JOURNAL OF LEARNING ANALYTICS

Volume 13(1), 31–41. https://doi.org/10.18608/jla.2026.9077

Profiling Pre-service Teachers' Computational Thinking: The Role of Metacognition and Coding Experience

Tanya Chichekian[1], Maria Cutumisu[2], Annie Savard[3] and Yi-Mei Zhang[4]

Abstract

Computational thinking (CT) is a vital skill set for pre-service teachers who will need to foster computational literacy in K–12 classrooms, yet the factors influencing their CT skills remain less understood than those for K–12 students or in-service teachers. This study leverages multimodal data to investigate how pre-service teachers (n=128) differ in CT skills, the predictive role of metacognitive strategies and prior coding experience, and variations in online behaviours. Using latent profile analysis, we identified three profiles based on digital literacy, problem-solving, and coding comfort (Novice, Developing, and Proficient), revealing heterogeneity in CT, and supporting non-linear skill acquisition. Linear discriminant analysis revealed that metacognitive strategies and prior coding experience significantly predict profile membership, validating the interplay of technical and cognitive factors in the development of CT skills. Behavioural data from an interactive problem-solving task showed that, compared to Novices and Developing learners, Proficient learners were more task efficient and perceived fewer challenges during task completion. Implications for designing a learning analytics dashboard to visualize profiles and behavioural metrics to support adaptive, equitable, and personalized teacher training are discussed, thereby enhancing pre-service teachers' readiness to integrate CT into K–12 education.

Notes for Practice

- This study identifies three distinct CT learner profiles (Novice, Developing, Proficient) among pre-service teachers.
- Metacognitive strategies and prior coding experience are key factors that differentiate these profiles and influence CT task performance.
- Proficient learners demonstrate greater task efficiency and lower perceived difficulty, while Novice learners benefit from targeted scaffolding.
- CT profiles can guide teacher educators in tailoring instruction, offering structured support for Novice learners and more complex CT activities for Proficient learners.
- Monitoring behavioural indicators such as time-on-task and perceived difficulty can inform adaptive feedback and future dashboard design to support personalized CT development.

Keywords: Computational thinking, pre-service teachers, prior coding experience, metacognitive strategies, latent analysis

Submitted: 30/05/2025 — **Accepted:** 24/11/2025 — **Published:** 14/03/2026

Corresponding author [1]Email: tanya.chichekian@usherbrooke.ca Address: Department of Pedagogy, Faculty of Education, Université de Sherbrooke, 150, place Charles-Le Moyne, C. P. 200, Longueuil (Québec) J4K 0A8; 2500 boulevard de l'Université, Sherbrooke, Québec, J1K 2R1, Canada. ORCID iD: https://orcid.org/0000-0002-6156-0779

[2]Email: maria.cutumisu@mcgill.ca Address: Department of Educational & Counselling Psychology, Faculty of Education, McGill University, Education Building, 3700 rue McTavish, Montréal, Québec, H3A 1Y2, Canada. ORCID iD: https://orcid.org/0000-0003-2475-9647

[3]Email: annie.savard@mcgill.ca Address: Department of Integrated Studies in Education, Faculty of Education, McGill University, Education Building, 3700 rue McTavish, Montréal, Québec, H3A 1Y2, Canada. ORCID iD: https://orcid.org/0000-0002-0285-0952

[4]Email: yi-mei.zhang@mail.mcgill.ca Address: Department of Integrated Studies in Education, Faculty of Education, McGill University, Education Building, 3700 rue McTavish, Montréal, Québec, H3A 1Y2, Canada. ORCID iD: https://orcid.org/0000-0002-4955-6726

1. Introduction

In our increasingly technology-oriented society, computational thinking (CT) has become a critical skill set for engaging with digital innovation (Kotsopoulos et al., 2017). It is vital for addressing complex problems in a technology-driven world. It includes skills such as abstraction (viewing problems at varying levels of detail), algorithmic thinking (expressing problems as a series of step-by-step instructions), decomposition (tackling a problem by solving smaller constituent issues), and pattern recognition (identifying connections between a new problem and previous ones) (Wing, 2006; Hoyles & Noss, 2015). These skills are essential for pre-service teachers, who must master CT to design and deliver K–12 lessons that foster computational literacy across disciplines, such as using decomposition to create scaffolded coding activities or abstraction to simplify data analysis for students (Yadav et al., 2016). This interdisciplinary skill set enhances problem-solving across educational contexts, making its integration into teacher preparation essential (Barr & Stephenson, 2011). Despite CT's growing importance, research on factors influencing pre-service teachers' CT skills remains limited, particularly compared to studies on K–12 students or in-service teachers. This multimodal study leverages latent profile analysis (LPA) and behavioural data from an interactive CT task to examine metacognitive strategies and prior coding experience as predictors of CT proficiency among pre-service teachers, responding to calls for data-driven teacher preparation (Liu et al., 2024; Knight et al., 2017). Specifically, we address the following research questions: (1) What distinct profiles of pre-service teachers emerge based on their CT skills? (2) To what extent do metacognitive strategies and prior coding experience explain differences between CT profiles? and (3) How do pre-service teachers' online problem-solving behaviours differ across CT skill profiles?

