Advances in Education 教育进展, 2026, 16(11), 129-136
Published Online November 2026 in Hans. https://www.hanspub.org/journal/ae
https://doi.org/10.12677/ae.2026.16112366

Hans汉斯

生成式人工智能支持的数据结构教学改革

张力天[1,2]，李朋逊[1,2]

[1]北京邮电大学网络空间安全学院，北京
[2]可信分布式计算与服务教育部重点实验室，北京

收稿日期：2026年9月30日；录用日期：2026年10月26日；发布日期：2026年11月4日

摘　要

针对生成式人工智能辅助编程条件下数据结构课程中代码完成与概念理解脱节、算法分析缺少验证依据、学习评价难以识别独立能力等问题，本文围绕绪论、算法分析、线性表、栈与队列设计递进式教学任务，将独立建模、人机协同诊断、测试解释和约束变化迁移贯穿教学过程。以有界循环队列为例，给出容量约定、错误诊断、状态追踪及评价依据，并通过有限操作序列枚举检验示例程序与诊断用例。该方案发挥生成式人工智能在反馈生成、错误诊断和测试辅助等方面的作用，并将工具使用记录与学生独立解释共同纳入评价，为前四章教学资源优化和后续课堂验证提供依据。

关键词

数据结构，生成式人工智能，教学改革，反例诊断，过程性评价

Exploring Teaching Reform in the Data Structures Course with Generative Artificial Intelligence

Litian Zhang[1,2], Pengxun Li[1,2]

[1]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing
[2]Key Laboratory of Trustworthy Distributed Computing and Service, Ministry of Education, Beijing

Received: September 30, 2026; accepted: October 26, 2026; published: November 4, 2026

Abstract

To address the problems in the Data Structures course under generative artificial intelligence-assisted programming, including the disconnection between code completion and conceptual understanding, the lack of verification basis for algorithm analysis, and the difficulty in identifying students'

文章引用: 张力天, 李朋逊. 生成式人工智能支持的数据结构教学改革[J]. 教育进展, 2026, 16(11): 129-136.
DOI: 10.12677/ae.2026.16112366

independent competence in learning assessment, this study designs progressive teaching tasks for the units of introduction, algorithm analysis, linear lists, stacks and queues, and integrates independent modeling, human-AI collaborative diagnosis, test-based explanation and transfer under changing constraints throughout the teaching process. Taking the bounded circular queue as an example, this paper specifies capacity conventions, fault diagnosis, state tracing and assessment evidence, and verifies sample implementations and diagnostic test cases via finite enumeration of operation sequences. The proposed scheme leverages the strengths of generative artificial intelligence in feedback generation, fault diagnosis and test assistance, and incorporates both tool usage records and students' independent explanations into the assessment framework, providing a foundational basis for optimizing teaching resources of the first four chapters and carrying out subsequent classroom validation.

Keywords

Data Structures, Generative Artificial Intelligence, Teaching Reform, Counterexample Diagnosis, Formative Assessment

Copyright © 2026 by author(s) and Hans Publishers Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/
Open Access

1. 引言

数据结构课程承担从程序编写向问题建模和算法设计过渡的教学任务。学生需要将实际问题转化为数据对象、关系与操作，选择相应的存储表示，并说明算法正确性及其执行代价。CS2023 有关知识与胜任力模型的讨论强调知识掌握与任务情境中的能力表现之间的联系[1]。这一取向提示课程改革既要关注知识组织，也要明确学生能够独立完成什么，以及依据什么判断其完成质量。

基于建构主义学习理论与能力本位教育理念，学习的核心目标是推动学习者主动完成概念建构与问题解决能力的生成。围绕数据结构的抽象性和实践性，已有研究从可视化调试、小班教学与智能辅助等方面展开探索。连远锋等将算法运行状态展示与虚拟仿真实验结合，为理解数据结构的动态变化提供支持[2]；厉夫兵等采用动画、代码演示和随堂测试组织教学[3]。在生成式人工智能应用方面，唐玮嘉讨论了 AIGC 与数据结构教学的结合方式[4]，蒋松冬等从教学融合模式及伦理挑战分析了相关实施路径[5]。这些工作为教学资源建设与课堂组织提供了参考。

