

基于生成式教学智能体的问题提示支架在编程教育中对学生学习效果的影响研究

The Impact of Problem Prompting Scaffolds Based on Generative Pedagogical Agents on Students' Learning Outcomes in Programming Education

程玉梅¹, 李知霞², 阙荣革¹, 乔爱玲^{1*}

¹首都师范大学教育学院

²山西省孝义中学

* qiaoal@126.com

【摘要】 随着智能时代的深入发展, 利用生成式人工智能技术 (GAI) 改善学习效果已经受到学界越来越多的关注。但目前生成式教学智能体 (GPA) 在学习中的应用效果还未得到充分探究。因此, 研究以参与编程教育的 92 名高一学生为研究对象进行为期 6 周的随机对照试验, 旨在调查基于 GPA 的问题提示支架工具对学生计算思维、学业表现、元认知能力的影响。研究发现, 基于 GPA 的问题提示支架能够促进学生的计算思维、学业表现与元认知能力。文章深入分析和讨论了研究结果, 旨在为 GPA 融入教学提供证据参考。

【关键词】 生成式教学智能体; 问题提示支架; 编程教育; 计算思维; 元认知

Abstract: The use of generative artificial intelligence technologies to improve learning has received increasing attention from the academic community, but the effectiveness of generative pedagogical agent in learning has not yet been fully explored. Therefore, the study conducted a 6-week randomized controlled trial with 92 senior high school students participating in programming education with the aim of investigating the effects of a GPA-based question prompt scaffolding tool on students' computational thinking, academic performance, and metacognitive abilities. The study found that the GPA-based question prompt scaffolding was able to promote students' computational thinking, academic performance, and metacognitive abilities. The article analyzes and discusses the results of the study, aiming to provide evidence to inform the integration of GPA into teaching and learning.

Keywords: generative pedagogical agent, question prompt scaffolding, computational thinking, metacognition

1. 前言

21 世纪的学习者应具备创新能力、批判性思维、问题解决能力 (Rehman et al., 2023)。在计算机科学领域, 编程教育被认为是培养创造力、计算思维的首选方法, 对于培养解决问题的能力 and 批判性思维具有至关重要的作用 (Mathew et al., 2019)。然而, 有研究表明, 学生在编程学习过程中也可能会遇到问题, 如缺乏相关技能或无法获取关键资源 (Tom, 2015), 而这种挑战或许会对学生学习产生负面影响。对此, 有必要为学生的编程学习提供全面支持, 如设计教学支架帮助学生整理思维、设计算法、顺利地参与问题解决的过程 (Sun et al., 2021)。

在智能技术的深入推进下, 生成式人工智能技术 (GAI) 近年来受到学界广泛关注, 其基于自然语言支持的对话能力使其在教学与反馈中具有良好应用潜力, 为学生提供个性化和便捷的学习体验 (Guo et al., 2023), 这为解决编程教育中存在的问题提供了新的思路。已有研究证实人工智能支持工具和环境的优势可以有效地提高学生的计算思维技能、编程自我效能感和上课的积极性 (Yilmaz & Yilmaz, 2023)。而纵观目前关于 GAI 的研究, 大多集中在单一 GAI 支持的教育对学生学习效果的影响, 或是集中在教育应用框架、风险及应对建议 (Yusuf et al., 2024) 等理论层面, 鲜有学者将研究聚焦在面向学生个人特征设计智能体支持学习过程, 以探讨基于生成式人工智能技术的智能体对学生学习效果的影响。而学生在学习过程中具有个性差异, 基于学生

知识水平、个人特征等进行针对性反馈仍至关重要(Keuning & van Geel, 2021)。基于此, 本研究立足于高中编程课程, 旨在调查基于 GAI 的智能体辅助编程教育对学生计算思维、学业成绩与元认知能力的影响。

2. 基于生成式教学智能体的问题提示支架设计

2.1 生成式教学智能体设计

生成式教学智能体(Generative Pedagogical Agent, 简称 GPA)是基于 GAI 创建的具有教学功能的智能体, 使用预设的指令和教学材料引导 GAI 实现在指定学科教学中支持学生学习(张渝江 et al., 2024)。主要是通过提示词和知识库内容结合来增强 GAI 以展现更好的教学行为和内容输出。研究参考在 Chee 的提示词设计基础上进行改编(Ng & Fung, 2024), 如表 1 所示改变后的提示词设计旨在从学习者那里引出特定信息或提供有针对性的建议。输入通常包括学习者特定的数据, 例如当前的知识水平、面临的挑战和教育目标。输出是定制的建议或解释, 指导学习者完成个性化的学习之旅。

