

编程教育能给中小學生带来什么影响？——一项基于境外 78 篇相关研究的系统性文献综述

What Impact Does Programming Education Have on K-12 Students? — A Systematic Literature Review Based on 78 Relevant Overseas Studies

刘敏¹, 宿庆^{2*}

¹⁻² 西北师范大学教育技术学院（智能教育学院）
suqing8909@nwnu.edu.cn

【摘要】 随着教育数字化转型，编程教育已成为培养中小學生核心素养的关键途径，但其多元影响与学段适配规律仍待厘清。本文系统综述 2015—2025 年间 78 项实证研究，探讨编程教育的影响、学段差异及作用机制。研究发现，编程教育在认知、心理、社交与跨学科迁移等方面均产生积极影响；学段差异显著：小学重在兴趣启蒙与基础思维，初中侧重结构化思维与跨学科应用，高中聚焦高阶逻辑与创新。其中，工具类型、教学方法与干预时长是关键调节因素。未来需扩大样本代表性、完善评估体系并深化机制研究，以推动编程教育高质量发展。

【关键词】 编程教育；中小學生；系统性文献综述；计算思维

Abstract: Amid education's digital transformation, programming has emerged as vital for cultivating K-12 students' core competencies. This systematic review of 78 empirical studies (2015-2025) reveals its positive impacts on cognitive, psychological, social, and cross-disciplinary learning, with distinct stage-specific focuses: interest and foundational thinking in primary, structured application in junior secondary, and higher-order innovation in senior secondary. Tool type, pedagogy, and intervention duration are key moderators. Future work should expand sample diversity, improve assessments, and deepen mechanistic research to advance programming education quality.

Keywords: Programming education, K-12 students, Systematic literature review, Computational thinking

1. 引言

数字时代下编程教育已成为全球基础教育改革的重要内容，成为中小學生计算思维、创新能力等核心素养培育的关键载体，国内外学界围绕其教育影响展开了多维度实证探索。但现有研究仍存在研究结论分散、体系化不足的问题：既有成果多聚焦单一维度影响或特定学段研究，缺乏对编程教育多元育人价值的系统整合，且对学段适配规律、教育效果的内在作用机制探讨尚不深入，尚未形成统一的研究框架与结论体系。因此，本研究旨在系统梳理 2015-2025 年境外编程教育相关实证研究，重点探索以下问题：(1)编程教育对中小學生的具体影响有哪些？(2)编程教育对不同年龄段学生的影响差异体现在哪些方面？(3)这些影响产生的原因是什么？

2. 研究方法过程

(一) 2.1. 文献筛选

本研究遵循 PRISMA 标准，在 Web of Science、EBSCOERIC、Elsevier Science Direct、Wiley

Online Library、Taylor & Francis 五个主要的文献数据库中，以字符串("programming education" OR "coding education") AND (student OR "K-12") AND (impact OR effect)在数据库中进行检索，检索截止日期为 2025 年 12 月 31 日，共获得 3258 篇文章。经去重与两轮筛选，剔除不符合纳入标准的文献，最终纳入 78 篇相关研究为分析样本（筛选流程见图 1）。

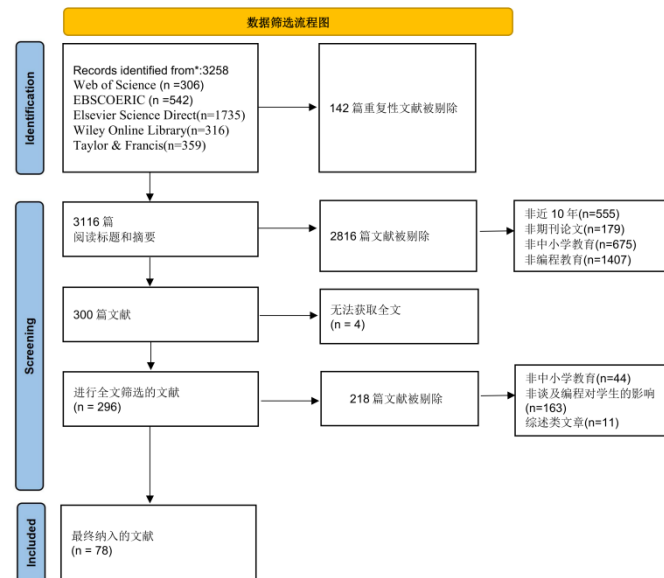


图 1 基于 PRISMA 的筛选纳入流程图

(二) 2.2. 文献编码

本研究采用内容分析法对纳入文献进行标准化编码，编码维度涵盖文献基本信息、学段、样本量、方法论、编程工具类型、影响维度、实验时长等，由 2 名研究者独立编码并对分歧文献协商达成一致，保障编码信度。其中编程工具类型包括软件类、硬件类、不插电类及混合类四大类别(Rich et al., 2024)，影响维度包括认知能力、心理、社交协作、其它四个维度(Gharbaoui et al., 2024; Yao & Lin, 2025)。