2. Literature Review

Over the past decade, various researchers have explored models of CT, emphasizing concepts, practices, and perspectives (Barr & Stephenson, 2011; Weintrop et al., 2016). From an educational standpoint, an effective CT model should transcend mere coding and programming language knowledge, prioritizing pedagogical approaches to teaching CT through specific classroom activities. For instance, Shute et al. (2017) propose a CT model emphasizing a mindset for creating reusable solutions, applicable with or without technology, which can be evaluated through performance-based pedagogical approaches. According to Shute et al. (2017), CT involves the ability to solve problems effectively and efficiently, whether algorithmically with or without computer assistance, and to create solutions that can be reused in various situations. This definition underscores that CT is fundamentally a mindset and a way of acting, demonstrated through specific skills that can serve as the basis for evaluating CT abilities through performance-based assessments.

In educational settings, CT development has been approached through unplugged methods (puzzles, manipulatives) to foster foundational skills and plugged methods (e.g., Scratch, robotics kits) for applications (Jaipal-Jamani & Angeli, 2017; Kotsopoulos et al., 2017; Mouza et al., 2017). Research comparing these approaches suggests that a blend of both may be most effective, with unplugged activities building foundational understanding and progressing to plugged activities that solidify practical application and optimize CT proficiency, particularly for pre-service teachers who must master and teach these skills (Román-González et al., 2019).

2.1. Pre-service Teachers' CT Development

Despite growing recognition of CT's importance, there remains a significant disparity in pre-service teachers' preparedness to teach CT skills. CT, as defined by Shute et al. (2017), includes decomposition (breaking problems into manageable parts), abstraction (focusing on essential details), algorithmic thinking (designing step-by-step solutions), and pattern recognition (identifying similarities across problems). These skills are critical for pre-service teachers, who must master CT to design effective K–12 lessons and teach students to apply CT across disciplines. For example, decomposition enables pre-service teachers to create scaffolded lesson plans by breaking complex CT concepts into manageable activities, such as sequencing tasks in Scratch. Abstraction helps simplify coding concepts for diverse learners, such as focusing on key patterns in data analysis. Algorithmic thinking equips pre-service teachers to develop structured instructional strategies, such as step-by-step coding exercises, while pattern recognition helps identify student learning trends to tailor instruction (Yadav et al., 2016). Many pre-service teachers enter programs with limited CT exposure, facing challenges such as insufficient CT-focused coursework and misconceptions that equate CT with programming or STEM exclusivity (Bower & Falkner, 2015; Sadik et al., 2017). These barriers contribute to a steep learning curve, uneven skill development, and low confidence among pre-service teachers to integrate CT into their future classrooms (Bers, 2017; Grover & Pea, 2013; Tang et al., 2020). Recent research highlights the need for targeted professional development (PD) programs to address these gaps, emphasizing structured CT interventions that leverage prior coding experience to enhance task efficiency and pedagogical readiness (Chaabi et al., 2025; Ramirez-Salgado et al., 2025). Targeted interventions, such as robotics courses or Scratch-based curricula, can address these challenges by building CT skills, such as decomposition and abstraction, through hands-on, constructionist learning environments, thereby improving self-efficacy and pedagogical readiness (Butler & Leahy, 2021; Jaipal-Jamani & Angeli, 2017; Mouza et al., 2017). These interventions, however, must be tailored to novices' needs, progressively building CT skills to address coding disparities and misconceptions.

2.2. Metacognition and Prior Coding Experience

According to Pintrich et al. (1993), metacognitive strategies, involving self-regulation and reflection, enhance CT by enabling pre-service teachers to plan and evaluate problem-solving processes. In the context of CT, metacognitive strategies can enhance pre-service teachers' ability to plan, monitor, and evaluate their problem-solving processes, thereby improving their overall computational proficiency (Cheng et al., 2023). Recent studies highlight that prior coding experience also predicts CT proficiency, increasing digital literacy and coding comfort, though it may not ensure higher-order skills like abstraction (Chan et al., 2021; Bower & Falkner, 2015). Emerging reviews further emphasize the synergy between metacognitive strategies and coding experience, advocating for multimodal assessments that combine self-reports and behavioural data to comprehensively capture CT development (Aldemir et al., 2025; Chaabi et al., 2025). This aligns with the broader literature on CT, which emphasizes the importance of practical coding knowledge in enhancing computational proficiency (Grover & Pea, 2013; Shute et al., 2017). However, the relationship between prior coding experience and CT skills is not always straightforward. Although coding experience can provide a foundation for understanding computational concepts, it does not necessarily lead to the development of higher-order CT skills, such as abstraction and algorithmic thinking (Bower & Falkner, 2015). While prior research has examined CT components in isolation (e.g., coding drills or problem-solving frameworks), few studies have investigated how metacognition and coding experience interact, a critical gap given pre-service teachers' dual role in mastering and teaching CT (Dong et al., 2024; Liu et al., 2024). Pre-service teachers with prior coding experience may, for instance, rely more heavily on metacognitive strategies to navigate complex problems; however, this hypothesis has yet to be tested empirically.