表 1 生成式教学智能体设计

通用设计	主题/背景	介绍本节课程的主题或者背景信息
	学习目标	根据布鲁姆的教学目标分类理论设计知识、理解、应用、分析、综合、评价六个层次学习目标
个性设计	初步评估	[提示]: 根据学习者当前对[主题]的理解, 评估他们的知识水平并建议他们接下来应该学习的概念, 以掌握[主题]; [输入]: 学习者当前对给定主题的理解或表现; [输出]: 根据学习者当前知识水平定制建议概念的列表;
	澄清	[提示]: 学习者认为[主题]中的哪些具体领域具有挑战性? 请提供详细信息; [输入]: 学习者对他们认为困难的领域的反馈; [输出]: 对具有挑战性的领域的详细描述, 将用于调整学习路径;
	解释	[提示]: 解释为什么在[概念 B]之前学习[概念 A]有利于学习者对[主题]的理解; [输入]: 学习路径中的概念序列; [输出]: 详细说明建议序列的解释;

2.2 问题提示支架设计

在 GAI 应用过程中, 具体、明确的提问更能激发其生成高质量的回答(Chen et al., 2024)。然而, 大多数学生缺少该方面的元认知能力。一个可能的解决方案是教师为学生提供更多关于以适当方式查询 ChatGPT 的策略的指导。其中, 问题提示支架是一种可能的方法, 它支持学生将更好的提示表述为问题(White et al., 2023)。本研究设计的问题提示支架在 5W(戴岭 et al., 2023) (Who—分配模型扮演的角色, What—设置模型执行的任务或附上示例, When—确定模型完成任务的时间, Where—赋予提示的位置或场景, Why—告知提示的理由、动机或目标)、5S(Set the Scene、Be Specific、Simplify your Language、Structure the Output、Share Feedback)(Tassoti, 2024)提示的基础上, 融合不良结构问题解决步骤(问题表述、生成解决方案、提出理由、监控和评估)及其提示(Ge & Land, 2004)进行改编, 设计如图 1 所示。

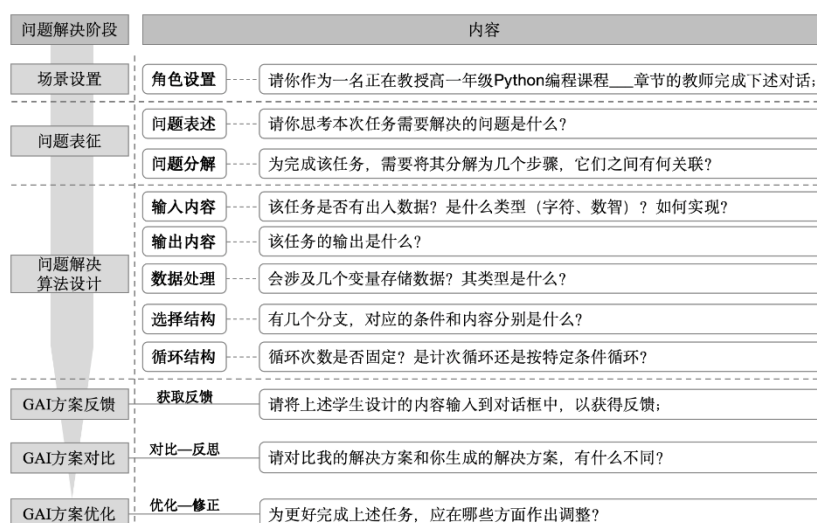


图 1 问题提示支架

3. 研究设计

3.1 研究对象

本研究在中国山西省某高中高一年级《信息技术》课程中随机选择了两个平行班,共 92 名学生,学生年龄 15-16 岁之间。其中,实验班为面向生成式教学智能体的问题提示支架组,共学生 47 名,对照组为面向生成式教学智能体的传统教学组,共 45 名学生。

