Journal of Research in Mathematics, Science, and Technology Education
ISSN 3047-5163 (online) & 3047-6933 (print)
July 2026, Vol. 3, No. 3, pp. 160–169

Review Article

A Systematic Review of Instructional Pedagogies for Teaching Computer Education Among Secondary School Learners in Uganda

Pison Asiimwe[1], Justine Momanyi Omare[2], Frank Murangira[3]

[1]Department of Open Learning and Computer, Kabale University, Kampala, Uganda
[2]Department of Foundations, Kabale University, Kampala, Uganda
[3]Department of Science Education, Kabale University, Kampala, Uganda

Abstract
Computer Education has become part of modern curricula because of its impact on the development of digital literacy, problem-solving skills, and innovation. Curriculum reforms such as the Competency-Based Curriculum in Uganda have emphasized these competencies through learner-centered methods. However, successful implementation is still limited by the infrastructural factors, the traditional pedagogical methodology, and the lack of proper teacher preparation. The purpose of the study was to find out the most common pedagogical strategies in computer education worldwide, assess their efficiency, and discuss their applicability in education. A systematic literature review was conducted by using PRISMA 2020. ERIC, Scopus, JSTOR, ProQuest, Springer, Sage, Wiley, and Taylor and Francis were used to select 23 peer-reviewed publications published between 2010 and 2025. Constructivist theory, competency-based education, and critical thinking provided the basis of the review. The research involved a content analysis approach, which synthesized evidence on the strategies used in pedagogy, research designs, and learning outcomes. The results show that learner-centered pedagogies, specifically problem-based learning, project-based learning, computational thinking, flipped instruction, and collaborative learning, are most commonly applied and show a significant benefit in terms of increasing the engagement of learners, their critical thinking skills, their problem-solving capabilities, and their computational competence. These strategies are highly aligned with the main competencies highlighted in the competency-based curriculum. Nonetheless, they are hindered by contextual problems, such as the lack of ICT infrastructure, access to technological resources, and a lack of teacher training. The paper concludes that although globally accepted pedagogical strategies are very applicable to competency-based education structures, they require contextualization in order to be effectively implemented. Initiatives should be focused on enhancing teacher capacity, ICT infrastructure, and endorsing context-sensitive instructional practices to be implemented effectively in education environments characterized by resource constraints.

Keywords: Computer Technology Education, Teaching Pedagogy, Competency-Based Curriculum

Correspondence
Pison Asiimwe
pasiimwe@kab.ac.ug

Received
March 18, 2026
Accepted
June 21, 2026
Published
July 1, 2026

Citation: Asiimwe, P., Omare, J. M., & Murangira, F. (2026). A systematic review of instructional pedagogies for teaching computer education among secondary school learners in Uganda. *Journal of Research in Mathematics, Science, and Technology Education, 3*(3), 160–169.

DOI: 10.70232/jrmste.v3i3.76

1. INTRODUCTION

1.1. Introduction to the Problem

Computer education has become an essential component of school curricula globally due to its role in technological innovation, employability, and foundational problem-solving skills (Guzdial, 2016). Numerous nations have integrated computing, ICT, or computer science into primary and secondary

education, emphasizing programming, computational thinking (CT), and digital literacy (Lye & Koh, 2014). The effectiveness of computer education is closely linked to educational pedagogies (Yadav et al., 2016).

In East Africa, major curriculum reforms have emphasized learner-centred pedagogies, including Kenya's Competency Based Curriculum (CBC), Tanzania's ICT-enhanced curriculum, Rwanda's digital literacy integration, and Uganda's CBC launched in 2020. Uganda's CBC focuses on developing core competencies such as creative thinking, problem solving, communication, self-directed learning, and digital competence, which are highly aligned with global computer/ICT education pedagogies. Despite reforms, computer/ICT education in East Africa faces several challenges, including insufficient ICT infrastructure, traditional teaching practices, inadequate teacher digital skills, and limited access to updated pedagogical models (Mtebe & Raisamo, 2014). There is a growing need to identify pedagogical strategies that are both globally validated and locally adaptable.