本研究依托面向编程场景的生成式大语言模型类工具，其核心具备代码解释、错误诊断、测试用例生成三类功能，可分别匹配概念辨析、故障排查、验证拓展三个教学环节。李清勇等对大模型在计算机实践教学中的作用进行分析，并通过过程报告和考核设计关注学生的实际掌握情况[6]。由此，本文聚焦一个具体问题：如何在允许合理使用生成式人工智能的条件下，使学生仍然承担结构选择、错误判断和算法解释的责任。围绕课程前四章，本文将教学任务、人工智能介入时点和学习证据对应起来，通过循环队列案例说明实施方法，并给出可复核的程序核验结果与后续评价方案。

2. 教学问题与改革目标

2.1. 从代码结果转向问题求解过程

数据结构学习中的困难常发生在相邻环节之间。学生能够复述队列的先进先出特性，却未必能够据此定义空队列和满队列的操作行为；能够运行一段程序，却未必能够解释下标如何变化；能够写出复杂度结论，却未必知道结论成立的操作条件。生成式人工智能缩短了获得代码的过程，也可能使这些理解

上的缺口被可运行的结果掩盖。上述四个方面均指向学生独立解决问题的能力，同时兼顾基础知识训练以及人工智能辅助活动中的任务边界。

本研究将前四章的学习要求归纳为建模、验证、解释和迁移四个方面。建模要求学生确定数据关系、操作接口及边界约定；验证要求学生构造能够区分正确与错误实现的输入；解释要求学生将程序状态与逻辑结构、基本操作与复杂度相联系；迁移要求学生在容量、操作规则或存储表示改变后调整方案。四个方面共同服务于独立问题求解，既保留基础知识训练，也为人工智能辅助活动设置明确的学习任务。

2.2. 围绕前四章组织递进任务

章节安排依照“绪论–算法分析–线性表–栈与队列”展开，以记录管理和请求处理等简化情境贯穿教学。情境用于说明数据结构的选择依据，不要求学生掌握大模型训练或服务部署。各章任务与学习证据的对应关系见表 1。绪论重在从需求中识别结构，算法分析重在明确计数对象，线性表重在比较表示方式，栈与队列重在根据操作约束实现和验证抽象数据类型。

Table 1. Progressive tasks and learning evidence in the first four units
表 1. 前四章递进任务与学习证据

章节	核心问题	任务安排	学习证据
绪论	需求如何转化为结构与操作	分析记录管理需求， 定义对象、关系和接口	数据关系图；接口及边界约定
算法分析	代价由哪些基本操作决定	对记录访问和删除进行计数与比较	计数过程；复杂度及适用条件
线性表	存储表示如何影响操作代价	比较顺序与链式记录维护方案	状态图；实现及选择依据
栈与队列	操作约束如何转化为实现规则	实现括号匹配与有界请求队列	边界测试；反例与迁移说明

任务难度通过改变约束逐步增加，而不依靠不断扩大程序规模。以记录维护为例，首先确定记录之间的逻辑关系，再分析按位置访问与按值删除的代价，随后比较顺序存储和链式存储，最后在先进先出约束下实现有界队列。同一情境在不同章节承担不同的认知任务，有助于学生区分“问题要求什么”和“某种实现恰好怎样工作”。对于尚未掌握前置概念的学生，提供状态填空和局部代码任务；完成基础任务后，再要求其解释表示方式变化带来的影响。