3.2 研究流程

整个实验实施过程如图 2 所示,教学活动持续 6 周,每周 1 课时(50 分钟)。正式实验开始前,两个班的学生进行计算思维、Python 知识测试和元认知能力的前测调查。同时,授课教师向所有学生介绍如何基于上述 GPA 框架设计符合个人特征的 GPA,并向实验组学生介绍问题提示支架如何使用,两组学生分别结合 ChatGPT-4 进行练习。之后,对照组与实验组学生同时开展 Python 编程程序设计的学习活动,两组学生涉及相同的学习内容和任务,实验组学生在采用基于 GPA 的问题提示支架,对照组采用基于 GPA 的传统学习方式。为防止学生认知懒惰,教师要求在学生完成算法设计后再应用基于 GPA 的问题提示支架进行后续交互。教学活动结束后,两组学生进行计算思维、Python 知识测试和元认知能力的前测调查。

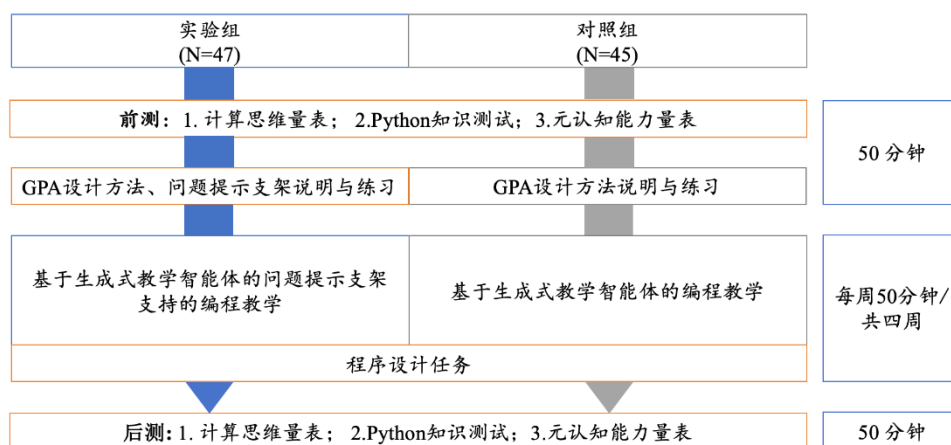


图 2 实验流程图

3.3 研究工具

3.3.1 计算思维量表

研究选用 Korkmaz 等人开发的计算思维测评量表(Korkmaz et al., 2017), 通过前后测试评估学生计算思维的变化。该量表包含创造力 (1-8 题)、问题解决 (9-14 题)、算法思维 (15-20 题)、批判思维 (21—25 题) 和协作能力 (26—29 题) 5 个子维度, 采用李克特 5 级量表进行评分, 分数越高, 学生的计算思维能力越高。整个量表的 Cronbach's α 值为 0.931, 分维度依次为 0.887、0.847、0.912、0.884、0.903, 均大于 0.7 表明该量表具有较好信度。

3.3.2 Python 知识测试试卷

研究基于 Python 预测试评估学生的知识水平, 结合高中教材数据与计算章节的综合题和往年的信息技术考试题, 由研究人员与信息技术教师协商共同编织。Python 知识测试来自实验期间涵盖的材料, 由完全客观的 20 个选择题组成, 涵盖 Python 语言的基础知识, 重点是循环结构、多分支结构和基本编程语法。为确保测试结果的可靠性, 研究对试题提供了明确的说明, 以避免任何误解。

3.3.3 元认知能力量表

研究使用 Sperling 等人提出的元认知能力量表(Sperling et al., 2002), 该量表包括元认知知识 (1-6 题) 和元认知调节 (7-18 题) 两个维度, 采用李克特 5 级量表进行评分。该量表的 Cronbach's α 值为 0.901, 分维度依次为 0.783、0.868, 均大于 0.7 表明该量表具有较好信度。

4. 研究结果

4.1 基于 GPA 的问题提示支架对学生计算思维的影响

研究采用独立样本 T 检验, 对两个组学生进行实验前的计算思维进行分析, 经分析可知, 实验组与对照组的计算思维不具有显著性差异($p=0.309>0.05$), 并且在五个具体的分维度也不具有显著性差异。说明实验开始前, 实验组和对照组的计算思维水平基本相同, 不存在差异。同样, 研究检验了实验结束后实验组与对照组的计算思维水平。独立样本 T 检验的结果如表 2 所示, 计算思维水平表现出显著差异($t=0.80, p=0.040<0.05$), 并且在算法思维($p=0.020<0.05$)维度上存在显著性差异, 但在其他分维度不具有显著性差异。