Though the body of research on computer education pedagogy in East Africa is fragmented, with few systematic reviews that synthesise global pedagogies and evaluate their fit with Uganda's CBC. This study addresses this gap by conducting a PRISMA-guided systematic review of globally recognized computer/ICT education pedagogies sourced exclusively from high-quality academic databases.

1.2. Conceptual Framework

This review is informed by three theoretical perspectives: constructivism, competency-based education, and computational thinking.

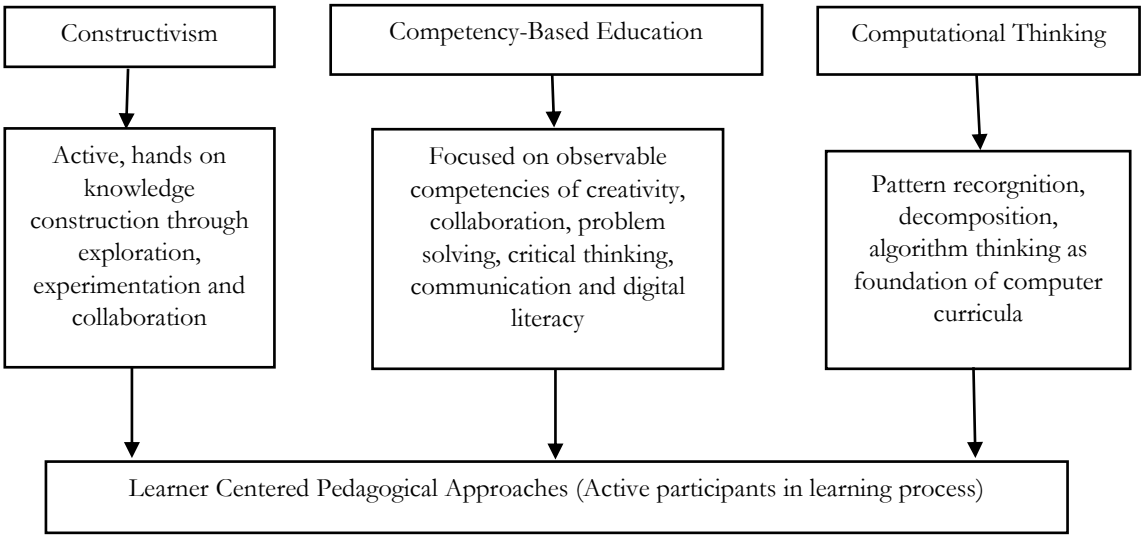

Figure 1. Theoretical Perspectives and Computer Education Pedagogy

Figure 1 illustrates the interconnection between three theoretical perspectives of constructivism, competency-based education, and computational thinking. For constructivism, computer education thrives on active, hands-on construction of knowledge through exploration, experimentation, and collaborative engagement (Papert, 1980). Based on competency theory, Uganda's CBC emphasizes observable competencies such as creativity, collaboration, problem solving, critical thinking, communication, and digital literacy, whereas computational thinking theory looks at the decomposition, abstraction, pattern recognition, and algorithmic thinking that serve as foundational skills in computing curricula (Grover & Pea, 2013). Together, these theories support pedagogical approaches that are centred on learners as active participants.

1.3. Research Questions

a) What instructional pedagogies are widely used in global computer education?
b) What evidence exists regarding the effectiveness of these pedagogies?

c) How can these pedagogies be adapted to computer education within developing countries, particularly in Uganda's CBC?

2. METHODOLOGY

This study uses a systematic literature review guided by PRISMA 2020 standards due to its standardized framework that ensures transparency, structural rigor, and reproducibility in systematic reviews, which facilitates clear reporting.

