BILINGUAL PUBLISHING GROUP

Journal of Computer Science Research
https://journals.bilpubgroup.com/index.php/jcsr

REVIEW

Innovating Pedagogical Practices through Professional Development in Computer Science Education

Xiaoxue Du[1*], Ellen B Meier[2]

[1] *MIT Media Lab, MIT, Cambridge, MA 02139, USA*
[2] *Teachers College, Columbia University, New York City, NY 10027, USA*

ABSTRACT

Recent advancements in technology have opened up new avenues for educators to facilitate teaching and leverage more learning access in the digital age. As the demand for computational skills continues to grow in preparation for future careers, both teachers and students face the challenge of developing problem-solving, critical thinking, communication, and collaboration skills within an emerging digital landscape. Technology adoption, big data, cloud computing and artificial intelligence pose ongoing challenges for both teachers and students in adapting to the changing workforce development landscape. To tackle these challenges, the paper highlights the importance of exploring the implications of learning sciences in classroom teaching, developing a holistic vision for professional development in education, and understanding the complexities of teacher change. To effectively implement these components, it is crucial to adopt design approaches that prioritize student ownership in education and embrace the principles of inclusive education to reconceptualize the teaching practices in education and technology.

Keywords: Education; Computational thinking; Teacher education; Professional development; Design; Equity

1. Introduction

As computers continue to automate our routine and complex tasks, equity in technology access, content, and use becomes a key barrier to future learning opportunities. In particular, a recent emphasis on education requires the development of intellect, ethical judgment, societal understanding, and creativity [1]. The technological challenges raise the critical question of how to prepare teachers to face

*CORRESPONDING AUTHOR:
Xiaoxue Du, MIT Media Lab, MIT, Cambridge, MA 02139, USA; Email: xiaoxued@media.mit.edu

ARTICLE INFO
Received: 30 May 2023 | Revised: 9 July 2023 | Accepted: 13 July 2023 | Published Online: 20 July 2023
DOI: https://doi.org/10.30564/jcsr.v5i3.5757

CITATION
Du, X.X., Meier, E.B., 2023. Innovating Pedagogical Practices through Professional Development in Computer Science Education. Journal of Computer Science Research. 5(3): 46-56. DOI: https://doi.org/10.30564/jcsr.v5i3.5757

the unprecedented changes in the immediate future of education [2]. Teachers must acquire fundamental knowledge and adopt innovative teaching methods in order to effectively incorporate technology into their instruction and meet both the academic and social-emotional requirements of students in the realm of technology [3,4]. Drawing upon insights from the learning sciences and teacher education literature, technology possesses the capability to pave the way for groundbreaking teaching approaches within classrooms. By harnessing the power of technology, educators can unlock diverse learning opportunities that cater to the needs of all students, including those from diverse cultural and language backgrounds [5]. Emerging technologies could serve as cognitive tutors, peer learners, and conversational agents, in order to introduce students to novel methods of reflection, reasoning, and learning in their everyday lives [6,7].

The growing demands in computer science (CS) education among schools and educational entities have shown the need to strengthen students' knowledge and skills in problem-solving and analytical thinking [8,9]. Therefore, establishing meaningful pedagogical practices and fostering a culture of lifelong learning are crucial aspects when it comes to computer science education. The basic premise of the paper is that integrating CS is a complex process, which requires much more than simply "shoehorning" a new curriculum into the school day. Teachers need to cultivate the skills to design and establish a student-centered learning environment, purposefully enabling the effective integration of computer science education. Moreover, it is essential to foster a rigorous community space that encourages learners to make connections between their acquired knowledge and other areas within the computing field. This approach helps foster a sense of belonging to the wider computing community. In this paper, we will first synthesize key challenges and opportunities identified in computer science education. Then, we will introduce the key components of a research-based, systematic professional development approach to build teachers' capacity to design a student-centered learning environment. Finally, we will discuss the implications of strengthening teachers' professional development in the computing science education community.

