Asia Research Network Journal of Education
Vol. 3, No. 2, pp. 41-47, May. – Jun. 2023

A Pedagogical Framework of Computational Thinking Components for Enhancing Pre-service Teachers' Problem- Solving Capability

Hi-Lian Jeng*, Zu Heng Khoo, and Li-Wen Liu

Graduate Institute of Digital Learning and Education, National Taiwan University of Science and Technology 43, Section 4, Keelung Road, Taipei, Taiwan, ROC
*Corresponding author email: jenghl@mail.ntust.edu.tw, M11011023@mail.ntust.edu.tw, M10611003@mail.ntust.edu.tw

Received: 14 May 2023 Revised: 27 Aug 2023 Accepted: 29 Aug 2023

Abstract: It has been a global trend to incorporate computational thinking (CT) into K-12 education and curriculum, magnifying the importance of CT in the technological era. Teachers are the key drivers in the process and success of integrating CT into classroom instruction. It is even imperative to enhance pre-service teachers' CT literacy before they enter the education ecosystem. A pedagogical framework of teaching CT in solving problems is derived for pre-service teachers in the study, which is purposefully simplified from a previous study for the sake of the pre-service teachers' training and development. With frequent practice, it is expected to increase pre-service teachers' CT literacy so that they can apply in daily problem solving and future course design, in all subject and cross domains.

Keywords: Computational Thinking, Pre-service Teacher, Problem-solving Capability

Introduction

Technology and tools are now essential parts of daily lives, with software and hardware frequently updated to meet evolving demands. As society changes rapidly, individuals must not only acquire knowledge but also solve various problems. Although problem-solving has been practiced since the 19th century, it was viewed as a systematic cognitive skill used to address clear-cut problems. Today's problems are more complex, and problem-solving is now considered a critical ability to adapt to the 21st century's challenges (Chalkiadaki, 2018; Dede, 2010; Van Laar et al., 2017; Voogt & Roblin, 2010). Computational Thinking (CT) has become a valued literacy in education and walks of life as a thinking routine that can help solve complex problems.

Studies have focused on cultivating CT to enhance problem-solving abilities (Gökçe & Yenmez, 2022; Majeed et al., 2022; Park & Kwon, 2022). The worldwide K-12 education has integrated CT, emphasizing its importance in curriculum standards (Grover & Pea, 2013; Sanusi et al., 2022; Wang et al., 2019; Xing & Zhang, 2020). Scholars debate whether K-12 students can receive effective high-order thinking cultivation in CT, which is contingent on the role of teachers (Fessakis et al., 2018; Israel et al., 2015; Kim & Kim 2018; Yang et al., 2018). Teachers are the key drives in the process and success of integrating CT into classroom instruction.

Literature has indicated that introducing CT into classroom instruction is challenging for teachers, given challenges such as shortages of specialized teachers, teaching strategies and instructional materials, inadequate understanding of the importance of CT among teachers in fields other than computer and information sciences, and insufficient professional competence among teachers (Kong et al., 2020; Lee et al., 2023; Mouza et al., 2017; Weintrop et al., 2016; Yadav et al., 2017). It is more than important in pre-service teachers. While still in pre-service teacher training, there is time to acquaint them with CT to become habitual in practicing CT in daily living, and course designs as well. Therefore, this paper proposes a pedagogical framework of CT components for enhancing their problem-solving capability. The framework aims to provide a useful reference for various professional domains and cross disciplinaries such as STEAM (Science, Technology, Engineering, Arts, and Mathematics).

CT Education for Pre-Service Teachers

CT education for teachers and pre-service teachers is a growing area of interest in education research. As technology use continues to become more and more prevalent in classrooms, it is important for teachers to have a strong foundation in CT skills and pedagogy (Bower at al., 2017; Mouza et al., 2017). Studies have shown that pre- service teachers who receive instruction in CT are more likely to incorporate CT into their teaching practices (Chang & Peterson, 2018). Mouza et al. (2018) argue that teaching CT to pre-service teachers is essential in order to prepare future generations of students in a technology-driven society. Cutumisu and Guo (2019) suggest that incorporating CT into teacher education programs can enhance pre-service teachers' problem-solving abilities and promote interdisciplinary thinking. Furthermore, Dong et al. (2023) found that pre-service teachers who received CT training showed significant improvement in their CT skills and coding abilities. Therefore, it is important for teacher education programs to provide pre-service teachers opportunities to learn and practice CT skills, as this can improve their teaching practices and further, better prepare their students for the skills needed in the 21st century.