3. 人机协同教学过程

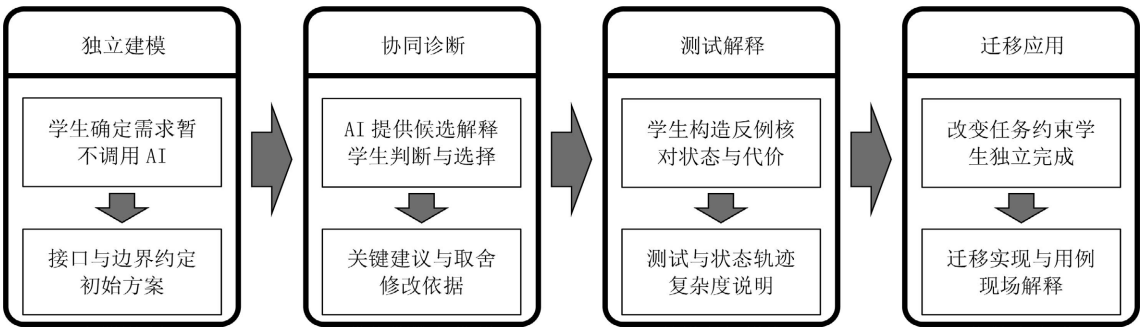


Figure 1. Human-AI collaborative teaching process and learning evidence
图 1. 人机协同教学过程及学习证据

教学过程按照独立建模、协同诊断、测试解释和迁移应用组织，如图 1 所示。生成式人工智能主要用于生成备选解释、识别可能遗漏的边界条件并辅助构造测试用例，为学生分析和验证程序提供支持。学生在调用工具前保留独立作答，在获得反馈后说明接受或拒绝建议的依据，在任务结束时独立完成约束变化题。各环节形成的过程性材料共同构成评价证据，据此可区分人工智能参与部分与学生独立完成部分。

3.1. 独立建模：先确定问题再调用工具

教师首先给出任务情境及输入、输出要求，暂不提供完整实现。学生用自然语言或伪代码描述数据关系和基本操作，写出正常情况与边界情况的处理规则。例如，对有界请求队列，需要明确容量表示可保存的元素数，队满时拒绝新请求还是替换旧请求，空队列出队时返回什么。学生同时给出一个正常操作序列和一个边界序列，形成可供后续对照的初始方案。

教师通过短时作答了解学生是否混淆逻辑结构与物理位置、是否忽略异常状态，再选取典型方案开展课堂讨论。保留初始错误判断记录，使后续修改能够追溯到具体概念根源，便于定位认知误区，厘清底层原理。这一环节可借助 AI 的代码解释功能对比初始方案与标准实现的概念差异，为学生自主定位认知偏差提供参照。

3.2. 协同诊断：用证据筛选人工智能建议

进入协同环节后，学生向人工智能提供自己的接口约定、局部代码和失败现象，请其提出可能的原因及区分这些原因的测试，不直接索取整份作业答案。例如，学生可以借助生成式人工智能检查队列实现是否满足给定的容量约定，分析可能遗漏的边界条件，并获得相应的测试建议。教师不将某一提示词视为固定答案，而是要求学生判断反馈是否与任务前提一致：建议若改变了满队列处理策略，即使代码能够运行，也不能直接用于原任务。

学生为关键修改保留“原判断－工具建议－验证依据－最终处理”四项记录，只截取影响决策的内容。支持 AI 反馈的在线编程平台研究已讨论代码分析、测试生成等功能[7]，本文据此将反馈落实到可检验的问题上。对于相互矛盾的解释，学生需通过运行轨迹、反例或教材中的定义作出选择。教师在课堂讨论中追问“哪一条证据改变了判断”，据此识别学生是否理解修改原因。