2. Key challenges and opportunities

Implementing in school settings presents new challenges for educators for at least five key reasons. First, there is a lack of "shared meaning" [10] for computer science as an academic discipline in K-12 education. Teachers should strive to develop a shared comprehension of both content knowledge and effective pedagogical practices in order to seamlessly integrate them into their curriculum planning. Second, computational thinking is increasingly considered a foundational skill in the 21st century, but is not systematically addressed in the curriculum. It serves as a process for recognizing aspects of computation in the surroundings and introduces techniques from CS to understand both natural and artificial systems and processes [11]. Third, there is not a clear scope and sequence for standards in each grade, which creates challenges for educators interested in developing student learning plans to integrate CS across disciplines. Because of the lack of a scope and sequence, there is insufficient empirical evidence for student learning and a lack of clear assessment objectives to support content definition and sequencing. Fourth, teachers' professional development in computer science is a new process, which requires more empirical evidence and research to identify the core professional development content material and resources needed to prepare educators for designing student-centered learning experiences in education [12]. Lastly, recent research has emphasized the significance of nurturing young people's capacity to create through the acquisition of computing skills. The development of these skills holds substantial implications for their personal lives and the betterment of their communities. To ensure that students develop essential computing design skills, it is imperative to create an inclusive, motivating, and empowering learning environment. This could provide students with greater autonomy to code, break down complex problems, and apply their learning across various contexts to solve real-world problems. However, in many cases, the curriculum may focus exclusively on technical challenges and entry points, requiring students to have prior programming experience. This limits opportunities for students to engage critically with a broader curriculum and participate in a larger computing community. To address the challenge, it is important to create a wide and deep learning space for learners, allowing them to connect what they have learned to other computing and content fields and fostering a sense of belonging to the greater computing community [13].

One way to address the challenges faced by both educators and students in computer science education is to provide more professional development opportunities for educators that position teachers as designers and effectively integrate computational-oriented curriculum into daily learning and teaching. Teachers need the essential knowledge and skills to plan enriched lessons, select the most relevant user cases, and design curricula to develop students' skills in problem-solving, computational thinking, and critical awareness in education. In addition, in the design process, teachers can develop student-centered learning environments that allow students with different background knowledge to engage curriculum, develop their interests, and build confidence that empowers them to learn and grow in the computer science curriculum. Finally, more research should be conducted to develop high-quality professional development, which could, in turn, prepare a cohort of change leaders to innovate pedagogical practices in computer science beyond programming, while building local community networks to sustain the change and innovation in daily teaching.

3. Applying the innovating instruction model in education

The Innovating Instruction © model has been developed by the Center for Technology and School Change, Teachers College, Columbia University. The model is developed and built upon the theory of change, learning science, professional development theories, and the emerging capabilities of technology. In this model, teachers take on the role of designers collaborating with facilitators to co-design projects that can be implemented in their real school settings. Technology plays a pivotal role as a catalyst for driving pedagogical innovation and motivates teachers as changemakers to advocate and sustain change in classroom teaching [14,15]. (see **Table 1**, CTSC Professional Development model: Innovating Instruction model).

The model comprises three fundamental components: Design, Situate, and Lead, all aimed at assisting teachers in transforming their teaching and learning approaches. It is imperative for teachers to grasp effective teaching practices aligned with principles derived from the learning sciences. This understanding enables them to design environments that foster meaningful learning experiences and facilitate opportunities for students to deepen their understanding of the subject. Equipping teachers with the capacity to design curriculum goals, employ formative assessments, and engage diverse students in inquiry-based learning environments holds great promise in supporting their professional growth [15].

The Situate component plays a crucial role in customizing the learning experience for each teacher's classroom and their students. It not only showcases engaging pedagogical practices through a hands-on approach but also offers personalized support to teachers. Incorporating insights from the learning sciences, it establishes a foundation for comprehending the intricacies of learning and thinking. The science of learning and development (SoLD) approach has been utilized to expound upon the "whole child model", which underscores the necessity of addressing various aspects of students' academic, cognitive, ethical, physical, psychological, and socio-emotional well-being. Specifically, creating a supportive environment fosters strong relationships and a sense of community among students. The situated approach encourages teachers to position students as active "knowledge-builders" within an inquiry-based learning environment [16,17].