3. The Present Research

This study aims to profile pre-service teachers' CT skills, examine how metacognitive strategies and prior coding experience shape CT proficiency, and inform personalized teacher education. The study extends Shute et al.'s (2017) CT model, which defines CT as cognitive skills including decomposition, abstraction, algorithmic thinking, and pattern recognition by incorporating metacognitive strategies (Pintrich et al., 1993) and prior coding experience (Bower & Falkner, 2015) as key predictors. These CT skills are essential for pre-service teachers to master and teach CT in K–12 classrooms. For instance, decomposition aids in designing scaffolded lesson plans, abstraction simplifies CT concepts for diverse learners, algorithmic thinking supports structured teaching strategies, and pattern recognition helps tailor instruction to students' needs, addressing gaps in CT readiness (Chichekian et al., 2025). These predictors, absent from Shute et al.'s framework, are hypothesized to enhance the development of CT skills in pre-service teachers, thus addressing their readiness to integrate CT into K–12 curricula.

To achieve these goals, the study employs a learning analytics approach, integrating survey measures of digital literacy (supporting abstraction), problem-solving (reflecting decomposition and algorithmic thinking), and coding comfort (linked to pattern recognition confidence), aligned with Shute et al.'s CT components. Using LPA, the study identifies distinct CT skill profiles that contribute to Shute et al.'s framework, thereby providing a more comprehensive model for understanding CT development in teacher education. Survey data is complemented with trace data from a Qualtrics-hosted CT task. These behavioural traces automatically capture participants' interactions during the CT task, providing process-oriented insights into problem-solving efficiency and engagement. This multimodal approach responds to calls for data-driven, analytics-informed teacher preparation and offers a scalable framework for personalized interventions (Knight et al., 2017).

4. Methods

4.1. Participants

This quantitative study was conducted with 128 pre-service teachers (17 males, 101 females, 8 identified as other, and 2 missing) from a Faculty of Education. There were $n = 19$ participants in Year 1, $n = 9$ in Year 2, $n = 8$ in Year 3, $n = 73$ in Year 4, and $n = 19$ in Year 5+. Participants were recruited from various programs: $n = 73$ from Elementary Education, $n = 37$ from Secondary Education, and $n = 18$ from Physical Education, TESL, or Music. Finally, 62 participants had some familiarity with coding languages, such as Scratch, Python, C++, JavaScript, and HTML. The uneven year distribution, with Year 4 students predominant, reflects recruitment from advanced courses.

4.2. Data Collection

A research assistant visited classes with the instructor's consent to explain the study and obtain participant consent. Participants completed an online Qualtrics survey, including Likert-type scales for demographics, CT skills, and metacognitive strategies, followed by the Programming Lamps CT task. The task's difficulty level (easy, medium, hard) was assigned randomly using the Qualtrics randomization feature, ensuring balanced distribution (24–30% per level).

4.2.1. CT Interactive Task Design

The Programming Lamps task, adapted from the Bebras contest (Dagienė & Futschek, 2008), assessed CT by requiring participants to sequence button presses to light specific lamp patterns, targeting algorithmic thinking and pattern recognition. Hosted on Qualtrics, the task was accessible to those with no programming knowledge and featured three difficulty levels: easy, medium, and hard. A practice playground, a preliminary Qualtrics block for exploring the interface, preceded the task. Qualtrics automatically captured behavioural data: playground time (PG_Time), task completion time (L_Time), and number of clicks (L_Clicks). Participants' interactions, such as button presses and replays of animated feedback, generated traceable events (e.g., timestamps, click counts), enabling learning analytics to uncover patterns in problem-solving efficiency and engagement (Wise & Schwarz, 2017). Three computer science interns optimized the task by adding a playground, incremental difficulty, and animated feedback (response replays) via embedded JavaScript. After animated feedback for an incorrect attempt, participants were allowed a second try, thereby enhancing behavioural trace data collection (Wise & Schwarz, 2017).

4.2.2. Measures

Participants reported gender, program, year of study, and prior coding experience (yes/no, specific languages). Coding experience was assessed with a yes/no question ("Are you familiar with one or more of these coding languages?") and a checklist follow-up ("If yes, select the ones you are familiar with, e.g., Scratch, Python, C++").

Computational Thinking (CT): CT skills were assessed using an adapted computational thinking instrument (Cutumisu et al., 2019) with pre-service teachers rating agreement (1 = strongly disagree to 7 = strongly agree) on digital technology (CT_DL, 3 items, $\alpha = 0.77$), problem-solving (CT_PS, 3 items, $\alpha = 0.77$), and coding comfort (CT_CD, 3 items, $\alpha = 0.73$). Example items included: CT_DL ("I find it easy to use new technology"), CT_PS ("I can break down complex problems into smaller steps"), and CT_CD ("I feel confident writing simple code").