3.3. 测试解释：使运行结果能够支持结论

测试活动同时关注结果与状态。结果检查比较返回值及元素顺序，状态检查关注下标范围、元素个数和存储内容之间的关系。学生需说明某个用例针对哪一种错误，以及若删除该用例会遗漏什么。由此，测试不仅用于核对运行结果，还用于验证程序实现是否满足抽象数据类型的语义约束。对能够通过常规用例的错误版本，教师可引导学生利用生成式人工智能提出候选反例，再通过缩短操作序列寻找最早出现状态分歧的位置。这一过程可借助 AI 的测试用例生成功能扩充边界测试集合，辅助学生完成更全面的状态验证。

解释活动要求学生将程序执行与算法分析相互对应。对于删除线性表首元素，学生标出被移动的元素及其数量；对于循环队列出队，学生说明为何只需更新有限个状态量。若使用计时实验，则应保持操作语义、数据规模和运行环境可比，区分算法本身与打印、绘图等辅助开销。实测结果与理论预测不一致时，应首先检查计量对象和实验条件，避免将单次运行时间直接当作复杂度结论，重视复现多组实验数据。

3.4. 迁移应用：在变化条件下检查独立掌握

迁移任务改变一项关键约束，并要求学生先独立说明哪些部分保持不变、哪些部分必须修改。例如，

将队满拒绝新请求改为保留最近的若干条记录，或者将顺序存储改为链式存储。学生提交修改前的预测、核心代码和验证用例，并在限定时间内解释一处关键决策。迁移环节关闭生成式人工智能工具，仍可提供统一的语言语法资料，使评价重点落在结构与算法的理解上。

前述环节可安排在一个理论教学单元与配套实验中：课前完成初始方案，课内讨论模型反馈和典型反例，实验阶段完成测试与修改，单元结束进行独立迁移。课堂讨论优先处理共性概念错误，个别语法问题交由工具提示和课后答疑解决。未能获得稳定工具服务的学生使用教师预先核验的反馈材料，完成同样的判断任务，以减少工具条件差异对学习要求的影响。

4. 循环队列教学案例

4.1. 明确容量约定与状态表示

案例将请求处理简化为有界先进先出队列，每个请求用一个标识表示，不涉及并发调度。设容量 C 为正整数，最多保存 C 个元素；入队成功返回 OK，队满返回 FULL 且原内容不变；出队返回最早进入的元素，队空返回 EMPTY 且状态不变。返回状态与数据值在实现中应采用可区分的表示，示例中的请求标识不使用这些保留值。统一约定后，再讨论存储结构，能够避免学生因接口含义不同而得出相互矛盾的判断。

采用长度为 C 的数组 a、队头下标 f 和元素个数 n 表示队列。状态应满足 $0 \le f < C$ 和 $0 \le n \le C$；逻辑位置 i 处的元素保存于 $a[(f+i) \bmod C]$，其中 $0 \le i < n$。入队前检查 n 是否等于 C，未满时写入 $a[(f+n) \bmod C]$并将 n 加 1；出队前检查 n 是否为 0，非空时读取 a[f]，随后更新 $f = (f+1) \bmod C$ 并将 n 减 1。空队列不要求 f 必须回到 0，学生需区分“为空”与“数组下标复位”这两个不同条件。

教材中还常使用队头、队尾指针并预留一个槽位区分空、满状态。该表示本身是合理的，但物理数组长度与可存元素数的关系不同。本案例采用计数表示，不能将预留槽位表示中的满判断直接移入。教学讨论的重点因此落在表示约定是否一致，而不是将不同教材中的公式简单判为正确或错误。

4.2. 通过反例定位三类概念错误

教师围绕容量、写入位置和回绕构造三个错误版本，具体诊断任务见表 2。这些版本是用于教学的人工构造程序，不代表学生实际错误分布，也不用于估计任何模型的出错概率。学生先预测输出，再运行用例，并根据最早出现的状态分歧定位问题；必要时可向人工智能询问解释，但最终判断需回到接口约定及程序证据。

Table 2. Artificial faults and diagnostic tasks for the circular queue
表 2. 循环队列的人工构造错误与诊断任务