Finally, the Lead component of the model focuses on preparing teachers to become leaders and col-

Table 1. Innovating instruction model.

DESIGN—Engage teachers as designers of student-centered, authentic learning experiences	
1. Embrace a Design Approach	Model and support a backwards design approach to project planning that creates meaningful learning experiences for students.
2. Enrich Content Knowledge	Provide opportunities for deepening teachers' understanding of content, including cross-curricular connections, learning standards, and student misconceptions.
3. Integrate Assessment Practices	Facilitate the design of authentic assessment and data use to identify and respond to student needs.
4. Leverage Digital Tools	Teach the integration of digital tools as part of the design process to facilitate interactive student learning and to enrich content.
SITUATE—Provide learning experiences for teachers that respects them as professionals and adapts the learning for their particular school and situation	
1. Contextualize Teacher Learning	Situate the design work in the professional lives of teachers in order to connect deeply to the realities of teachers' classrooms and their students.
2. Model Effective Practice	Provide interactive, hands-on professional development that engages teachers and models project-based learning with available tools and resources.
3. Individualize Support	Co-construct project plans based on student and curricular needs, provide ongoing support for classroom implementation, and facilitate reflection on teaching and learning.
LEAD—Support leaders in guiding and sustaining change initiatives, while positioning teachers as agents of change	
1. Envision Change	Prioritize instructional leadership and develop actionable goals to promote change in self-identified areas of need.
2. Empower Leadership at All Levels	Provide a forum for identifying leaders--administrators, teachers, and community members--who can spearhead efforts that contribute to the common vision.
3. Sustain A Culture for Innovation	Scaffold educators' efforts toward instructional innovation to realize goals beyond the immediate scope of the professional development.
4. Research	Lead research that informs the transformative use of technology in existing and emerging practices in schools, while contributing to evolving scholarship on innovations for teaching and learning

Figure. Situate. Design. Lead. © CTSC's Professional Development Model for Innovating Instruction, Detailed

laborates with building administrators to empower individual leadership and foster a culture of change and innovation. The guiding theoretical framework is the theory of change, which recognizes the complexity of the learning environment and the essential components required to facilitate transformative shifts in education. Strategic planning, the effective implementation of new teaching approaches, and the continuous development of teachers' beliefs about teaching and learning are emphasized as pivotal factors for driving meaningful change within the school system. The model also underscores the importance of creating "shared meaning" among key stakeholders, considering the institutional, historical, and cultural perspectives that shape relationships and language in the field of education [10,17].

Two recent National Science Foundation (NSF) grants—the Systemic Transformation of Inquiry Learning Environments (STILE 1.0) for STEM (Exploratory Award No. DRL-1238643) and STILE 2.0 (Early-Stage Design and Development Award No. DRL-1621387)—have established the model's positive impact on teachers' ability to design projects, to shift from disciplinary to transdisciplinary project design, and to shift instructional thinking to include inquiry-based approaches. The research findings from the STILE initiatives demonstrate the positive impacts of the model on teachers' pedagogical change as defined by shifts in STEM perspectives, STEM design practices, and STEM classroom practices [11]. The research in thirteen diverse school contexts, included 169 classroom visitations, 372 planning meetings, and over 51 hours of administrator interaction. The total average dosage was estimated at 61 hours per teacher, supporting 169 teachers in the New York City public school systems, cumulatively ultimately serving over 7536 students. The research identified positive changes that teachers have made under the STILE 2.0 program, specifically in teachers' ability to design projects, shift from a disciplinary/subject-orientation to a more sophisticated transdisciplinary focus, and broaden their instructional thinking to include more inquiry-based approaches [11].