此外, 研究对测试前和测试后的数据进行了配对样本 T 检验, 以评估计算思维是否存在差异。

表 3 的结果表明, 实验组的计算思维能力显著提高。基于生成式教学智能体的问题提示支架有助于增强计算思维, 在计算思维的各个维度上显示出显著的提升。在对照组中, 学生主要提高了算法思维。

表 2 学生实验后计算思维的比较

	实验组		对照组		t	p
	M	SD	M	SD		
创造力	3.92	0.56	3.76	0.49	1.47	0.78
问题解决	3.58	0.68	3.40	0.67	1.24	0.88
算法思维	3.37	0.90	3.28	0.62	0.56	0.02*
批判思维	3.72	0.65	3.57	0.61	1.10	0.58
协作能力	4.19	0.73	3.96	0.66	1.54	0.14
计算思维	3.74	0.56	3.58	0.47	0.80	0.04*

* $p<0.05$.

表 3 学生测试前后计算思维差异

	实验组		t	对照组		t
	M	SD		M	SD	
创造力	3.68	0.61	2.23*	3.58	0.55	2.04*
	3.92	0.56		3.76	0.49	
问题解决	3.29	0.74	2.36*	3.09	0.72	2.48
	3.58	0.68		3.40	0.67	
算法思维	3.27	0.78	0.83*	3.27	0.57	0.08*
	3.37	0.90		3.28	0.62	
批判思维	3.47	0.70	2.09*	3.43	0.63	1.36
	3.72	0.65		3.57	0.61	
协作能力	3.73	0.85	3.25*	3.70	0.66	2.10
	4.19	0.73		3.96	0.66	
计算思维	3.48	0.54	2.71**	3.40	0.44	2.22
	3.74	0.56		3.58	0.47	

* $p < 0.05$; ** $p < 0.01$.

4.2 基于 GPA 的问题提示支架对学生 Python 知识测试成绩的影响

研究使用协方差分析检查两组学生之间 Python 学习表现的差异，模型中包含协变量（预测表现）和自变量（实验方法）。排除预测试的影响后，观察到两组学习成绩存在显著差异（ $F = 8.50$, $p < 0.01$ ）。之后，研究对测试前和测试后的数据进行了配对样本 T 检验，以评估 Python 知识测试成绩是否存在差异。表 4 的结果表明，与不使用任何脚手架的对照组相比，使用面向生成式教学智能体的问题提示支架的实验组学生取得了更高的学习成绩。

表 4 学生测试前后学习测试成绩差异

	M	SD	M	SD	t
实验组	72.87	17.15	76.86	20.69	1.27*
对照组	66.83	21.37	74.33	12.64	2.69

* $p < 0.05$.

4.3 基于 GPA 的问题提示支架对学生元认知能力的影响

研究对两个班级的前后测数据进行配对样本 T 检验，以探讨基于 GPA 的问题提示支架是否能够增强学生的元认知能力，结果如表 5 所示。从中可以看出，经过教学实践后，实验组学生的元认知能力（ $p = 0.018$ ）有显著所提高。此外，实验班学生在元认知知识维度上存在极显著差异（ $p < 0.01$ ），在元认知调节维度上存在显著差异（ $p = 0.012$ ），表明结合 GPA 的问题提示支架的编程教学实践可以提高学生的元认知能力。此外，对照组学生在元认知知识维度上有显著性提升（ $p = 0.03$ ）。

表 5 学生测试前后元认知能力差异

	实验组		t	对照组		t
	M	SD		M	SD	
元认知知识	3.63	0.57	2.23**	3.63	0.55	1.89*
	3.93	0.61		3.78	0.68	
元认知调节	3.55	0.51	1.09*	3.48	0.51	2.32
	3.64	0.64		3.64	0.65	
元认知能力	3.58	0.50	2.15*	3.53	0.48	2.66
	3.74	0.60		3.69	0.63	

* $p < 0.05$; ** $p < 0.01$.