Several studies have explored the impact of programming activities on pre-service teachers' CT abilities, with encouraging results. Jaipal-Jamani and Angeli (2017) implemented a robotics course that required pre-service teachers to build and program three robotic models. The researchers found that after the robotics intervention, the pre-service teachers' CT skills improved significantly, as evidenced by their successful completion of programming tasks. Mouza et al. (2017) investigated how pre-service teachers implement CT concepts in their courses using Scratch and other educational technologies. The findings of the study revealed that the pre-service teachers' CT skills, utilization of technological tools, and classroom activities witnessed significant enhancements post the completion of the training course. Butler and Leahy (2021) reported success in deepening pre-service teachers' understanding of CT and embedding it into practice with tools such as Scratch. Overall, these studies suggest that incorporating programming activities and CT concepts into pre-service teacher education programs can effectively enhance their CT skills and pedagogical practices. However, these studies do not explicitly provide a teaching framework suitable for application in the classroom.

A Problem-Solving-Based Framework for Teaching CT

Jeng et al. (2019) synthesized literature related to problem-solving and CT and proposed a procedural framework that integrates seven problem-solving stages and ten CT components for designing and evaluating CT instructional courses. This framework acknowledges the value of integrating problem-solving with CT as it provides a clear and structured procedure to guide instructional design and assessment. It can be customized to meet specific requirements such as various nature of tasks, target audiences, lengths of study time, and the needs for plugged or unplugged activities, by adjusting the procedural stages in the framework. This framework offers a practical tool for educators and researchers seeking to develop CT literacy in students. It facilitates the design of CT teaching and learning assessments, ultimately leading to developing more effective CT education programs.

For the purpose of an intensive workshop for pre-service teacher training, a concise framework (FIGURE 1. and TABLE 1.) is developed based on Jeng et al.'s, in which major CT components are reserved and merged, with the addition of a very first participation stage, and still the corresponding problem-solving procedural flow is followed. The framework includes evaluation and creation which are often difficult to achieve and are overlooked in traditional learning. They are sustained in the framework to ensure a complete and ideal learning, even in the context of an intensive workshop as in the present study. An elucidation of the framework is followed.

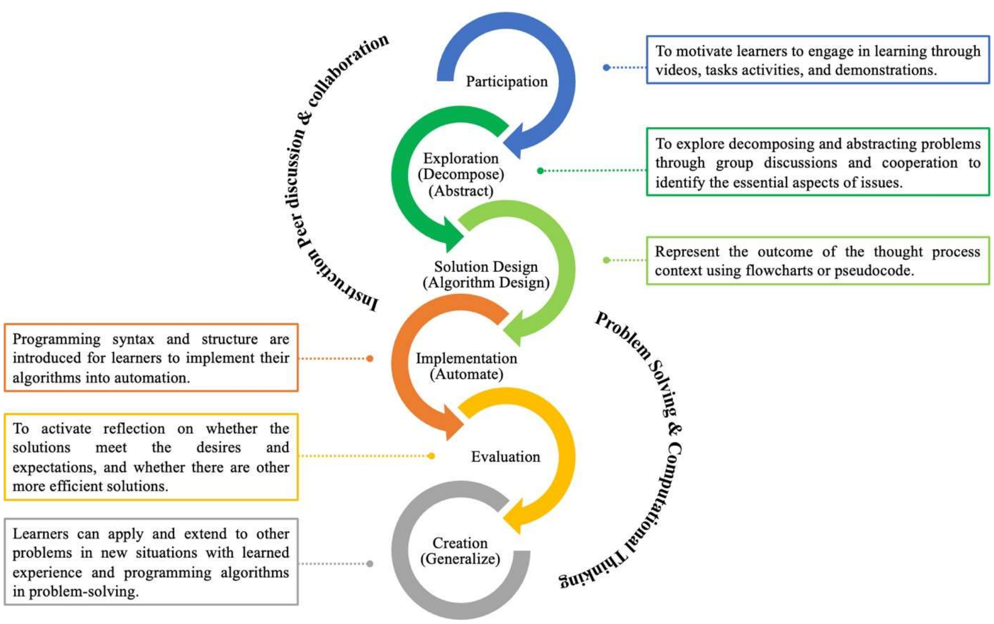


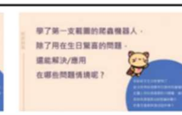
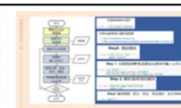
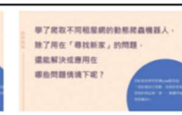
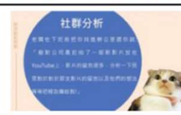
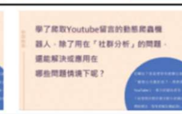
Figure 1: A six-stage CT Teaching Framework

