助教则承担基础解释、即时代码示例生成、分层练习推送和个别答疑支持。校本资源在课堂上的最佳运行形态,不是教师讲完、学生再练,而是资源与教学同步驱动。

更重要的是,课中差异化教学必须建立在资源的可分层供给之上。分这说明,同一课堂中应存在基础题、提高题、拓展题和项目任务的同步供给。所谓校本化实施,并不是每所学校都追求同一种分层比例,而是根据自身学生特点确定哪些学生需要更多提示、哪些学生需要更强挑战、哪些内容必须由教师重点兜底。资源的分层能力,直接决定差异化教学能否真正落地。

5.3 课后智能答疑、作业反馈与能力延伸

程序设计课程的学习真正发生在课后。学生在作业中遇到的语法错误、逻辑漏洞、测试失败和调试困境,往往是能力提升的关键节点。大模型 7×24 小时在线辅导答疑把学生问题平均解决时间从传统的 48 小时大幅缩短至 32.3 分钟,并通过问题接收、分析、解答和效果反馈的机制实现持续优化。AI 助教在课后可以进行代码作业自动诊断、个性化巩固练习推送和拓展资源推荐。冷祥光则从资源建设角度说明,论坛与 AI 机器人协同回答能够提升答疑效率和覆盖范围。

由此可见,校本化课后资源至少应形成三层支持。第一层是即时性答疑,围绕课程知识库和常见错误库,为学生提供快速反馈。第二层是作业后分析,不仅告知对错,还指出错误原因、相关知识点与改进建议。第三层是能力延伸资源,根据学生表现推送补救材料、专题训练或拓展项目。这里的关键不在于给出答案,而在于引导学生自己修正。

6 实验从标准答案任务走向长逻辑链开放任务

如果说课前、课中、课后主要解决知识与练习问题,那么实验环节则是程序设计课程能否真正培养系统能力、工程能力与创新能力的关键。穆玲玲的研究提供了一个非常重要的启发:在大模型时代,实验教学的重点应从完成一个程序转向完成一条有逻辑、有分析、有平台比较和系统联系的任务链。这对程序设计类数字教学资源建设提出了新的要求,即实验资源不能只是一份任务说明,而必须是一组经过精心设计的任务结构。

第一,实验资源要强调跨知识点融合与跨平台实现。穆玲玲提出,应围绕相关课程知识点设计实验任务,并建立多课程知识图谱。对程序设计课程而言,这意味着一个优质实验不应只让学生完成某个语法点练习,而应使其理解程序语言、编译工具、操作系统、硬件平台与程序执行之间的关系。比如同样是输出字符串,在不同操作系统、不同 CPU 架构、不同工具链下实现,其教育价值远高于单平台单答案任务。对于 C/C++ 程序设计课程,同样可以设计“同一功能在命

令行环境、在线评测环境和可视化 IDE 环境中实现”的任务链。

第二,实验资源要强调长逻辑链设计。穆玲玲提出,通过增加实验任务完成链,可以有效防止大模型直接生成整套答案。比如求素数,不再只要求输出正确结果,而是要求用不同方法实现,并与编译器生成代码对比分析。对程序设计课程而言,这种思路非常值得推广。一个真正有质量的实验任务,可以设计为问题分析、算法选择、伪代码表达、编码实现、调试记录、复杂度分析、优化比较和反思总结八步链条。学生即使借助大模型,也必须经历多次判断、选择、修改与说明,才能完成任务。这样,资源建设的重点就从题目本身转向任务逻辑。

第三,实验资源要强调开放型与分级选择。穆玲玲的研究指出,多知识点长逻辑链开放任务并不适合强制所有学生全部完成,因此应设计不同难度、不同类型、不同分值的任务,允许学生根据自身情况或兴趣选择,达到规定分值即可。这一点对于程序设计类课程尤其重要。因为学生水平差异客观存在,若所有实验都按统一难度配置,要么压垮基础薄弱者,要么束缚基础较好者。李杰关于分层教学的研究、杨晓贤关于不同学习路径的设计,也都为此提供支持。未来的实验资源建设,不应再用一套实验手册适合所有人的想法来组织,而应形成必做任务、选做任务和挑战任务的层次化结构。