2.1. Temporal Scope

The research study analyzed articles from peer-reviewed academic journals that were published during the period from 2010 to 2025 to show how computing education and technological advancements have developed from their initial stage up to their present state. The time period experienced a rapid increase in learner-centered teaching methods, which included problem-based learning, project-based learning, flipped instruction, and collaborative learning in computing education. The year 2025 was the upper limit to make sure that the review considers the latest empirical and conceptual research available and to represent modern trends in computing education.

2.2. Literature Search Period and Procedure

This review was done in the period between September and November 2025. A systematic search was conducted in the selected databases because they have a wide coverage of peer-reviewed articles in computing education and instructional technology. A systematic search strategy was utilized with pre-determined keywords that were connected to computational thinking and programming education, and teaching methods like problem-based learning, project-based learning, flipped learning, and collaborative learning. Titles, abstracts, and keywords were used to apply the search terms to guarantee that all relevant studies were retrieved. The records were then filtered based on the preset inclusion and exclusion criteria to obtain studies that were useful.

2.2.1. Inclusion Criteria

We searched for articles in peer-reviewed journals, conference papers, and graduate theses that are indexed in ERIC, Scopus, ProQuest, JSTOR, Wiley, Springer, Sage, or Taylor & Francis. We considered articles that were published between 2010 and 2025, focused on computer education, ICT education, computer science pedagogy, and critical thinking pedagogies, and published in English in full text.

2.2.2. Exclusion Criteria

We eliminated studies not related to pedagogy and computer/ICT education, opinion pieces without empirical data, and studies outside ERIC, Scopus, ProQuest, JSTOR, Wiley, Springer, Sage, or Taylor & Francis databases.

2.3. Scanning

Search terms included: "computer education pedagogy", "computer science teaching methods", "ICT pedagogy", "computational thinking instruction", "project-based learning programming", "problem-based learning computing." Databases searched were: ERIC, Scopus, JSTOR, ProQuest, Springer, Sage, Wiley Online Library, and Taylor & Francis Online.

2.4. Justification of Databases and Search Scope

The databases used in this study were selected because they provide strong coverage of educational research, computing education, and interdisciplinary studies. For example, ERIC was prioritized for its focus on educational practices and pedagogy, ProQuest and JSTOR provide access to peer-reviewed academic research across disciplines, and Wiley and Taylor & Francis were selected due to their strong publication record. These databases were preferred over Scopus and Web of Science due to institutional access limitations and their strong relevance to education-focused research.

2.5. Description of Search Process

The research was conducted in a systematic review process based on PRISMA principles. The first search through a few databases revealed ninety-six articles. After removing duplicates (eighteen articles), seventy-eight articles remained for screening. During title and abstract screening, forty-two articles were excluded due to a lack of relevance to computing pedagogy. The remaining thirty-six articles were assessed for full-text eligibility. Out of these, nine articles were excluded due to a lack of focus on computing education, missing empirical or conceptual relevance, and general deviation from the CBC pedagogical focus. This resulted in a final sample of 23 studies included in the review.

2.6. Synthesized Themes

The analysis of the 23 included studies revealed five major interrelated themes: pedagogical approaches, sample characteristics, research designs, learning outcomes, and database distribution. A thematic synthesis was undertaken to identify dominant patterns, methodological tendencies, and emerging directions within computing education research.

2.7. Data Analysis

A content analysis approach was used to code: Pedagogical approach, sample characteristics, research design, year of publication, title, and database source. Themes were synthesized to identify patterns across studies.

2.8. Limitations

The review was limited to English publications. Some of the relevant East African published articles were unindexed in the target databases. Variation in research methodologies limited comparability, and infrastructure challenges in developing countries are not fully captured by global studies.

2.9. Ethical Statement

This study uses only secondary data from publicly accessible sources; no ethical approval is required.