Two recent grants from the National Science Foundation (NSF), namely STILE 1.0 (Systemic Transformation of Inquiry Learning Environments for STEM) and STILE 2.0, have demonstrated the positive impact of the model on teachers' capacity to design projects, transition from disciplinary to transdisciplinary project design, and adopt inquiry-based approaches in their instruction. The research findings from these initiatives highlight the constructive influence of the model on teachers' pedagogical change, as evidenced by shifts in STEM perspectives, STEM design practices, and STEM classroom practices. The research study took place in thirteen different school settings, comprising 169 classroom observations, 372 planning meetings, and over 51 hours of engagement with administrators. On average, each teacher received around 61 hours of support from the program, benefiting a total of 169 teachers in the New York City public school systems and ultimately influencing over 7536 students. The research findings revealed significant improvements in teachers' abilities to design projects, shift from a narrow disciplinary focus to a broader transdisciplinary perspective, and enhance their instructional thinking by integrating more inquiry-based methods. These positive changes were observed as a result of the STILE 2.0 program [11].

4. Visionary goals in professional development and CS education

4.1 A grand vision for professional development in CS education

To effectively introduce computer science curriculum in schools, professional development plays a crucial role for teachers to adopt a broader vision for the teaching profession that prioritizes learning and the needs of learners from diverse communities [18]. For example, professional development programs can provide teachers with training and resources to enhance their pedagogical skills in computer science education. This could include workshops on integrating technology into lessons, designing engaging coding activities, or implementing project-based learning in the computer science classroom. By equipping teachers with the necessary knowledge and tools, professional development empowers them to create meaningful learning experiences and cater to the diverse needs of their students in the realm of computer science [19,20].

With access to resources in computer science (CS) education, teachers receive valuable support in designing student-centered learning experiences that foster the development of students' identity and their willingness to actively engage in the broader computing community [21]. For instance, through professional development, teachers can learn about various tools, platforms, and instructional strategies that enable them to create interactive coding projects, collaborative programming exercises, or real-world CS applications. By incorporating these resources into their teaching, teachers can empower students to explore their interests, develop problem-solving skills, and cultivate a sense of belonging within the computing field [22]. This not only enhances students' learning experiences but also nurtures their enthusiasm and motivation to actively participate in the wider CS community beyond the classroom [23]. As a result, students will have more ownership and responsibility to explore concepts beyond essential programming ideas (e.g., loops, arrays, conditional statements). They will utilize their skills to build a deeper understanding of how these concepts apply to broader social and cultural contexts. This expanded perspective encourages students to consider the practical applications of computer science in various domains, such as healthcare, environmental sustainability, or social justice [24,25]. By connecting programming skills to real-world contexts, students develop a more comprehensive understanding of the societal impact and significance of computer science, empowering them to become critical thinkers and active contributors to their communities [26,27].

Finally, CS professional development should deepen teachers' content knowledge and provide curriculum resources, encouraging teachers to utilize different means, access, and representations to simulate abstract concepts in order to develop students' interests and curiosity in the field [28,29]. Specifically, ongoing research in professional development should prepare teachers to continuously innovate pedagogical practices to design, pilot and implement computer science curriculum in classroom settings [30]. It should invite educators to consider a broader, culturally relevant approach that designs curriculum situated for a range of learners, especially students who have been traditionally under-represented in the computing fields [31]. The CS literature has shown that women and students of color have been overlooked and excluded by the wider computing community [8], therefore, it is critical for teachers to identify effective approaches for engaging all students in the field of computer science education [32,33].

4.2 The complexity of the teacher changes in CS education

Research findings demonstrate that thoughtful professional development can significantly impact teachers' ability to incorporate technology into their classroom practices [34]. Studies indicate that teachers' beliefs and practices can evolve when provided with clear and specific instructions during professional development sessions in CS education [35]. In the context of computer science education, ongoing research should place a strong emphasis on exposing teachers to the design process and enabling them to explore the integration of technology in projects that promote a shift in instructional thinking [36]. For example, professional development programs can provide teachers with opportunities to engage in coding and computational thinking activities themselves, allowing them to experience firsthand the creative process involved in designing and developing computer programs. These programs can also offer teachers access to various technology tools and platforms specifically designed for computer science education. This includes platforms for creating interactive programming projects, virtual environments for exploring computer science concepts, and collaborative coding platforms. Through hands-on workshops and training sessions, teachers can gain practical knowledge of these tools and learn effective strategies for incorporating them into their teaching. To summarize, professional development initiatives offer teachers increased opportunities to gain familiarity with a variety of technology tools, demonstrate their use in classroom instruction, and provide checkpoints for reflection on the implementation process [37,38]. By successfully employing new practices and research-based methods, teachers receive further support in assimilating innovative approaches into their existing belief systems [39,40].