7 将课程思政与AI素养资源全过程融入

大模型进入程序设计课程后,课程思政的对象、方式和评价逻辑都发生了变化。许岩等的研究非常明确地指出,在“师一生一机”三元结构下,课程思政已不再是传统课堂中穿插几个案例、强调几句道理那么简单,而必须回应学生如何认识大模型、如何筛选信息、如何使用工具、如何保持独立思考等新问题。程序设计类数字教学资源建设若不主动回应这些问题,就会把大模型仅仅当作提效工具,而忽视其对学生认知方式和价值判断方式的深刻影响。

课前重辨别、课中重引导、课后重创新、考核多维度的路径,可为资源建设提供了清晰的实施逻辑。第一,在课前资源中,应融入价值辨识功能。教师不是放任学生直接去问开放模型,而是通过整合书籍、文章、视频和提示模板,为学生提供更精炼、准确、具有方向感的预习材料。第二,在课中资源中,应加入 AI 使用技巧与辨别训练。例如,让学生比较不同提示词得到的回答质量,识别代码示例中可能存在的逻辑问题,从而理解机器不能代替人。第三,在课后资源中,应设计探索创新类任务,如“AI 与学生代码对抗”“检测代码中的伦理风险”“开发具有社会价

值的小程序”等。第四,在评价资源中,应把责任意识、协作能力、信息辨别能力、AI 使用反思纳入考核证据。

这说明,程序设计课程中的课程思政资源,不应再以单独板块存在,而应嵌入到资源的结构层、过程层和评价层。对程序设计类课程来说,最自然的思政资源不是空泛说教,而是与工程实践紧密相关的真实案例。例如数据精度转换导致工程事故、算法偏差带来的社会后果、软件缺陷造成的公共安全问题、团队协作中的责任分担与诚信问题等。冷祥光利用阿里安5火箭事故生成思政案例,尹良泽利用软件缺陷事故构建安全意识,许岩利用 AI 使用场景培养辨别与创新精神,这些都说明程序设计课程的价值资源完全可以做到自然嵌入、真实触发、任务驱动。

8 平台联动

校本化实施不能建立在“换一个平台就能解决问题”的设想上,而要构建平台联动机制。在理想状态下,MOOC 或校内教学平台主要承担结构化内容供给与预习任务;PTA 或在线评测平台承担作业、考试与基础代码评测;头歌类实践平台承担项目式任务与闯关式训练;论坛、群组或课程智能体承担碎片化答疑、经验共享与共同体学习;而校本模型系统则贯穿其间,承担知识检索、资源生成、错误诊断、学习分析与教师支持功能。这样,数字资源不再是孤立存在于不同平台,而是围绕同一课程目标形成协同生态。学生看到的是一个连续学习空间,教师看到的是一个可追踪、可迭代的资源系统。

平台联动的校本化关键,在于统一标准与统一证据。各平台产生的数据和资源必须能够互相对接,至少要在知识点标签、题目难度、任务类型、评价指标和学习画像维度上做到一致。否则,平台越多,资源越容易进一步碎片化。只有将不同平台的行为数据纳入同一分析框架,资源系统才能真正支持差异化教学与过程评价。

9 实施成效

为检验程序设计类数字资源建设与校本化实施的实际效果,本文以本校《C 语言程序设计》课程为例,选取近四个学期的课程教学数据进行分析,评价指标包括选课人数、平均分、中位数、最高分、及格率、优秀率和评教得分等,相应结果如图 2 和图 3 所示。该图中,后 2 个学期是推行数字资源使用后的结果。

首先,从课程覆盖规模看,近四学期的选课人数分别为 789、1890、1786 和 2332 人。近一学期增幅较大。为了满足全校学生的程序课程学习,该资源建设与实施模式已不再局限于小规模试点,而是逐步进入多班级、大规模、常态化运行阶段。尤其是在学生规