5. 讨论

5.1 基于 GPA 的问题提示支架能够显著促进学生的计算思维水平

在本研究中，实验组相较于对照组表现出了更高水平的计算思维，可以认为，基于 GPA 的问题提示支架能够显著促进学生的计算思维，这与 Gong 等人的研究结果一致(Gong et al., 2024)。其原因可能在于，计算思维的培养主要依托于编程任务，按照问题提示支架的步骤为学生提供元认知支持。具体来看，开发程序时，学生根据教师提供的支架分析程序语句、补充完善自己的程序结构和程序语句；调试程序时，学生可以进一步分析 GPA 反馈的程序内容与相关解释，进而修正自己的错误。此外，面对复杂的编程问题，如果将其作为一个整体来寻找解决方案，学生可能会产生畏惧心理，相较于对照组学生，基于 GPA 的问题提示支架帮助学生将一个复杂的编程任务进行拆分，为明确的编程任务和编程语句提供支持，学生可以根据 GPA 反馈的编程任务提示深度加工编程知识和概念，借助这些提示对编程过程中出现的问题进行校对，并在反复操作和试错的过程中积累程序编写的经验(Molenaar et al., 2010)，以此不断强化对编程语言的理解，加深对计算概念和计算实践的掌握。

5.2 基于 GPA 的问题提示支架能够提升 Python 知识测试成绩

对学生后测 Python 知识测试结果的分析表明，实验组学生的成绩获得了更显著的提升。可以认为，基于 GPA 的问题提示支架能够提升 Python 知识测试成绩，这与 Malik 等人的研究结果相似(Malik et al., 2022)。这可能是由于学生与 GPA 互动能够获得个性化支持与反馈，帮助他们按照自己的节奏进步并提高编程技能。具体来看，具有明确学习目标与学习者个人特征的 GPA 能够为学习者解决程序设计过程中的常见编程错误、语法和语义等内容提供具体的反馈与解释。此外，GPA 能够激发有趣的学习体验来满足学生的需求，缓解了学生因长时间等待反馈而产生的心理压力，从而增强学生对技术易用性和有用性的感知，改善了学生的学习体验，为有效促进学生学习成绩的提升提供可能(Muñoz-Carril et al., 2021)。

5.3 基于 GPA 的问题提示支架能够显著促进学生元认知水平的发展

相较于对照组，实验组表现了更高的元认知能力，在元认知知识和元认知调节两个维度具有显著性提升。可以认为，基于 GPA 的问题提示支架能够显著促进学生元认知水平的发展，这与 Abdul-Rahman 等人的研究结果一致(Abdul-Rahman & Du Boulay, 2014)。究其原因，一方面，GPA 采用提示学习与人类反馈相结合的训练方式，具有基于用户提问进行个性化知识生产的能力，能够根据提问提供多轮次、流畅、自然的回答。另一方面，问题提示支架为 GPA 提供具体的背景、明确的问题边界、连贯的上下文、指定形式的反馈内容，即提供了一种明确形式的元认知指导，帮助学习者能够轻松地使用和内化，特别是帮助学习者回忆和激活先验知识来促进思想的阐述、明确解决问题的过程和相应的提问策略(Xie & Bradshaw, 2008)。同时，GPA 在对比、评估学生生成的问题解决方案后生成的反馈能够增强学生对任务的理解，这种反思性提示能够促进学生元认知水平的发展(Sijmkens et al., 2023)。此外，研究发现问题提示支架可以促进学生讨论的深度，对照组的学习者在不使用问题提示时很少会主动提出问题或深入讨论他们的想法和观点，导致无法检查改进点以及在解决方案中出现问题时准备替代方案来进行监控和评估(Ge & Land, 2004)。

6.总结与展望

随着人工智能教育的深入发展,生成式人工智能近年来受到学界越来越多的关注。本研究探讨了基于 GPA 的问题提示支架在编程教育中的应用效果,发现其能够显著促进学生的计算思维与知识测试成绩,并能够激发学生产生更多元认知知识与元认知调节行为。总体而言,研究为基于 GPA 的问题提示支架与学生学习的融合应用提供了证据参考。但也还存在一定局限性,如样本量较少、实验周期短、学习者的年龄和专业分布单一等。未来的研究将进一步突破上述这些限制,创新 GPA 在教学模式、策略等方面的融合应用,并从长周期、多视角对其应用效果进行分析。