4.3 Reconceptualizing teaching practices in CS education

In the realm of computer science education, it is crucial to critically examine practices and emerging research in general education [41]. For example, teacher education programs can incorporate pedagogical strategies that promote hands-on learning experiences, such as coding workshops or robotics projects. By engaging students in these practical activities, they can develop a deeper understanding of programming concepts and gain valuable problem-solving skills. Additionally, project-based approaches can be implemented in computer science education [42]. For instance, students can work on real-world projects like designing a mobile application or creating a website for a local business. These projects not only reinforce technical skills but also encourage critical thinking and creativity as students navigate challenges and make design decisions [43].

Collaborative problem-solving is another important strategy to consider. An example of this could be organizing group activities where students collaborate to solve complex coding problems or develop a software solution together [44]. Through teamwork, students learn how to communicate effectively, share ideas, and leverage each other's strengths to achieve a common goal [45]. To stay up-to-date with the latest advancements, it is important for educators to explore research and developments in educational technology, computational thinking, and computer science instruction [46]. For instance, they can learn about new tools and platforms that facilitate interactive learning experiences or discover innovative teaching methods that enhance student engagement and understanding [47].

By reconceptualizing teacher education in computer science, educators can better prepare future teachers to design and deliver engaging and meaningful learning experiences [48]. For example, they can develop new curricular materials that incorporate coding exercises, multimedia resources, and interactive simulations to make learning more interactive and dynamic [49]. Furthermore, assessment approaches can be adapted to evaluate students' computational thinking skills, creativity, and problem-solving abilities [50]. This could involve designing coding challenges or projects that require students to apply their knowledge in practical contexts, as opposed to traditional exams or quizzes that solely focus on theoretical concepts [51]. By focusing on these aspects, students not only gain technical knowledge, but also develop the necessary skills to thrive in an ever-changing digital landscape. They become equipped with computational thinking skills, creativity, and problem-solving abilities, which are highly sought-after in the industry [52,53].

4.4 Embracing computer science education for all students with real-world connections

Specifically, effective professional development should provide teachers with more accessible resources that reduce barriers for teachers to learn, adopt, and integrate into the daily curriculum [54]. For instance, providing teachers sample curriculum that allows teachers to adapt and integrate into current lesson plans, could be effective for teachers to develop capacity in computer science disciplines, develop students' interests in exploring CS topics and encourage educators to understand the value of designing situated, culturally relevant computer science curriculum [55]. In addition, professional development should provide teachers with an easy-to-use platform that could encourage students to quickly build prototype, implementation solutions without creating complicated programming syntax [56]. For instance, the growing usage of block-based programming languages (e.g., PoseBlocks, App Inventor) has shown the value for students to build solutions, implement design, and create functional mobile applications without complex debugging and programming process [57,58]. The ongoing research study could further explore core processes and components that prepare teachers to use, adapt, and implement computer science curriculum with technology integration across diverse classroom settings domestically and internationally in computer science education [59,60].

5. Conclusions

To enhance teacher education in the field of computer science, it is crucial to equip teachers with the skills to strategically utilize technology in designing engaging curriculum that promotes deep learning in computer science education. Given the complexity of school systems, collaborative efforts involving researchers, scientists, and professionals are necessary to drive these transformative shifts. To prepare for the change, the *Innovating Instruction* model has shown an effective model to incorporate interactive and hands-on activities, project-based learning, and real-world applications of computer science concepts, tailor their instruction to meet the diverse needs and interests of their students, and constantly refine their instructional techniques and become more effective educators. By nurturing a community of practice and fostering collaboration among teachers, researchers, and professionals, the field can collectively drive the adoption of innovative technology and create a more engaging and impactful learning environment for students.

Author Contributions

Both authors made equal contributions to the manuscript.