There are six procedural stages in the framework. Firstly, the initial phase involves motivating learners through videos, task activities, and demonstrations for them to recognize the everyday CT application in real life, thereby encouraging their **participation** and involvement. Secondly, problem **exploration** is practiced by using real-life problems for learners to think about the essence of the problems through instructional guide, and peer collaboration and discussion, to identify the cores of the problem and break down complex problems into smaller, more manageable parts. Thirdly, using specific representations as in computer science, such as flowcharts or pseudocodes, learners illustrate their algorithms into **solution design**. Fourthly, a specific programming syntax and structure (such as Python) is introduced to learners and help them **implement** the previously established representations into computer program instructions, allowing them to verify the benefits of automation and the effectiveness of computing tools in solving complex problems. Next, after the implementation of the program, learners **evaluate** the solution, reflecting on whether the solution meets as expected and whether there are other more efficient solutions. Lastly, learners extend and generalize the learning experiences beyond the specific contexts provided by the instructor into new contexts, solving new problems and **creating** for new needs.

Conclusion

CT has become an essential skill set for individuals to thrive in the 21st century. As such, it is crucial to incorporate CT into teacher education programs to prepare future teachers and students adequately. Studies have shown that providing pre-service teachers with CT instruction improves their CT skills, in which programming activities have proven to be an effective way. However, a pedagogical framework is needed to guide instructional design and assessment in CT education. Jeng et al.'s (2019) procedural framework integrates problem solving with CT, providing a clear and structured procedure for CT instructional courses. In this study, a concise CT framework based on Jeng et al.'s work is proposed, suitable for an intensive workshop for pre-service teacher training. This framework consists of six procedural stages, including motivating learners to participate, problem exploration, solution design, implementation, evaluation, and creation. Through these stages, pre-service teachers can gain a deep understanding of CT and problem solving, enabling them to apply CT in daily lives and future teaching practices as well. Ultimately, the proposed framework offers a practical tool for educators and researchers seeking to develop CT literacy in students and provides a means of assessing the effectiveness of CT education. Incorporating CT into teacher education and training is essential to prepare future generations of students for a technology-driven society.

Table 1: The Correspondence between Teaching Activities and Components of CT

Topic	Participation	Exploration (Abstract)	Exploration (Decompose)	Solution Design (Algorithm Design)	Implementation (Automate)	Evaluate	Creation (Generalize)
	To motivate learners to engage in learning through videos, tasks activities, and demonstrations.	To explore decomposing and abstracting problems through group discussions and cooperation to identify the essential aspects of issues.		Represent the outcome of the thought process context using flowcharts or pseudocode.	Programming syntax and structure are introduced for learners to implement their algorithms into automation.	To active reflection on whether the solutions meet the desires and expectations, and whether there are other more efficient solutions.	Learners can apply and extend to other problems in new situations with learned experience and programming algorithms in problem-solving.
Automated Screenshot Bot							
Tailored Price Comparison Exploration							
In Relation to Sentiment Word Cloud							

References

Bower, M., Wood, L. N., Lai, J. W., Howe, C., Lister, R., Mason, R., Highfield, K., & Veal, J. (2017). Improving the computational thinking pedagogical capabilities of school teachers. *Australian Journal of Teacher Education,42*(3), 53-72.

Butler, D., & Leahy, M. (2021). Developing preservice teachers' understanding of computational thinking: A constructionist approach. *British Journal of Educational Technology, 52*(3), 1060-1077.

Chalkiadaki, A. (2018). A systematic literature review of 21st century skills and competencies in primary education. *International Journal of Instruction, 11*(3), 1-16

Chang, Y. H., & Peterson, L. (2018). Pre-service teachers' perceptions of computational thinking. *Journal of Technology and Teacher Education, 26*(3), 353-374.

Cutumisu, M., & Guo, Q. (2019). Using topic modeling to extract pre-service teachers' understandings of computational thinking from their coding reflections. *IEEE Transactions on Education, 62*(4), 325-332.