错误类型	错误设置	最短诊断序列	应解释的问题
容量约定混用	计数表示中误以 $n = C - 1$ 判断队满	连续入队 3 次，第 3 次被误拒绝	为何实际容量少了 1 个元素
写入位置偏移	误将写入下标设为$(f + n + 1) \bmod C$	入队 1 次，逻辑首元素不符	下一写入位置如何从状态推导
队头遗漏回绕	出队后仅令 $f = f + 1$，未按容量取模	入队 3 次、出队 3 次，队头下标越界	为何输出正确仍需检查状态

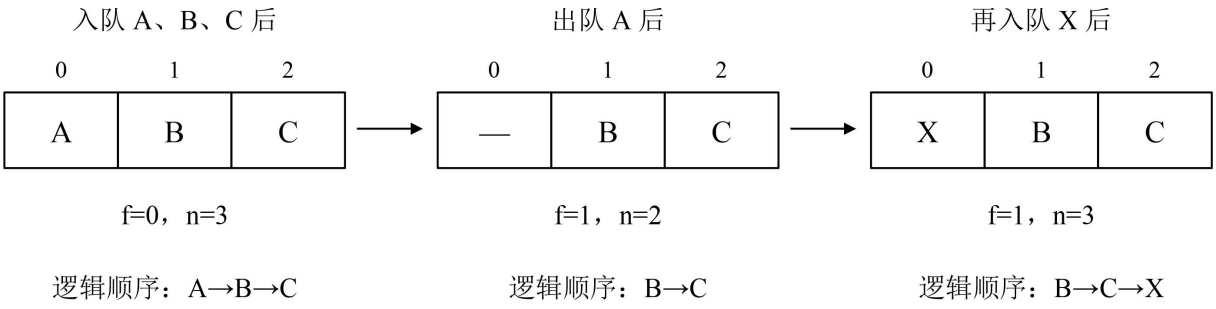
注：均固定 $C = 3$，从空队列开始；诊断在每一步检查返回值、逻辑内容和状态约束。

三类错误分别对应三个需要辨析的问题。容量错误需要比较“有多少存储单元”和“允许保存多少

元素”；位置错误需要将逻辑队尾与下一次写入位置分开；回绕错误则要求学生检查中间状态的合法性。对于遗漏取模的版本，连续取出全部元素时返回值可能仍然正确，但队头下标已经越界。如果只核对输出，错误会在后续操作才暴露，因此应将状态约束纳入验证。

4.3. 用状态追踪解释物理位置与逻辑顺序

图 2 给出容量为 3 时的典型轨迹：依次入队 A、B、C，出队 A 后再入队 X。最后数组按物理下标排列为 X、B、C，而逻辑出队顺序为 B、C、X。学生需根据 f 和 n 重新构造逻辑序列，解释为什么回绕写入不会破坏先进先出关系。相较于直接展示动画，按照预测、状态展示和差异解释的顺序组织教学，可利用状态图检验学生对物理位置与逻辑顺序关系的理解。



数组上方数字为物理下标；“—”表示该槽位当前无有效元素。

Figure 2. State tracing of a circular queue with capacity three
图 2. 容量为 3 的循环队列状态追踪

为检查示例实现及诊断用例，本文使用 Python 标准库编写有限枚举脚本，以顺序列表作为行为参照。操作字母表包含入队 A、入队 B 和出队三种操作，枚举长度 0 至 6 的所有序列；对每个容量，共检查 $1+3+3^2+\ldots+3^6=1093$ 条序列。容量分别取 1、2、3、5，共形成 4372 个“容量–序列”组合，每一步比较返回值、逻辑内容及状态约束，正确版本在该范围内未发现不一致。