Conflict of Interest

The authors have no conflicts of interest to declare.

Data Availability Statement

Situate. Design. Lead. © CTSC's Professional Development Model for Innovating Instruction is available at the website from Center for Technology and School Change, Teachers College, Columbia University. https://ctsc.tc.columbia.edu/

References

[1] Breazeal, C., 2022. AI Literacy for All with Prof. Cynthia Breazeal [Internet]. Available from: https://openlearning.mit.edu/news/ai-literacy-all-prof-cynthia-breazeal

[2] Darling-Hamond, L., Oakes, J., 2019. Preparing teachers for deeper learning. Harvard Education Press: Cambridge, MA.

[3] Podolsky, A., Kini, T., Darling-Hammond, L., 2019. Does teaching experience increase teacher effectiveness? A review of US research. Journal of Professional Capital and Community. 4(4), 286-308.

[4] Sutcher, L., Darling-Hammond, L., Carver-Thomas, D., 2019. Understanding teacher shortages: An analysis of teacher supply and demand in the United States. Education Policy Analysis Archives. 27(35).

[5] National Academies of Sciences, Engineering, and Medicine, 2018. How people learn II: Learners, contexts, and cultures. National Academies Press: Washington, D.C.

[6] Papadopoulos, I., Lazzarino, R., Miah, S., et al., 2020. A systematic review of the literature regarding socially assistive robots in pre-tertiary education. Computers & Education. 155, 103924.

[7] Rosenberg-Kima, R.B., Koren, Y., Gordon, G., 2020. Robot-supported collaborative learning (RSCL): Social robots as teaching assistants for higher education small group facilitation. Frontiers in Robotics and AI. 6, 148.

[8] 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.

[9] 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.

[10] Fullan, M., 2016. The new meaning of educational change. Teachers College Press: New York.

[11] Grover, S., Pea, R., 2013. Computational thinking in K-12: A review of the state of the field. Educational Researcher. 42(1), 38-43.

[12] Webb, M., Bell, T., Davis, N., et al. (editors), 2017. Computer science in the school curriculum: Issues and challenges. Tomorrow's Learning: Involving Everyone. Learning with and about Technologies and Computing: 11th IFIP TC 3 World Conference on Computers in Education, WCCE 2017; 2017 Jul 3-6; Dublin, Ireland. p. 421-431.

[13] Du, X., Parks, R., Tezel, S., et al. (editors), 2023. Designing a computational action program to tackle global challenges. SIGCSE 2023: Proceedings of the 54th ACM Technical Symposium on Education; 2023 Mar 15-18; Toronto ON, Canada. New York: Association for Computing Machinery. p. 1320-1320.

[14] Meier, E.B., Mineo, C., 2021. Pedagogical challenges during COVID: Opportunities for transformative shifts. Handbook of research on transforming teachers' online pedagogical reasoning for engaging K-12 students in virtual learning. IGI Global: Hershey. pp. 86-108.

[15] Meier, E.B., 2021. Designing and using digital platforms for 21st century learning. Educational Technology Research and Development. 69(1), 217-220.

[16] Darling-Hammond, L., Flook, L., Cook-Harvey, C., et al., 2020. Implications for educational

practice of the science of learning and development. Applied Developmental Science. 24(2), 97-140.

[17] Scardamalia, M.B., 2014. Knowledge building and knowledge creation. Cambridge handbook of the learning sciences. Cambridge University Press: Cambridge. pp. 297-417.

[18] Harju, V., Niemi, H., 202). Newly qualified teachers' support needs in developing professional competences: The principal's viewpoint. Teacher Development. 24(1), 52-70.

[19] Ng, D.T.K., Lee, M., Tan, R.J.Y., et al., 2022. A review of AI teaching and learning from 2000 to 2020. Education and Information Technologies. 1-57.

[20] Dash, B.B., 2022. Digital tools for teaching and learning English language in 21 st century. International Journal Of English and Studies. 4(2), 8-13.

[21] Biswas, S., Benabentos, R., Brewe, E., et al., 2022. Institutionalizing evidence-based STEM reform through faculty professional development and support structures. International Journal of STEM Education. 9(1), 1-23.