Dede, C. (2010). Comparing frameworks for 21st century skills. *21st century skills: Rethinking how students learn, 20*(2010), 51-76.

Dong, W., Li, Y., Sun, L., & Liu, Y. (2023). Developing pre-service teachers' computational thinking: a systematic literature review. *International Journal of Technology and Design Education*, 1-37.

Fessakis, G., Komis, V., Mavroudi, E., & Prantsoudi, S. (2018). Exploring the scope and the conceptualization of computational thinking at the K-12 classroom level curriculum. *Computational thinking in the STEM disciplines: Foundations and research highlights*, 181-212.

Gökçe, S., & Yenmez, A. A. (2022). Ingenuity of scratch programming on reflective thinking towards problem solving and computational thinking. *Education and Information Technologies*, 1-25.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational researcher, 42*(1), 38-43.

Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education, 82*, 263-279.

Jaipal-Jamani, K., & Angeli, C. (2017). Effect of robotics on elementary preservice teachers' self-efficacy, science learning, and computational thinking. *Journal of Science Education and Technology, 26*, 175-192.

Jeng, H.-L., Liu, L.-W., & Chen, C.-N. (2019). Developing a Procedural Problem-based Framework of Computational Thinking Components. *In the Conference Proceedings of 2019 8th International Congress on Advanced Applied Informatics (IIAI-AAI)*, 272-277. Toyama, Japan.

Kim, S., & Kim, H. Y. (2018). A computational thinking curriculum and teacher professional development in South Korea. *In the Proceedings of Computational Thinking in the STEM Disciplines*, 165-178.

Kong, S. C., Lai, M., & Sun, D. (2020). Teacher development in computational thinking: Design and learning outcomes of programming concepts, practices and pedagogy. *Computers & Education, 151*, 103872.

Lee, J., Joswick, C., & Pole, K. (2023). Classroom play and activities to support computational thinking development in early childhood. *Early Childhood Education Journal, 51*(3), 457-468.

Majeed, B. H., Jawad, L. F., & ALRikabi, H. T. (2022). Computational Thinking (CT) Among University Students. *International Journal of Interactive Mobile Technologies, 16*(10).

Mouza, C., Yadav, A., & Ottenbreit-Leftwich, A. (2018). Developing computationally literate teachers: Currentperspectives and future directions for teacher preparation in computing education. *Journal of Technology and Teacher Education, 26*(3), 333-352.

Mouza, C., Yang, H., Pan, Y. C., Ozden, S. Y., & Pollock, L. (2017). Resetting educational technology coursework for pre-service teachers: A computational thinking approach to the development of technological pedagogical content knowledge (TPACK). *Australasian Journal of Educational Technology, 33*(3).

Park, W., & Kwon, H. (2022). Bringing computational thinking to technology education classrooms: Hacking car activity for middle schools in the republic of korea. *International*

Journal of Technology and Design Education, 1-19.

Sanusi, I. T., Olaleye, S. A., Oyelere, S. S., & Dixon, R. A. (2022). Investigating learners' competencies for artificial intelligence education in an African K-12 setting. Computers and Education Open, 3, 100083.

Van Laar, E., Van Deursen, A. J., Van Dijk, J. A., & De Haan, J. (2017). The relation between 21st-century skills and digital skills: A systematic literature review. Computers in human behavior, 72, 577-588.

Voogt, J., & Roblin, N. P. (2010). 21st century skills. Discussienota. Zoetermeer: The Netherlands: Kennisnet, 23(3), 2000.

Wang, J., Zhang, Y., Li, X., Ren, Q., & Mei, L. (2019). Research on the Current Situation and Development Trend of Computational Thinking in K-12 Education in China. CoolThink@JC, 233.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. Journal of science education and technology, 25, 127-147.

Xing, Y., & Zhang, Y. (2020). International Comparative Study on Computational Thinking Education in K-12. The Educational Review, USA, 4(8), 166-175.

Yadav, A., Good, J., Voogt, J., & Fisser, P. (2017). Computational thinking as an emerging competence domain. Proceedings of Competence-based vocational and professional education, 1051-1067.

Yang, H., Mouza, C., & Pan, Y.-C. (2018). Examining pre-service teacher knowledge trajectories of computational thinking through a redesigned educational technology course. In the Conference Proceedings of the International Society of the Learning Sciences, 368-375.