3. FINDINGS

A total of 23 studies were included in this study as a result of screening with inclusion and exclusion criteria. The selected studies are presented in Table 1.

Table 1. Characteristics of Included Studies

Author(s)	Year	Title	Method	Sample	Pedagogical Focus	Database
Grover & Pea	2013	Computational Thinking in K-12	Review	N/A	CT frameworks	Wiley
Lye & Koh	2014	Teaching CT Through Programming	Systematic Review	N/A	CT & programming pedagogy	ERIC

Author(s)	Year	Title	Method	Sample	Pedagogical Focus	Database
Guzdial	2016	Learner-Centered Computing	Conceptual Analysis	N/A	Constructivist computing	Springer
Yadav, Hong & Stephenson	2016	Computational Thinking in Teacher Education	Mixed Methods	325 teachers	CT integration	ERIC
Sentance & Waite	2018	Pedagogy for Computing Education	Review	N/A	Computing pedagogy models	Wiley
Sentence & Csizmadia	2017	Barriers in Computer Science Education	Mixed Methods	54 teachers	Barriers to pedagogy	Springer
Arfe, Vardanega & Ronconi	2020	Collaborative Programming in CS	Experimental	65 students	Pair programming	Wiley
Blau & Shamir-Inbal	2017	Flipped Computer Science Classrooms	Experimental	102 students	Flipped learning	Taylor & Francis
Kong & Lai	2023	Project-Based Learning in ICT	Experimental	112 students	PBL	Wiley
Csizmadia, Curzon, Dorling, Humphrey, Ng, Selby & Woollard	2015	Developing CT Through Programming Tasks	Qualitative	28 teachers	CT task design	Wiley
Kafai & Burke	2014	Connected Code: Engaging Youth in Programming	Mixed	45 learners	Constructionist programming	Wiley
Brennan & Resnick	2012	New Framework for CT	Conceptual	N/A	CT assessment framework	JSTOR
Repenning, Webb & Ioannidou	2010	Computational Participation	Conceptual/Case Studies	N/A	Design-based computing	Wiley
Hsu, Chang & Hung	2018	How to learn and how to teach CT	Experimental	64 upper primary	Block-based learning	Scopus
Thota & Negreiros	2015	Problem-Based Learning in CS	Design-Based Research	52 undergraduates	PBL in CS	Taylor & Francis
Seiter & Foreman	2013	Assessing CT in Scratch	Experimental	40 learners	CT assessment	Sage
Wang, Liu & Li	2021	Impact of Peer Instruction in CS	Experimental	98 undergraduates	Peer instruction	Wiley
Vivian, Falkner & Falkner	2014	Success Factors in CS Learning	Mixed	200 students	Engagement-focused pedagogy	JSTOR
Hermans & Aivaloglou	2017	Teaching Programming to Children	Review	N/A	Block-based pedagogy	Wiley
Qian & Lehman	2017	Programming and CT: A Meta-Analysis	Meta-analysis	N/A	CT skill development	Taylor & Francis
Grover, Pea & Cooper	2016	CT in Middle Schools	Mixed	124 students	CT curricular design	Wiley
Tew & Guzdial	2011	How Students Understand CS Concepts	Qualitative	42 students	Conceptual understanding	ProQuest
Hamidi	2025	CT as 21st Century Skill	Conceptual	N/A	CT theory & pedagogy	Sage

3.1. Synthesis of Study Characteristics

The reviewed studies show several patterns:

Sample sizes ranged from small qualitative samples (28 teachers) to larger quantitative studies (over 300 participants). Geographically, most studies were conducted in developed countries, with limited representation from Africa.

Learner groups included primary, secondary, and undergraduate students, as well as teachers. Experimental and mixed-method designs were the most dominant, reflecting a focus on measuring the effectiveness of pedagogical interventions.

3.2. Pedagogical Approaches

Problem-Based Learning (PBL); We found out that this pedagogy increases conceptual understanding of algorithms, debugging, and troubleshooting skills (Blau & Shamir-Inbal, 2017).