[22] McGill, M.M., Reinking, A., 2022. Early findings on the impacts of developing evidence-based practice briefs on middle school computer science teachers. ACM Transactions on Computing Education. 22(4), 1-29.

[23] Apiola, M., Sutinen, E., 2021. Design science research for learning software engineering and computational thinking: Four cases. Computer Applications in Engineering Education. 29(1), 83-101.

[24] Casey, E., Jocz, J., Peterson, K.A., et al., 2023. Motivating youth to learn STEM through a gender inclusive digital forensic science program. Smart Learning Environments. 10(1), 2.

[25] Tissenbaum, M., Weintrop, D., Holbert, N., et al., 2021. The case for alternative endpoints in computing education. British Journal of Educational Technology. 52(3), 1164-1177.

[26] Schaper, M.M., Smith, R.C., Tamashiro, M.A., et al., 2022. Computational empowerment in practice: Scaffolding teenagers' learning about emerging technologies and their ethical and societal impact. International Journal of Child-Computer Interaction. 34, 100537.

[27] Tsortanidou, X., Daradoumis, T., Barberá, E., 2019. Connecting moments of creativity, computational thinking, collaboration and new media literacy skills. Information and Learning Sciences. 120(11/12), 704-722.

[28] Alfaro-Ponce, B., Patiño, A., Sanabria-Z, J., 2023. Components of computational thinking in citizen science games and its contribution to reasoning for complexity through digital game-based learning: A framework proposal. Cogent Education. 10(1), 2191751.

[29] Ketelhut, D.J., Mills, K., Hestness, E., et al., 2020. Teacher change following a professional development experience in integrating computational thinking into elementary science. Journal of Science Education and Technology. 29, 174-188.

[30] Bragg, L.A., Walsh, C., Heyeres, M., 2021. Successful design and delivery of online professional development for teachers: A systematic review of the literature. Computers & Education. 166, 104158.

[31] Mystakidis, S., Fragkaki, M., Filippousis, G., 2021. Ready teacher one: Virtual and augmented reality online professional development for K-12 school teachers. Computers. 10(10), 134.

[32] Li, M., 2020. Multimodal pedagogy in TESOL teacher education: Students' perspectives. System. 94, 102337.

[33] Kafai, Y.B., Baskin, J., Fields, D., et al. (editors), 2020. Looking ahead: Professional development needs for experienced CS teachers. SIGCSE'20: Proceedings of the 51st ACM Technical Symposium on Computer Science Education; 2020 Mar 11-14; Portland OR, USA. New York: Association for Computing Machinery. p. 1118-1119.

[34] Tshukudu, E., Cutts, Q., Goletti, O., et al. (editors), 2021. Teachers' views and experiences on teaching second and subsequent programming languages. ICER 2021: Proceedings of the 17th

ACM Conference on International Computing Education Research; 2021 Aug 16-19; Virtual Event, USA. New York: Association for Computing Machinery. p. 294-305.

[35] Rich, P.J., Larsen, R.A., Mason, S.L., 2021. Measuring teacher beliefs about coding and computational thinking. Journal of Research on Technology in Education. 53(3), 296-316.

[36] Bereczki, E.O., Kárpáti, A., 2021. Technology-enhanced creativity: A multiple case study of digital technology-integration expert teachers' beliefs and practices. Thinking Skills and Creativity. 39, 100791.

[37] Griful-Freixenet, J., Struyven, K., Vantieghem, W., 2021. Exploring pre-service teachers' beliefs and practices about two inclusive frameworks: Universal design for learning and differentiated instruction. Teaching and Teacher Education. 107, 103503.

[38] Dignath, C., Rimm-Kaufman, S., van Ewijk, R., et al., 2022. Teachers' beliefs about inclusive education and insights on what contributes to those beliefs: a meta-analytical study. Educational Psychology Review. 34(4), 2609-2660.

[39] Almazroa, H., Alotaibi, W., 2023. Teaching 21st century skills: Understanding the depth and width of the challenges to shape proactive teacher education programmes. Sustainability. 15(9), 7365.