L_Score: The *Programming Lamps* task is a CT assessment adapted from the Bebras contest (Dagienė & Futschek, 2008). The task is implemented as a Qualtrics survey block with interactive elements (e.g., buttons labelled A, B, C) and embedded JavaScript for logic and feedback. The task requires participants to sequence button presses to light specific lamp patterns. Each participant is randomly assigned a task item at one of three difficulty levels (easy, medium, hard) using the Qualtrics randomization feature.

For example, consider a medium-difficulty task item: "Press buttons A, B, C in a sequence to light lamps 1 and 3 but not 2." In the first attempt, the participant submits "A, C, B," which lights lamps 1 and 2 (incorrect). Qualtrics records this, displays an animated replay showing the error (e.g., lamp 2 should not light), and prompts a second attempt. The participant then adjusts to "B, A, C," which correctly lights lamps 1 and 3. Qualtrics assigns L_Score = 1. If the second attempt was also incorrect (e.g., "C, A, B" lights all lamps), L_Score = 0. If the first attempt had been correct, L_Score = 2. L_Score is stored as an embedded data field in Qualtrics (e.g., L_Score = 0, 1, or 2) and exported alongside other behavioural metrics. The scoring logic is embedded in the survey's JavaScript, ensuring consistency across participants. An example of the easy, medium, and difficult levels of the task, showing the Qualtrics interface with buttons and a target lamp pattern, appears in Appendix 1.

PG_Time (Seconds in Playground): PG_Time measured time spent in the preliminary "practice playground" block, a dedicated Qualtrics page for interface exploration before the main task. Qualtrics recorded the duration from page load (when the playground appears) to page submit (when the user advances). Using a timing question on this isolated block, it calculated Page Submit - Page Load in seconds. Upon loading the page, Qualtrics sets a start timestamp. As the user interacts (e.g., testing buttons without consequences), the timer runs. Page submit triggers the end timestamp. For example, if a user loaded the playground at $t = 0$ and advances after 64 seconds, PG_Time = 64.

L_Time (Seconds on Task): Similar to PG_Time, L_Time tracks the main task block using a timing question. It measures from page load to page submit, including time for attempts, feedback reviews, and second tries after incorrect responses. JavaScript handles incremental difficulty and animated replays, ensuring timestamps align with interactive phases. The timestamp begins on task page load and ends on submit. For multi-attempt tasks, time is cumulative (e.g., first attempt + feedback review + second attempt). If users go back (via browser), Qualtrics adds the revisited time. For example, a sample mean of 124.33 seconds would include sequencing button presses (e.g., for a medium-difficulty pattern) and feedback loops.

L_Clicks (Click Count): Qualtrics logs clicks via JavaScript on interactive elements (e.g., buttons A, B, C in the task). Each click (button press, replay request) increments a counter, stored as embedded data. The timing question's "click count" metric aggregates total interactions on the task page. Clicks are tallied until submission, including errors or replays. For example, a sample mean of 50.07 clicks might include 20–30 button presses per attempt plus replay clicks.

L_Diff (Perceived Difficulty): Post-task, participants rated the difficulty of the task using an original single-item scale developed for this study. The item asked, "How difficult did you find the Programming Lamps task?" with responses on a 10-point Likert-type scale (1 = very easy to 10 = very hard). This measure was designed to capture participants' immediate, subjective perceptions of task difficulty in the digital learning environment, aligning with learning analytics approaches that prioritize concise, context-specific assessments to minimize participant burden (Wise & Schwarz, 2017). Similar single-item difficulty measures have been used in CT research to gauge task perceptions (Román-González et al., 2019).

MetaCog (Metacognitive Strategies): Metacognitive strategies were assessed via self-report scales, using a nine-item subscale adapted from the Motivated Strategies for Learning Questionnaire (MSLQ; Pintrich et al., 1993; $\alpha = 0.80$), specifically tailored to focus on planning aspects of metacognition in the context of pre-service teachers preparing lessons on complex CT concepts, such as coding or algorithmic problem-solving. An example item is: "When planning a lesson on complex concepts, I set specific goals for what I want students to learn" (1 = strongly disagree to 7 = strongly agree). Think-aloud protocols or log data were not used to ensure consistency with the survey-based assessment. We deliberately adapted the scale to emphasize planning strategies because lesson planning is a core activity that requires anticipating and structuring CT instruction for K–12 students. Planning aligns closely with CT's emphasis on algorithmic thinking and decomposition, which involve upfront goal-setting and sequencing, skills central to designing effective CT-integrated lessons. Monitoring and evaluating, by contrast, are more reactive processes that typically occur during (e.g., in-class implementation) or after actual teaching experiences requiring observational methods (e.g., think-aloud protocols), which were not feasible and beyond the scope of this cross-sectional study.

4.3. Data Analysis

LPA was used to identify CT proficiency profiles based on standardized digital literacy (CT_DL), problem-solving (CT_PS), and coding comfort (CT_CD) scores, using a person-centred approach for nuanced heterogeneity (Ferguson et al., 2020). Model selection was based on statistical fit indices and theoretical expectations of CT skill variability. Statistical criteria included lower AIC / BIC for better fit, entropy ≥ 0.80 for clearer profile delineation, significant BLRT ($p < .05$) for model improvement, and profile sizes > 5% of the sample to ensure interpretability. Theoretical considerations anticipated distinct proficiency levels aligned with Shute et al.'s (2017) CT model, emphasizing decomposition, abstraction, and algorithmic thinking, as well as Román-González et al.'s (2019) non-linear CT development in pre-service teachers.