对于表 2 中的错误版本，固定容量为 3 并采用相同枚举方法，分别检出 640、1086 和 40 条失败序列。相应的最短反例为连续 3 次入队、1 次入队，以及连续 3 次入队后连续 3 次出队。失败序列数仅说明这些人工构造错误在当前枚举集合中的可检出情况，受操作集合、最大长度及状态检查规则影响。该核验用于保证课堂示例可复现，不构成任意容量与任意长度下的正确性证明，也不作为教学效果数据。

4.4. 从算法解释过渡到规则迁移

在计数表示下，循环队列单次入队和出队只执行常数次下标计算、读写及计数更新，时间复杂度为 O(1)，存储空间为 O(C)。若用顺序表实现队列，并在每次出队后移动剩余元素，则出队代价与当前元素数 n 相关，为 O(n)。这里的比较建立在相同队列语义下，不能把支持动态扩容的容器与固定容量拒绝策略混为同一任务。状态可视化若每次复制或展示全部有效元素，还会引入 O(n)辅助开销，应与被分析的队列操作分开。

迁移任务将应用场景改为“仅保留最近 C 条记录”。队满时，新记录应替换最早的记录，因而需要调整写入与队头更新规则；未满时仍按普通入队处理。学生必须先说明该变化已改变原接口语义，再给出容量为 1 和发生回绕时的测试。任务评价关注其能否从需求变化推导程序变化，而不是能否复用某段现成代码。基础较好的学生还可比较链式队列的表示与空间开销，说明两种实现共同遵守的逻辑约束。

5. 学习评价与实施保障

5.1. 结合过程证据与独立任务进行评价

仅以最终代码通过率评价学习，难以判断学生是否掌握建模和解释方法。何文广等将自动评价与程序设计实验教学结合，利用反馈支持教学改进[8]。在此基础上，本文将自动检查定位为正确性证据之一，并结合初始方案、修改依据、口头解释与独立迁移进行综合评价。表 3 给出建议量规，权重用于示范教学设计，实际应用需与课程既有考核要求协调。

具体评分采用“按维度识别证据，再判断完成程度”的方式。能够独立给出充分依据且完成迁移的表现，可评为相应维度的较高水平；经提示后能够修正并解释原因的表现，记为基本达到；仅保留正确结果而无法说明关键决策的表现，不能视为已达到该维度要求。人工智能使用次数、对话篇幅和代码长度不直接计分，合理拒绝不符合任务前提的建议同样可以成为有效学习证据。

Table 3. Learning assessment dimensions and proposed weights
表 3. 学习评价维度及建议权重

评价维度	权重/%	主要证据	达到要求的表现
问题建模	25	独立初稿、接口约定、边界条件说明	结构符合需求，正常与异常行为明确
实现与验证	35	关键修改记录、反例、状态轨迹与测试结果	能够定位错误，用证据说明修改理由
算法解释	20	基本操作计数、复杂度推导及条件说明	推导与实现一致，区分算法和辅助开销

为减少评价差异，教师在布置任务时公布典型证据及判定示例，使用一份匿名样例进行集体讨论，再对部分作业进行交叉复核。独立迁移结果与协同阶段表现分别记录，避免二者在总分中相互掩盖。若学生的协同任务完成度较高而独立解释薄弱，后续辅导应回到概念与状态推演，不能据此直接认定其存在不当使用工具的行为。

5.2. 依据错误类型调整资源与辅导

教学反馈以知识点和错误类型为索引，而非只按照总分排序。苏庆等通过学习状态与题目特征的关联研究编程实训推荐[9]，基于上述研究对学习差异的关注，本文根据错误类型设置分层补充任务：对容量约定理解不足的学生补充接口辨析任务，对下标变化理解不足的学生补充状态追踪任务，对复杂度解释不足的学生补充基本操作计数任务。教师根据后续独立任务判断是否需要继续补强，而不将一次错误固定为学生能力标签。