[40] Bhutoria, A., 2022. Personalized education and artificial intelligence in the United States, China, and India: A systematic review using a human-in-the-loop model. Computers and Education: Artificial Intelligence. 3, 100068.

[41] Bozkurt, A., 2020. Educational technology research patterns in the realm of the digital knowledge age. Journal of Interactive Media in Education. (1).

[42] Çiftci, S., Bildiren, A., 2020. The effect of coding courses on the cognitive abilities and problem-solving skills of preschool children. Computer Science Education. 30(1), 3-21.

[43] Saad, A., Zainudin, S., 2022. A review of Project-Based Learning (PBL) and Computational Thinking (CT) in teaching and learning. Learning and Motivation. 78, 101802.

[44] Wang, Y., 2023. The role of computer supported project-based learning in students' computational thinking and engagement in robotics courses. Thinking Skills and Creativity. 48, 101269.

[45] Bers, M.U., Blake-West, J., Kapoor, M.G., et al., 2023. Coding as another language: Research-based curriculum for early childhood computer science. Early Childhood Research Quarterly. 64, 394-404.

[46] Huang, W., Looi, C.K., 2021. A critical review of literature on "unplugged" pedagogies in K-12 computer science and computational thinking education. Computer Science Education. 31(1), 83-111.

[47] Yildiz Durak, H., Atman Uslu, N., Canbazoğlu Bilici, S., et al., 2022. Examining the predictors of TPACK for integrated STEM: Science teaching self-efficacy, computational thinking, and design thinking. Education and Information Technologies. 1-28.

[48] Lee, S.W.Y., Liang, J.C., Hsu, C.Y., et al., 2023. Students' beliefs about computer programming predict their computational thinking and computer programming self-efficacy. Interactive Learning Environments. 1-21.

[49] Lee, S.J., Francom, G.M., Nuatomue, J., 2022. Computer science education and K-12 students' computational thinking: A systematic review. International Journal of Educational Research. 114, 102008.

[50] Ung, L.L., Labadin, J., Mohamad, F.S., 2022. Computational thinking for teachers: Development of a localised E-learning system. Computers & Education. 177, 104379.

[51] Kallia, M., van Borkulo, S.P., Drijvers, P., et al., 2021. Characterising computational thinking in mathematics education: A literature-informed Delphi study. Research in Mathematics Education. 23(2), 159-187.

[52] Ogegbo, A.A., Ramnarain, U., 2022. A systematic review of computational thinking in science classrooms. Studies in Science Education. 58(2),

203-230.

[53] Chen, C.H., Liu, T.K., Huang, K., 2023. Scaffolding vocational high school students' computational thinking with cognitive and metacognitive prompts in learning about programmable logic controllers. Journal of Research on Technology in Education. 55(3), 527-544.

[54] Gao, X., Li, P., Shen, J., et al., 2020. Reviewing assessment of student learning in interdisciplinary STEM education. International Journal of STEM Education. 7(1), 1-14.

[55] Madkins, T.C., Martin, A., Ryoo, J., et al. (editors), 2019. Culturally relevant computer science pedagogy: From theory to practice. 2019 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT); 2019 Feb 27; Minneapolis, MN, USA. New York: IEEE. p. 1-4.

[56] Guskey, T.R., 2002. Does it make a difference? Evaluating professional development. Educational Leadership. 59(6), 45-51.

[57] Li, L., Ruppar, A., 2021. Conceptualizing teacher agency for inclusive education: A systematic and international review. Teacher Education and Special Education. 44(1), 42-59.

[58] Tissenbaum, M., Sheldon, J., Abelson, H., 2019. From computational thinking to computational action. Communications of the ACM. 62(3), 34-36.

[59] Madkins, T.C., Howard, N.R., Freed, N., 2020. Engaging equity pedagogies in computer science learning environments. Journal of Computer Science Integration. 3(2).

[60] Morales-Chicas, J., Castillo, M., Bernal, I., et al., 2019. Computing with relevance and purpose: A review of culturally relevant education in computing. International Journal of Multicultural Education. 21(1), 125-155.