CT_DL, CT_PS, and CT_CD were standardized (z-scores) before LPA to ensure variable comparability. Shapiro-Wilk tests assessed the normality of key variables to indicate normality levels, as well as *LPA*'s robustness to moderate non-normality, thus minimizing clustering bias (Ferguson et al., 2020). Linear discriminant analysis (LDA) was conducted to examine how metacognitive strategies (MetaCog) and coding experience (CodExp) predict profile membership. Finally, a one-way analysis of variance (ANOVA) was applied to assess differences in online behaviours (PG_Time, L_Time, L_Clicks) across profiles. Non-normality may influence clustering by skewing profile assignments, but standardization and robust fit indices helped ensure a reliable profile analysis.

5. Results

Table 1 presents the means, standard deviations, and correlations for study variables, revealing key relationships.

Table 1. Means, Standard Deviations, and Correlations of Study Variables

Variables	*M* (*SD*)	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.
1. Gender	--	--											
2. BEd	--	.136	--										
3. CodExp	--	–.011	–.159	--									
4. MetaCog	5.84 (0.74)	–.007	.055	.111	--								
5. CT_DL	5.12 (1.21)	–.061	.156	$.222^{*}$	$.216^{*}$	--							
6. CT_PS	5.24 (0.89)	–.023	$.182^{*}$	.162	$.448^{***}$	$.582^{***}$	--						
7. CT_CD	2.71 (1.36)	.062	$.275^{**}$	$.285^{**}$	.103	$.423^{***}$	$.493^{***}$	--					
8. PG_Time	64.27 (33.68)	–.039	$–.215^{*}$	$.177^{*}$	.139	–.096	.092	.077	--				
9. L_Clicks	50.07 (31.27)	–.051	–.011	$–.251^{**}$	.186	–.083	–.087	–.188	–.015	--			
10. L_Time	124.33 (108.62)	–.032	–.158	–.176	.055	–.064	–.051	–.147	.110	$.481^{***}$	--		
11. L_Score	1.39 (0.85)	.081	.080	$.230^{*}$	–.161	–.029	.080	$.229^{*}$	$.295^{**}$	$–.512^{***}$	$–.211^{*}$	--	
12. L_Diff	4.93 (3.45)	.018	–.127	$–.215^{*}$	.168	–.154	$–.308^{**}$	$–.320^{***}$	–.097	$.474^{***}$	$.243^{*}$	$–.673^{***}$	--

Note: Gender: 1 = Male, 2 = Female, 3 = Other;
BEd: 1 = Elementary, 2 = Secondary, 3 = Other (Physical Education, Music, TESL);
CodExp: 0 = No coding experience, 1 = With coding experience; MetaCog: Metacognitive strategies;
CT_DL: Digital literacy skills; CT_PS: Problem-solving skills; CT_CD: Coding comfort;
PG_Time: Playground time (seconds); L_Clicks: Clicks during task; L_Time: Task completion time (seconds);
L_Score: Task score (0 = incorrect, 1 = correct on second attempt, 2 = correct on first attempt);
L_Diff: Perceived difficulty (1 = very easy, 10 = very hard).
* $p < .05$, ** $p < .01$, *** $p < .001$

Prior coding experience (CodExp) positively correlated with digital literacy (CT_DL, $r = .22$, $p < .05$) and coding comfort (CT_CD, $r = .29$, $p < .01$), but negatively correlated with perceived task difficulty (L_Diff, $r = -.22$, $p < .05$), suggesting coding experience enhances CT skills and reduces task-related cognitive load. Perceived difficulty (L_Diff) negatively correlated with problem-solving skills (CT_PS, $r = -.31$, $p < .01$) and coding comfort (CT_CD, $r = -.32$, $p < .001$), indicating higher CT skills ease task perception. Metacognitive strategies (MetaCog) were strongly correlated with problem-solving skills (CT_PS, $r = .45$, $p < .001$), underscoring the role of reflective thinking in CT. Behavioural metrics showed that more clicks (L_Clicks) negatively correlated with task performance (L_Score, $r = -.51$, $p < .001$), reflecting greater efficiency among higher performers, whereas playground time (PG_Time) positively correlated with L_Score ($r = .30$, $p < .01$), highlighting practice benefits.