Project-Based Learning (PrBL); its benefits to learners include creativity, collaboration, and practical ICT competence (Kong & Lai, 2023).

Computational Thinking Pedagogy, which widely recognized pedagogy on the globe, is relevant to a developing country like Uganda, which is "young" in Competency Based Curriculum (Grover & Pea, 2013; Lye & Koh, 2014).

Flipped Classrooms was found to the effective pedagogy for programming and other computer-related studies (Blau & Shamir-Inbal, 2017).

Collaborative Learning (pair programming); This improves confidence, debugging skills, and engagement of learners (Arfe et al., 2020).

3.3. Effectiveness of Pedagogical Approaches

Evidence shows that Critical Thinking instruction improves problem-solving and algorithmic thinking (Yadav et al., 2016), PBL increases engagement and deep learning, and Flipped learning improves achievement in programming (Blau & Shamir-Inbal, 2017). Project-based learning leads to higher ICT performance (Kong & Lai, 2023), and Collaborative programming improves retention (Arfe et al., 2020).

3.4. Adaptation to Uganda's CBC

Uganda's CBC aligns closely with global pedagogical evidence because it emphasizes: critical thinking (CT), creativity (project-based tasks), collaboration (pair programming), communication (flipped and blended learning), and digital literacy (ICT-enhanced learning)

Table 2. Mapping Global Pedagogical Evidence to Uganda's CBC: Opportunities & Adaptation Strategies

Pedagogical Approach	Relevance to CBC Competencies	Potential Adaptation Strategies	Key Implementation Challenges
Computational Thinking / Programming	Supports critical thinking, problem-solving, digital competence	Use block-based / unplugged CT activities; integrate algorithm design and logic tasks	Teacher familiarity, limited devices, assessment alignment
Block-based / Visual Programming (Scratch)	Digital literacy, creativity, self-directed learning	Low-cost computing platforms; pair programming; integrate CT tasks in other subjects	Device access, electricity, teacher training
Problem- / Project-Based Learning	Collaboration, creativity, real-world problem-solving	Local projects (community, school, agriculture); integrate ICT	Planning, resources, teacher facilitation, assessment redesign
Blended / Flipped Learning	Self-directed learning, communication, digital competence	Mobile-based / offline content; group sessions	Infrastructure, internet, equitable access
Collaborative / Peer Learning	Collaboration, teamwork	Structured pair/group tasks; peer instruction; shared devices	Class size, device limitations, learner management

We noted that global pedagogical approaches map strongly onto Uganda's CBC competencies. However, adaptation requires contextual considerations such as: low-cost computers and related devices, unplugged CT activities, mobile and first learning resources, and teacher digital pedagogy training, echoing challenges reported globally (Grover & Pea, 2013; Lye & Koh, 2014).

4. DISCUSSION

The studies show strong global evidence supporting learner-centred pedagogies in computer education, which indicate a clear shift toward learner-centred teaching strategies in developing countries that are implementing CBC.

The effectiveness of these approaches is considered by how they enthusiastically engage learners in the learning process. As an illustration, Problem-Based Learning (PBL) and Project-Based Learning (PrBL) enhance learning, as they foster active learning, encourage deeper learning due to real-life application, and help students relate theory to practice. These approaches align with constructivist learning theory (Kong & Lai, 2023; Yağcı, 2018), which suggests that learners understand concepts better when they actively construct knowledge rather than passively receive it.

Likewise, Critical Thinking (CT) improves problem-solving since it divides complex problems into simple, manageable steps and makes it easier for learners to grasp the concepts required of them. CT also strengthens logical reasoning, which is essential for algorithm design and debugging.

The collaboration between learners, like pair learners in the lesson on the skill of coding, enhances performance, as learners share ideas, learn through each other, and develop confidence (Arfe et al., 2020). Social interaction in learning has been shown to reinforce understanding and reduce anxiety, especially in programming tasks.