模显著扩大的情况下，课程成绩并未出现下降，反而有所改善，表明校本化资源体系具有较好的推广适应性和教学支撑能力。

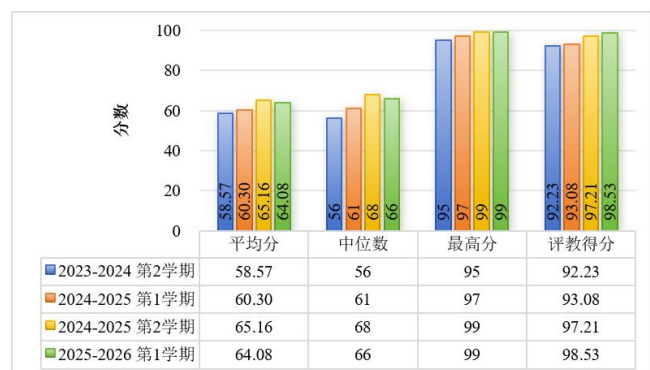


图2 《C语言程序设计》课程近四学期平均分、中位数、最高分和评教得分分析

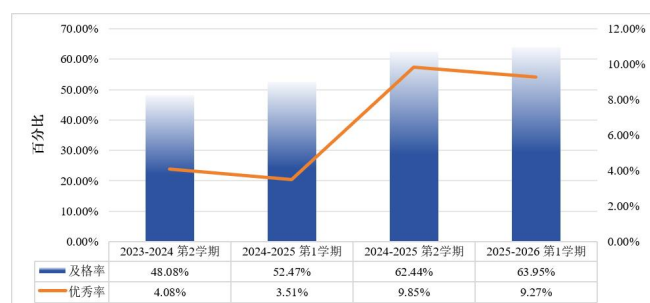


图3 《C语言程序设计》课程近四学期及格率和优秀率分析

其次，从学习结果看，学生平均分由 58.57 分提高至 64.08 分，提升 5.51 分；中位数由 56 分提高至 66 分，提升 10 分。提升幅度高于平均分，说明成绩提升并非仅由少数高分学生拉动，而是中等水平学生群体的整体学习表现得到改善。特别是 2024—2025 年第 2 学期，平均分达到 65.16 分，中位数达到 68 分，较前两个学期有明显跃升，反映出校本资源、智能答疑、分层导学和过程反馈等措施对学生基础能力提升具有积极作用。

再次，从及格率和优秀率看，课程及格率由 48.08% 提高至 63.95%，提升 15.87 个百分点；优秀率由 4.08% 提高至 9.27%，提升 5.19 个百分点。其中，2024—2025 年第 2 学期优秀率达到 9.85%，较前期有较大幅度增长。虽然 2025—2026 年第 1 学期优秀率较上一学期略有回落，但仍保持在 9% 以上，明显高于实施初期水平。这表明校本化资源建设不仅提升了学生通过课程基本要求的能力，也在一定程度上促进了高水平学生的发展。及格率连续提升尤其说明，数字资源在基础薄弱学生的知识补偿、作业反馈和学习支持方面发挥了较明显作用。

最后，从学生评教结果看，评教得分由 92.23 分提高至 98.53 分，连续四个学期保持上升趋势，累计提升 6.30 分。这说明学生对课程组织方式、资源供给

质量、学习支持服务和教学互动体验的认可度不断提高。结合成绩数据可以看出，校本化实施并非单纯增加数字资源数量，而是通过课前预习、课中互动、课后答疑、实验训练和评价反馈等环节的协同运行，改善了学生的学习体验和学习成效。

综合来看，表中数据表明，程序设计类数字教学资源的校本化实施在扩大课程覆盖面的同时，有效提升了学生整体成绩、及格率、优秀率和课程满意度。其成效主要体现在三个方面：一是资源建设由静态材料供给转向全过程学习支持，增强了学生对知识点的理解和应用能力；二是基于校本知识库和智能反馈的教学机制，改善了学生基础差异带来的学习困难；三是教师主导下的人机协同教学提升了课程运行效率和教学组织质量。