5.1. LPA Model Selection and Profile Characteristics

LPA on standardized CT_DL, CT_PS, and CT_CD scores identified a three-profile model (Novice, 34%, $n = 43$; Developing, 45%, $n = 58$; Proficient, 21%, $n = 27$) as optimal, based on statistical fit indices and theoretical alignment (Shute et al., 2017; Román-González et al., 2019). Models with one to five profiles were tested; the three-profile model showed substantial decreases in AIC (995.93) and BIC (1035.85) compared to the two-profile model (AIC = 1025.12, BIC = 1055.67), high entropy (.84, threshold ≥ .80), significant BLRT ($p = .01$), and interpretable profile sizes (> 5%). The four-profile model included a small profile (3%, $n = 4$) with marginal BLRT ($p = .06$), reducing interpretability. Departures from normality in CT_DL ($W = .968$, $p = .002$) and CT_PS ($W = .964$, $p = .001$) were mitigated by LPA's robustness and standardization, ensuring reliable clustering and clear profile delineation (Ferguson et al., 2020).

A Chi-square test of independence was conducted on the three-profile model to examine the relationship between cluster membership and year of study. The association was not statistically significant, $\chi^2(8, N = 128) = 4.24$, $p = .835$ (Cramér's $V = .13$), indicating that students' year of study did not differ significantly across the clusters. The ANOVA confirmed significant differences across profiles for CT_DL [$F(2, 125) = 90.969$, $p < .001$], CT_PS [$F(2, 125) = 75.508$, $p < .001$], and CT_CD [$F(2, 125) = 107.373$, $p < .001$].

Figure 1 shows the profile plot of standardized scores for digital literacy (DL), problem-solving (PS), and coding comfort (CD) across the three profiles proposed by the LPA.

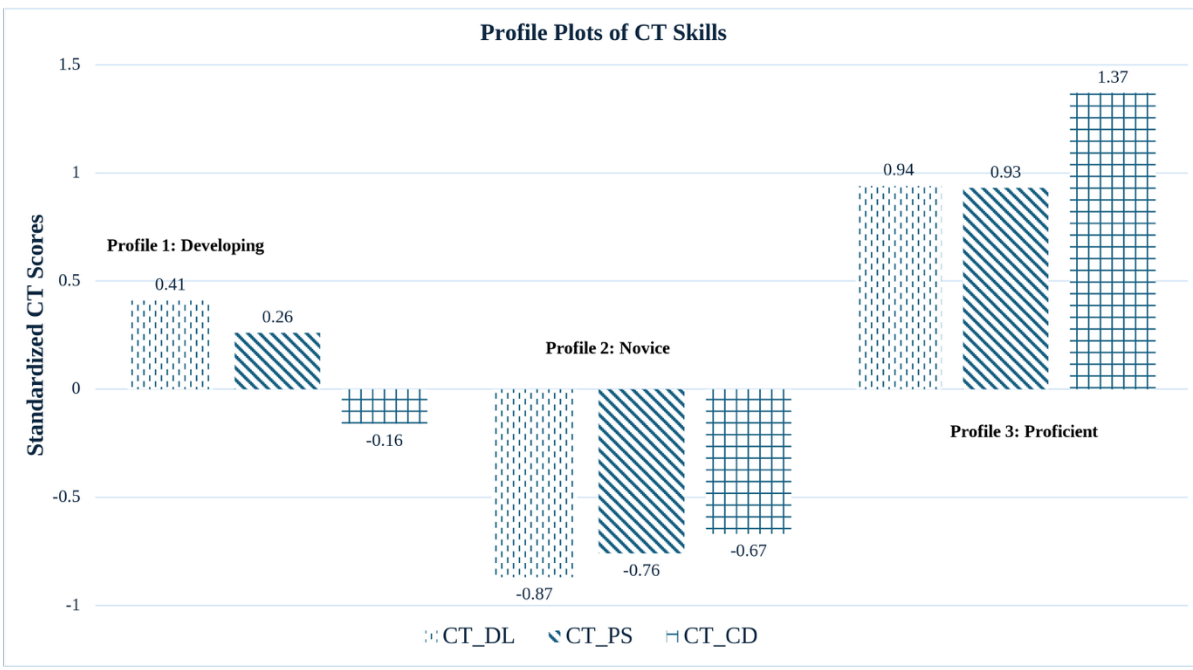

Figure 1. Profile Plots of Pre-service Teachers' CT Skills

5.1.1. Profile 1: Developing (n = 58)

This group (7 males, 45 females, 5 others, 1 missing) included students from Elementary ($n = 34$), Secondary ($n = 16$), and other programs ($n = 8$). They spent 30.14 seconds on the playground, with 53.1% completing the task on the first attempt, 16.3% on the second, and 30.6% incorrect. They showed moderate CT skills: DL ($M = 5.53$, $SD = 0.74$), PS ($M = 5.45$, $SD = 0.46$), and CD ($M = 2.49$, $SD = 0.87$), indicating intermediate skill application.

5.1.2. Profile 2: Novice (n = 43)

Comprising 3 males, 39 females, and 1 missing, this group included Elementary ($n = 31$), Secondary ($n = 8$), and other ($n = 4$) students. They spent 39.93 seconds on the playground, with 63.6% correct on the first attempt, 12.1% on the second, and 24.3% incorrect. They exhibited below-average CT skills: DL ($M = 3.88$, $SD = 0.83$), PS ($M = 4.40$, $SD = 0.76$), and CD ($M = 1.76$, $SD = 0.69$), reflecting basic CT understanding.