Conversely, the lack of infrastructure, unavailability of devices, and insufficient training of teachers are some of the challenges facing these approaches in Uganda. These contextual barriers explain why implementation remains uneven despite strong theoretical alignment.

5. CONCLUSIONS AND RECOMMENDATIONS

The research review based on PRISMA concludes that the global computing education pedagogies (CT, PBL, PrBL, flipped learning, and collaborative learning) have strong potential in developing countries, as well as in CBC in Uganda. Nevertheless, contextual barriers should be considered to achieve the maximum of these benefits.

5.1. Practical Implications

This paper has a number of practical implications for the following key stakeholders.

5.1.1. Teachers

Learner-centered teaching methods should be embraced by teachers while engaging students in the learning process. They should utilize real-life activities, problem-solving, and group learning to encourage the acquisition of practical skills as opposed to concentrating on theoretical skills.

5.1.2. Teacher Training Institutions

The teacher training institutions need to change their programmes to incorporate computational. modern teaching methods and thinking. Both pre-service and in-service training should focus on the real-life implementation of innovative teaching techniques, including problem-based and project-based learning.

5.1.3. Curriculum Developers

Curriculum developers ought to create flexible and competency-based curricula that can support. project-based learning, inquiry-based learning, and collaborative learning. The curriculum should promote the combination of knowledge, skills, and attitudes to make sure that learners are holistically developed.

5.1.4. Policymakers

The policy makers are encouraged to invest in ICT infrastructure, such as a reliable internet connection, sustainable computer resources, and proper maintenance systems.

5.1.5. Researchers

Researchers ought to concentrate on context-specific studies, particularly in developing nations respond to local education needs.

5.2. Recommendations

We advise that an intensive teacher computer pedagogy training should be embarked upon, the use of low-cost digital materials, the introduction of unplugged CT games, and the redesign of formats of assessment to demonstrate CBC competencies. Also, the integration of collaborative problem-solving tasks in teaching and learning processes.

5.3. Limitations of the Study

This study has several limitations:

a) It included only studies published in English, which may exclude relevant research in other languages
b) The review relied on selected databases (ERIC, ProQuest, JSTOR, Wiley, Taylor & Francis), which may not cover all relevant studies
c) Limited access to required studies in the popular databases of Scopus and Web of Science
d) The review focused on published studies, potentially excluding grey literature
e) Variation in study designs and sample sizes may affect generalizability

5.4. Areas for Future Research

Based on these limitations, future research should:

a) Include studies from a wider range of databases, such as Scopus and Web of Science
b) Explore non-English publications to provide a more global perspective
c) Investigate long-term effects of critical and computational thinking
d) Examine practical implementation challenges in real classrooms

Acknowledgments. The authors express gratitude to all individuals who helped them in this study. Even though they are not listed as co-authors, their helpful guidance, constructive feedback, encouragement, and access to useful academic materials made a real difference in completing this work.

Research Ethics. No ethical approval was required since the study was a systematic review. All sources used were appropriately acknowledged and cited in accordance with academic standards.

Data Availability Statement. All data supporting the findings and conclusions are available upon request from the corresponding author.

Conflicts of Interest. No conflict of interest is declared by the authors

Funding. There was no external funding for this study.