参考文献

- 戴岭,赵晓伟, & 祝智庭. (2023). 智慧问学:基于 ChatGPT 的对话式学习新模式. *开放教育研究*, 29(06), 42-51+111. <https://doi.org/10.13966/j.cnki.kfjyyj.2023.06.005>
- 张渝江,戴海军,罗太亮, & 兰勇. (2024). 生成式教学智能体的创建策略、角色与应用. *中小学信息技术教育*(10), 8-10.
- Abdul-Rahman, S.-S., & Du Boulay, B. (2014). Learning programming via worked-examples: Relation of learning styles to cognitive load. *Computers in Human Behavior*, 30, 286-298.
- Chen, E., Wang, D., Xu, L., Cao, C., Fang, X., & Lin, J. (2024). A Systematic Review on Prompt Engineering in Large Language Models for K-12 STEM Education. *arXiv preprint arXiv:2410.11123*.
- Ge, X., & Land, S. (2004). A conceptual framework for scaffolding ill-structured problem solving processes using question prompts and peer interactions. *Educational Research Technology and Development*, 52(2), 1042-1629.
- Gong, X., Li, Z., & Qiao, A. (2024). Impact of generative AI dialogic feedback on different stages of programming problem solving. *Education and Information Technologies*, 1-21.
- Guo, B., Zhang, X., Wang, Z., Jiang, M., Nie, J., Ding, Y., Yue, J., & Wu, Y. (2023). How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.
- Keuning, T., & van Geel, M. (2021). Differentiated teaching with adaptive learning systems and teacher dashboards: the teacher still matters most. *IEEE transactions on learning technologies*, 14(2), 201-210.
- Korkmaz, Ö., Çakir, R., & Özden, M. Y. (2017). A validity and reliability study of the computational thinking scales (CTS). *Computers in Human Behavior*, 72, 558-569.
- Malik, S. I., Ashfque, M. W., Tawafak, R. M., Al-Farsi, G., Usmani, N. A., & Khudayer, B. H. (2022). A chatbot to facilitate student learning in a programming 1 course: A gendered analysis. *International Journal of Virtual and Personal Learning Environments (IJVPLE)*, 12(1), 1-20.
- Mathew, R., Malik, S. I., & Tawafak, R. M. (2019). Teaching Problem Solving Skills using an Educational Game in a Computer Programming Course. *Informatics in education*, 18(2), 359-373.
- Molenaar, I., Van Boxtel, C. A., & Sleegers, P. J. (2010). The effects of scaffolding metacognitive activities in small groups. *Computers in Human Behavior*, 26(6), 1727-1738.
- Muñoz-Carril, P.-C., Hernández-Sellés, N., Fuentes-Abeledo, E.-J., & González-Sanmamed, M. (2021). Factors influencing students' perceived impact of learning and satisfaction in Computer Supported Collaborative Learning. *Computers & Education*, 174, 104310.

- Ng, C., & Fung, Y. (2024). Educational Personalized Learning Path Planning with Large Language Models. *arXiv preprint arXiv:2407.11773*.
- Rehman, N., Zhang, W., Mahmood, A., Fareed, M. Z., & Batool, S. (2023). Fostering twenty-first century skills among primary school students through math project-based learning. *Humanities and Social Sciences Communications*, 10(1), 424. <https://doi.org/10.1057/s41599-023-01914-5>
- Sijmken, E., De Cock, M., & De Laet, T. (2023). Scaffolding students' use of metacognitive activities using discipline-and topic-specific reflective prompts. *Metacognition and Learning*, 18(3), 811-843.
- Sperling, R. A., Howard, B. C., Miller, L. A., & Murphy, C. (2002). Measures of children's knowledge and regulation of cognition. *Contemporary educational psychology*, 27(1), 51-79.
- Sun, L., Hu, L., & Zhou, D. (2021). Which way of design programming activities is more effective to promote K-12 students' computational thinking skills? A meta-analysis. *Journal of Computer Assisted Learning*, 37(4), 1048-1062.
- Tassoti, S. (2024). Assessment of Students Use of Generative Artificial Intelligence: Prompting Strategies and Prompt Engineering in Chemistry Education. *Journal of Chemical Education*.
- Tom, M. (2015). Five C Framework: A student-centered approach for teaching programming courses to students with diverse disciplinary background.
- White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., & Schmidt, D. C. (2023). A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*.
- Xie, K., & Bradshaw, A. C. (2008). Using question prompts to support ill-structured problem solving in online peer collaborations. *International Journal of Technology in Teaching and Learning*, 4(2), 148-165.
- Yilmaz, R., & Yilmaz, F. G. K. (2023). The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. *Computers and Education: Artificial Intelligence*, 4, 100147.
- Yusuf, A., Pervin, N., & Román-González, M. (2024). Generative AI and the future of higher education: a threat to academic integrity or reformation? Evidence from multicultural perspectives. *International Journal of Educational Technology in Higher Education*, 21(1), 21.