5.1.3. Profile 3: Proficient (n = 27)

This group (7 males, 17 females, 2 missing) included Secondary ($n = 13$), Elementary ($n = 8$), and other ($n = 6$) students. They spent 70.80 seconds on the playground, with 79.2% correct on the first attempt, 12.5% on the second, and 8.3% incorrect. They demonstrated above-average CT skills: DL ($M = 6.22$, $SD = 0.76$), PS ($M = 6.14$, $SD = 0.61$), and CD ($M = 4.70$, $SD = 0.95$), suggesting advanced CT proficiency for teaching contexts.

5.2. Predictors and Behavioural Outcomes

The assumption of equal covariance matrices for LDA was met with Box's $M = 4.747$, $p = .593$. MetaCog ($M = 5.73$, $SD = .84$) and CodExp significantly predicted Proficient profile membership (Wilks' $\lambda = .62$, $p < .001$), with MetaCog contributing more strongly (standardized coefficient = .78).

A one-way ANOVA indicated significant differences in online behaviours across profiles (see Table 2). Participants were randomly assigned to difficulty levels (24–30% per level) for the CT task completion. For PG_Time ($M = 64.77$, $SD = 37.09$, $n = 127$), $F(2, 124) = 8.32$, $p < .001$, $\eta^2 = .12$, LSD post-hoc tests showed that Proficient learners ($M = 45.21$, $SD = 25.30$) spent less time in the playground than Novice ($M = 85.67$, $SD = 45.10$, $p < .001$) and Developing ($M = 65.43$, $SD = 35.20$, $p = .012$), suggesting greater initial familiarity.

For L_Time ($M = 124.33$, $SD = 73.27$, $n = 99$), $F(2, 96) = 10.45$, $p < .001$, $\eta^2 = .18$, Proficient learners ($M = 80.12$, $SD = 40.10$) completed tasks faster than Novice ($M = 150.45$, $SD = 80.20$, $p < .001$) and Developing ($M = 125.67$, $SD = 70.50$, $p = .002$), indicating greater efficiency in CT.

For L_Clicks ($M = 50.07$, $SD = 37.23$, $n = 99$), $F(2, 96) = 7.89$, $p < .001$, $\eta^2 = .14$, Proficient learners ($M = 35.89$, $SD = 20.40$) had fewer clicks than Novice ($M = 65.34$, $SD = 45.60$, $p < .001$) and Developing ($M = 52.12$, $SD = 35.30$, $p = .008$), possibly reflecting reduced trial-and-error. These differences align with RQ3, highlighting behavioural efficiency among higher CT profiles.

Finally, the Kruskal-Wallis test confirmed differences in L_Diff, $H = 12.34$, $p < .01$, with Proficient learners reporting lower perceived difficulty ($M = 2.50$, $SD = 1.20$) than Novice learners ($M = 4.10$, $SD = 1.50$).

Table 2. Means and Standard Deviations for Online Behaviours

Variable	Profile	*M*	*SD*	95% CI for Mean
PG_Time (in seconds)	1: Developing	62.28	30.41	[54.13, 70.42]
	2: Novice	60.15	38.75	[47.92, 72.38]
	3: Proficient	70.80	30.11	[58.89, 82.71]
L_Time (in seconds)	1: Developing	131.42	72.92	[110.24, 152.59]
	2: Novice	106.65	50.79	[87.69, 125.62]
	3: Proficient	87.91	55.44	[64.50, 111.31]
L_Clicks	1: Developing	49.81	27.13	[41.93, 57.69]
	2: Novice	51.70	28.07	[41.22, 62.18]
	3: Proficient	38.75	28.80	[26.59, 50.91]
L_Diff (1–10 scale)	1: Developing	5.42	3.21	[4.49, 6.35]
	2: Novice	5.90	3.60	[4.55, 7.25]
	3: Proficient	2.75	2.82	[1.56, 3.94]

6. Discussion

This study examined variations in pre-service teachers' CT skills, the predictive role of prior coding experience and metacognitive strategies, and how these skills influence online CT task behaviour. The findings revealed significant differentiation among pre-service teachers' CT skills based on a three-profile model identified through LPA: Novice (34%, $n = 43$), Developing (45%, $n = 58$), and Proficient (21%, $n = 27$). The Proficient profile displayed above-average levels of digital literacy (CT_DL, $M = 6.22$, $SD = 0.76$), problem-solving (CT_PS, $M = 6.14$, $SD = 0.61$), and coding comfort (CT_CD, $M = 4.70$, $SD = 0.95$), which translated into higher performance scores, lower perceived difficulty (L_Diff, $M = 2.50$, $SD = 1.20$), and greater efficiency (e.g., lower task completion time, L_Time, $M = 80.12$, $SD = 40.10$; fewer clicks, L_Clicks, $M = 35.89$, $SD = 20.40$). This corroborates the hypothesis that higher CT skills enhance problem-solving efficiency and ease cognitive load (Cheng et al., 2023; Wu et al., 2024; Barr & Stephenson, 2011).