3. 研究结果

(三) 3.1. 编程教育对中小学生的具体影响

(1) 认知能力维度

编程教育能有效提升学生的计算思维、逻辑推理、编程概念掌握及问题解决能力，其中计算思维（61.8%）与逻辑推理能力（55.3%）的提升效果最为突出。计算思维的培育涵盖算法思维、分解、抽象等核心技能，适用于所有编程类型与学段；逻辑推理能力则在程序设计、调试的因果分析与序列梳理中不断强化，且能迁移至其他学科学习。学生在实操中逐步理解各类核心编程概念，在完整的编程实践过程中形成结构化的问题解决方法，这类能力的提升也成为学生核心认知素养发展的重要支撑。

(2) 心理维度

编程教育主要激发学生学习动机、提升编程自我效能感。学习动机的提升源于内在动机激活与任务价值感知：通过游戏化设计、项目式开发让学生因兴趣主动参与，结合真实情境与学科联结让学生感知编程的实用价值。自我效能感则通过阶梯式任务累积成功体验、硬件操作与可视化编程的即时反馈得以强化，同时同伴互教互学、教师正向评价的社会反馈，进一步提升学生编程学习信心。

(3) 社交协作维度

现有编程教育多采用团队或小组合作的形式开展多人配合完成的编程任务，通过创设合作

学习场景，为学生提供了高频社交互动的机会，进而对学生团队沟通、分工协作及冲突解决能力产生影响。这种影响是通过结构化的任务设计和互动机制实现的，不同编程任务的协作需求强度直接影响社交技能的发展效果。团队沟通依托于任务依赖性得以增强，如在硬件编程等需要多人配合的任务中，学生不得不高频交流、共享认知，从而形成紧密的信息互动。在分工协作方面，通过角色轮换与责任绑定等策略，逐步培养学生的角色意识。在冲突解决上，借助客观排查问题与开展错误复盘会，引导学生形成“对事不对人”的处理态度。

(4) 其它维度

编程教育与数学、科学等学科存在显著协同效应，这种效应主要通过逻辑思维的共通性、问题解决方法的迁移性以及场景应用的关联性实现。在数学领域，编程任务中的逻辑推理与数学问题解决形成深度联动；在科学领域结合传感器操作与实验模拟，强化学生科学探究能力。对于特殊群体而言，不插电编程能够帮助 14 至 18 岁轻度智力障碍学生提升序列思维、因果判断能力与社交参与度。

(四) 3.2. 编程教育对不同年龄段学生的影响差异

编程教育对中小学生的影响呈现显著学段差异，与各阶段学生的认知发展水平、学习需求及能力基础高度适配，整体呈兴趣启蒙、能力拓展、高阶深化的进阶特征。

小学是编程教育的入门适应期，核心围绕兴趣唤醒与基础思维具象化培育展开。认知方面以建立基础编程概念、培养序列思维等初级计算思维为主，尚未涉及抽象逻辑的深度训练；心理层面以激发学习动机为核心，多采用游戏化教学降低入门门槛；社交协作以简单分工为主，学生初步掌握团队沟通规则，冲突解决能力仍需引导。

初中是编程教育的能力拓展期，核心转向抽象思维的结构化深化与复杂任务的项目式落地。认知层面在基础思维之上，强化逻辑思维流程优化、问题模块化拆解等能力，同时注重跨学科应用与基础调试能力的培养；心理层面，学习动机更受任务挑战性与实用性驱动，自我效能感与任务完成质量关联更为紧密，复杂任务带来的成就感反馈更加显著；社交协作方面，能够完成较为复杂的分工，自主尝试合理解决冲突，协作模式的科学性直接影响学习效果，是小学协作能力的进一步进阶。

高中是编程教育的能力成熟期，核心聚焦高阶逻辑的锤炼与个性化创新能力的专业化拓展。认知层面侧重高阶计算思维的培养，关注算法效率优化与抽象编程范式的理解，同时强化自主项目开发与复杂问题的创新性解决能力；心理层面，学习动机呈现个性化与目标导向特征，有专业发展预期的学生学习动机更为持久，无基础学生则多依托学科关联需求形成学习动力；社交协作以高效分工与成果导向为核心，团队沟通的精准度与任务完成效率呈正相关，协作的专业性与高效性显著提升。