虽然取得了一定成效，但是程序设计的高阶能力指标，如系统思维、问题分析设计能力、代码优化能力、创新能力、跨平台迁移能力等方面的训练和考核还未得到很好的实施。这些往往需要通过项目任务、实验报告、作品展示和口头答辩等方式评价。第四类是素养与价值指标，包括代码安全意识、伦理意识、责任感、团队协作、信息辨别能力和合理使用 AI 的能力。

10 结 语

本文提出从资源仓库到课程智能体的机制，强调程序设计类数字教学资源建设必须围绕校本课程目标、学生真实需求和人才培养质量进行系统重构。所谓校本化实施，就是让资源不再只是材料集合，而成为支持知识学习、能力发展、价值引导、过程评价与风险治理的协同体系。对于高校程序设计课程而言，这一转型的真正意义在于它能使课程资源从可被访问走向可被教学使用，从静态存在走向动态生成，从内容供给走向智能服务，最终推动程序设计类课程在大模型时代完成质量提升与范式更新。

参 考 文 献

- [1] 孙丽君, 杨志强, 高枚. 围绕计算思维培养的程序设计课程改革[J]. 计算机教育, 2013(5): 29-31+38.
- [2] 鉴萍, 陈宇峰, 李凤霞. 程序设计课程中计算思维和应用能力培养问题研究[J]. 计算机教育, 2016(7): 35-39.
- [3] 王海洋, 刘瑾, 胡静. AI 赋能的 C 语言程序设计课程教学改革[J]. 计算机教育, 2026(4): 105-109.
- [4] 程宏斌, 孙霞, 梁伟. AI 赋能的嵌入式系统设计与应用课程教学改革[J]. 计算机教育, 2026(4): 110-117.
- [5] 卢宇, 余京蕾, 陈鹏鹤. 基于大模型的教学智能体构建与应用研究[J]. 中国电化教育, 2024(7): 99-108.
- [6] 李杰. C 语言程序设计课程教学方法探讨[J]. 电脑知识与技术, 2018, 14(32): 179-181.
- [7] 杨晓贤, 张凯淇, 石林祥, 等. 基于大语言模型的 C 语言程序设计课程教学改革[J]. 计算机教育, 2025(12): 97-102.

-
- [8] 冷祥光, 罗永平, 胡新平. 基于多租户模型的高校校本教学资源大平台研究[J]. 现代教育技术, 2013, 23(3): 55-58.
- [9] 穆玲玲, 张行进, 职为梅, 等. 大模型背景下面向系统观的实验教学改革初探[J]. 计算机教育, 2025(7): 116-121.
- [10] 许岩. 程序设计类实验课的大单元教学模式设计与实践[J]. 计算机教育, 2025(8): 55-60.
- [11] 尹良泽, 徐建军, 李姗姗, 等. C 语言程序设计课程中的代码安全意识和能力培养[J]. 计算机教育, 2025(4): 53-57.
- [12] 鄢涛, 高朝邦, 古沐松, 等. 基于学科竞赛的编程类高阶课程改革[J]. 计算机教育, 2026(4): 59-64.
- [13] 丁林花, 王九如, 薛琳, 等. 地方高校计算机专业竞赛驱动分层教学的“普及+拔尖”创新人才培养[J]. 计算机教育, 2026(4): 65-71.
- [14] 王玉琢, 万程. 医学人工智能伦理融入思政教育的实践与探索——以智能医学工程专业程序设计基础 (Python) 课程教学为例[J]. 中国医学教育技术, 2023, 37(7): 168-170.
- [15] 刘强, 伍晓洁, 赵文涛, 等. 基于计算学科三形态和 CC2020 品行元素的网络安全课程思政教学实践方法[J]. 计算机教育, 2025(4): 58-64.