课程资源按照“任务说明、状态示例、参考实现、错误版本、测试集合、评价说明”组织。生成式人工智能可辅助生成解释材料、错误版本和测试用例，扩充课程教学资源；相关内容均应经过教师核验，重点检查术语、边界条件和复杂度分析的适用前提。课堂只使用去除个人信息的学习片段讨论，不要求学生向外部工具上传完整作业身份信息。教学记录保留能够支持判断的必要内容，使资料整理与实际教学工作相适应。

5.3. 后续课堂验证安排

目前已完成前四章部分课件的优化，本文进一步形成了配套任务设计及循环队列示例核验，尚未取得用于判断教学效果的班级对照数据。后续实施拟优先观察三个指标：边界条件解释的完整性、诊断用例的区分能力，以及独立迁移任务的完成质量。采用前后测时应设置知识点一致、情境不同的题目，避

免将对某一示例的熟悉程度误作能力提升；评价材料由教师依据统一量规评分，并报告实际参与人数和缺失情况。

针对大班教学场景，可引入同伴互评机制分担教师在“解释”环节的评价压力，也可开发半自动化分析工具辅助提取学生决策过程中的关键证据，降低方案落地的师资门槛。如具备多个教学班的实施条件，还需记录前置课程基础、教学时数、教师安排和人工智能使用条件，分析这些因素对结果解释的影响。除成绩变化外，可选取修改过程与独立解释存在差异的案例，检查改革措施在哪个环节发挥作用、在哪个环节仍需调整。正式的效果结论应在真实教学数据和分析结果基础上形成，本文不预设其方向。

6. 结语

生成式人工智能为数据结构教学提供了即时解释、代码诊断和测试生成等新的支持方式，同时要求课程进一步明确学生在问题求解中应独立承担的责任。本文围绕前四章建立递进任务，将独立作答、协同诊断、测试解释与迁移应用相衔接，并以循环队列给出接口约定、反例、状态图和程序核验，形成可以落实到课件、实验与评价中的教学方案。现阶段工作主要完成教学设计及示例验证，后续将结合课堂实施考察学生独立能力的变化，逐步将该教学框架拓展至图的遍历、最短路径算法等更复杂的数据结构内容，探讨方案的适用边界与针对性调整方法，依据证据持续调整任务难度和工具介入方式。

参考文献

[1] 向乔, 刘雨桐, 孔令和, 等. CS2023: 计算机科学教育知识与胜任力模型并轨的全新探索[J]. 计算机教育, 2023(4): 9-14.

[2] 连远锋, 张鑫, 庄永琪, 等. 数据结构与算法可视化调试虚拟仿真实验系统设计[J]. 实验技术与管理, 2023, 40(5): 122-129, 159.

[3] 厉夫兵, 孙剑. “数据结构与算法”小班教学方法探索实践[J]. 教育教学论坛, 2024(17): 133-136.

[4] 唐玮嘉. 融入 AIGC 的数据结构教学探讨[J]. 教育进展, 2025, 15(8): 1567-1572.

[5] 蒋松冬, 韦金琼, 李春青, 等. 生成式 AI 在计算机专业教学中的融合模式与伦理挑战研究: 以数据结构课程为例[J]. 教育进展, 2026, 16(2): 32-40.

[6] 李清勇, 耿阳李敖, 彭文娟, 等. “私教”还是“枪手”: 基于大模型的计算机实践教学探索[J]. 实验技术与管理, 2024, 41(5): 1-8.

[7] 厉旭杰, 顾雨辰, 姚持恩, 等. 集成 AI 大语言模型的在线编程实验平台设计与实现[J]. 实验技术与管理, 2024, 41(8): 215-221.

[8] 何文广, 周珂, 熊刚强, 等. 程序设计课程实验教学改革与实践[J]. 实验室研究与探索, 2016, 35(6): 163-165, 169.

[9] 苏庆, 陈佳欣, 黄鸿林, 等. 基于深度知识追踪的个性化推荐编程实训系统建设与教学实践[J]. 实验技术与管理, 2023, 40(3): 206-211.