(五) 3.3. 编程教育影响产生的关键原因

编程教育的多维积极影响，核心由编程工具类型、教学方法、实验时长三大关键因素调控，其与学生认知水平及学段特征的适配程度，直接决定了教育效果的强弱与持续性。

在编程工具的类型上，不同工具的技术属性与交互形式决定了教育作用的路径，可精准匹配不同学段与群体的需求。软件类自包含环境编程适用于小学至初中低年级，侧重基础思维培养；开发工具包则适配高中阶段，有助于深化高阶逻辑能力。硬件类工具适合小学高年级至初中，能够同步强化认知能力与社交协作能力。不插电类工具适用于低龄学生与特殊群体，也适合资源有限地区，兼顾教育公平与基础逻辑培育。

在教学方法上，结合学段与工具特性对教学方法进行选择，可最大化教育价值。如项目式教学覆盖小学高年级至高中，通过完整项目开发培养学生的模块化思维与跨学科应用能力；游戏化教学适配小学至初中低年级，借助趣味化设计激发内在动机，强化编程自我效能感，且可灵活结合各类编程工具。

在实验时长上，短期实验(少于 4 周)仅能实现基础能力的提升与即时心理效应，难以促成对编程概念的深入理解；长期实验(超过 8 周)有助于推动多维度能力的深度发展，但过长则易

引发学习疲劳。应根据学段与能力培养目标精准设计实验时长，以保障教育效果的深度与长效性。

4. 结论与展望

本研究基于 78 篇境外实证文献的系统分析发现，编程教育对中小学生的认知、心理、社交协作及跨学科迁移均呈积极影响，且存在显著学段差异：小学阶段重在兴趣激发与基础计算思维培养，初中阶段强化结构化思维与跨学科应用能力提升，高中阶段聚焦高阶逻辑思维与个性化创新能力形成。编程工具类型、教学方法、干预时长是影响教育效果的核心调节因素，相关结论可为中小学编程教育的课程设计、教学实施与效果评估提供实证参考。

现有研究仍存在明显局限：一是样本群体覆盖不均衡，研究多集中于小学阶段，初高中阶段研究相对不足，特殊群体研究占比偏低，部分小样本研究的结论有待进一步验证；二是评估体系尚不完善，依赖主观量表开展测评，缺乏客观化、标准化的评估工具，且研究多关注短期认知效果，对学生能力的长期迁移与非认知维度发展的关注不足；三是研究设计与机制分析深度有待加强，多数研究未设置对照组，随机分组设计占比较低，对编程教育的内在作用机制及各类干预策略的适用边界探究尚不充分。

针对上述问题，未来研究需从三方面重点突破：一是扩大研究样本的多样性，拓宽编程教育的研究学段，重点补足初高中阶段的研究空白。同时关注农村、教育资源薄弱地区及特殊需要学生群体，提升研究结论的普适性；二是构建客观化、多维度、标准化的编程教育效果评估体系，融合学习过程数据、实操行为表现与标准化测评方式，增强不同研究结果之间的可比性；三是深化编程教育的作用机制分析，系统解析其影响学生发展的内在路径、关键条件与适用边界，明确各类编程工具、教学方法、干预时长在不同学段的适配规律，为中小学开展精准化、差异化的编程教学实践提供坚实的理论支撑。

参考文献

- Gharbaoui, H., Mansouri, K., & Poirier, F. (2024). Improving Student Engagement and Success in Computer Programming Courses through Social Learning in Online Environments. *International Journal of Engineering Pedagogy (iJEP)*, 14(6), 54–68. <https://doi.org/10.3991/ijep.v14i6.48705>.
- Rich, P. J., Bartholomew, S., Daniel, D., Dinsmoor, K., Nielsen, M., Reynolds, C., Swanson, M., Winward, E., & Yauney, J. (2022). Trends in tools used to teach computational thinking through elementary coding. *Journal of Research on Technology in Education*, 56(3), 269–290. <https://doi.org/10.1080/15391523.2022.2121345>.
- Yao, D., & Lin, J. (2025). Cognitive enhancement through competency-based programming education: a 12-year longitudinal study. *Education and Information Technologies*, 30(14), 20347–20383. <https://doi.org/10.1007/s10639-025-13582-w>.