REFERENCES

Arfe, B., Vardanega, T., & Ronconi, L. (2020). The effects of coding on children's planning and inhibition skills. *Computers & Education*, *148*, Article 103807. https://doi.org/10.1016/j.compedu.2020.103807

Blau, I., & Shamir-Inbal, T. (2017). Re-designed flipped learning model in an academic course: The role of co-creation and co-regulation. *Computers & Education, 115*, 69–81. https://doi.org/10.1016/j.compedu.2017.07.014

Brennan, K., & Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking.* Paper presented at the American Educational Research Association (AERA) Annual Meeting, Vancouver, Canada. https://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). *Computational thinking: A guide for teachers*. Retrieved from https://eprints.soton.ac.uk/424545/

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43. https://doi.org/10.3102/0013189X12463051

Grover, S., Pea, R., & Cooper, S. (2015). Designing for deeper learning in a blended computer science course for middle school students. *Computer Science Education, 25*(2), 199–237. https://doi.org/10.1080/08993408.2015.1033142

Guzdial, M. (2015). *Learner-centered design of computing education: Research on computing for everyone. Synthesis lectures on human-centered informatics*. Springer. https://doi.org/10.2200/S00684ED1V01Y201511HCI033

Hamidi, A. (2025). Advancing computational thinking education: Insights from systems thinking applications. *Human Systems Management*, *44*(1), 157-172. https://doi.org/10.3233/HSM-240024

Hermans, F., & Aivaloglou, E. (2017). To Scratch or not to Scratch?! A controlled experiment comparing plugged first and unplugged first programming lessons. In *Proceedings of the 12th Workshop on Primary and Secondary Computing Education* (pp. 49–56). Association for Computing Machinery. https://doi.org/10.1145/3137065.3137072

Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education, 126*, 296–310. https://doi.org/10.1016/j.compedu.2018.07.004

Kafai, Y. B., & Burke, Q. (2014). *Connected code: Engaging youth in programming through constructionist learning*. MIT Press. https://mitpress.mit.edu/9780262529679/connected-code/

Kong, S. C., & Lai, M. (2023). Effects of a teacher development program on teachers' knowledge and collaborative engagement, and students' achievement in computational thinking concepts. *British Journal of Educational Technology, 54*(2), 489–512. https://doi.org/10.1111/bjet.13256

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior, 41*, 51–61. https://doi.org/10.1016/j.chb.2014.09.012

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Qian, Y., & Lehman, J. (2017). Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education, 18*(1), 1–24. https://doi.org/10.1145/3077618

Repenning, A., Webb, D., & Ioannidou, A. (2010). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education* (pp. 265–269). Association for Computing Machinery. https://doi.org/10.1145/1734263.1734357

Seiter, L., & Foreman, B. (2013). Modeling the learning progressions of computational thinking of primary grade students. In *Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research (ICER '13)* (pp. 59–66). ACM. https://doi.org/10.1145/2493394.2493403

Sentance, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies, 22*(2), 469–495. https://doi.org/10.1007/s10639-016-9482-0

Tew, A. E., & Guzdial, M. (2011). *The FCS1:* A language independent assessment of CS1 knowledge. *In Proceedings of the 42nd ACM Technical Symposium on Computer Science Education* (pp. 111–116). https://doi.org/10.1145/1953163.1953200

Thota, N., & Negreiros, J. G. M. (2015). Introducing educational technologies to teachers: Experience report. *Journal of University Teaching & Learning Practice, 12*(1), Article 5. https://doi.org/10.53761/1.12.1.5

Vivian, R., Falkner, K., & Falkner, N. (2014). Addressing the challenges of a new digital technologies curriculum: MOOCs as a scalable solution for teacher professional development. *Research in Learning Technology, 22*, Article 24691. https://doi.org/10.3402/rlt.v22.24691

Wang, Y., Li, H., Feng, Y., Jiang, Y., & Liu, Y. (2012). Assessment of programming language learning based on peer code review model: Implementation and experience report. *Computers & Education, 59*(2), 412–422. https://doi.org/10.1016/j.compedu.2012.01.007

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends, 60*(6), 565–568. https://doi.org/10.1007/s11528-016-0087-7

Yağcı, M. (2018). Web-mediated problem-based learning and computer programming: Effects of study approach on academic achievement and attitude. *Journal of Educational Computing Research*, *56*(2), 272-292. https://doi.org/10.1177/0735633117706908