The results also highlighted the critical role of prior coding experience and metacognitive strategies in predicting CT proficiency. Coding experience positively correlated with digital literacy (CT_DL) and coding comfort (CT_CD), while metacognition strongly correlated with problem-solving (CT_PS). LDA confirmed the latter as significant predictors of CT profile membership. Additionally, preparatory engagement (PG_Time) correlated positively with task performance (L_Score, $r = .30$, $p < .01$), underscoring the benefits of experiential learning (Argelagós et al., 2022). The multimodal approach, combining survey and trace data, provided comprehensive insights into these dynamics, responding to calls for analytics-informed assessments that bridge research and practice in learning analytics (Chaabi et al., 2025; Knight et al., 2017). Overall, these findings reveal heterogeneity in CT skills, with Proficient learners demonstrating lower trial-and-error and cognitive load than Novice and Developing learners.

6.1. Theoretical Implications

The study's findings extend Shute et al.'s (2017) CT model by incorporating metacognitive strategies and coding experience as key predictors, often underexplored in pre-service teacher education. The three-profile solution aligns with the model's emphasis on varying proficiency in decomposition, abstraction, algorithmic thinking, and pattern recognition: Novice profiles reflect basic CT understanding with deficits; Developing profiles reflect intermediate application; and Proficient profiles reflect more advanced generalization. This mapping validates the model while highlighting metacognition and coding as factors that bridge technical proficiency and reflective practice, thereby offering a holistic view of CT development (Pintrich et al., 1993; Ramirez-Salgado et al., 2025).

The profiles also provide empirical support for Román-González et al.'s (2019) assertion that CT acquisition is non-linear and context-dependent, capturing uneven progression influenced by individual experiences. This profiling approach is novel, offering a replicable framework for assessing CT readiness and addressing critiques of oversimplified assessments (Grover & Pea, 2013). Consequently, the study sets a precedent for multimodal analytics in teacher education, enriching theoretical understandings of CT-related interactions (Liao et al., 2022; Aldemir et al., 2025).

6.2. Practical Implications

This study leads to multiple practical implications. First, integrating CT tasks into pre-service teacher education can serve as a valuable diagnostic and developmental tool for educational practice and policy development. The clear delineation in CT skill levels among pre-service teachers suggests a need for differentiated instructional strategies that cater to the specific needs of each group, thereby enhancing overall CT proficiency before these individuals enter the professional field. These findings also underscore the importance of personalized learning interventions tailored to enhance coding experience and foster metacognitive abilities, as these have been shown to play a pivotal role in predicting CT skills. For instance, Novice learners can benefit from scaffolded coding workshops using tools like Scratch, while Proficient learners can engage in advanced projects, aligning with Liu et al.'s (2024) emphasis on personalized CT interventions for diverse skill levels.

Furthermore, given the demonstrated importance of coding experience and metacognitive strategies, integrating metacognitive training, such as reflective journaling or self-assessment, into CT courses can increase problem-solving efficiency, particularly for Developing learners. The strong predictive role of metacognition supports Pintrich et al.'s (1993) findings that metacognitive strategies enhance learning outcomes in complex tasks. This approach could improve CT skills uniformly across different learner profiles, equipping future educators with the necessary skills to tackle digital tasks and implement technology-driven pedagogies effectively in their teaching practices.

From a practical standpoint, this study also underscores the need for teacher education programs to adopt a CT-focused coursework, incorporating hands-on coding and adaptive learning technologies tailored to individual skill levels, as evidenced by the Proficient profile's efficiency (e.g., lower L_Time, $M = 80.12$, $SD = 40.10$, $F(2, 96) = 10.45$, $p < .001$). For instance, programs could incorporate project-based learning activities that require pre-service teachers to reflect on their problem-solving processes, such as journaling or peer feedback sessions. Tools like Scratch or robotics kits could be used to provide hands-on coding experience, while metacognitive strategies could be reinforced through guided reflection prompts or self-assessment checklists. Moreover, policymakers can allocate resources for PD workshops, as recommended by Dong et al. (2024), ensuring pre-service teachers are better prepared to create learning environments that encourage algorithmic thinking, abstraction, and problem-solving across disciplines, ultimately contributing to a more computationally literate society.

Finally, the ethical use of learning analytics—prioritizing privacy and consent—is critical given the variability in behavioural data across profiles (e.g., Proficient efficiency: fewer L_Clicks; $M = 35.89$, $SD = 20.40$ vs. Novice: $M = 65.34$, *SD* = 45.60). This aligns with Ferguson's (2014) framework and Prinsloo and Slade's (2017) emphasis on transparent data practices to ensure equitable CT development, particularly for Novice learners with limited coding knowledge, supporting fair and inclusive teacher training.

As a future direction, a learning analytics dashboard could be developed to visualize CT profiles and behavioural metrics, such as time spent on task completion and perceived task difficulty, to provide real-time feedback for teacher educators (Verbert et al., 2013; Wise & Schwarz, 2017). Such a dashboard, while conceptual at this stage, could identify Novice learners needing support and Proficient learners ready for advanced tasks, building on the empirical profiles and trace data from this study.

6.3. Limitations and Future Directions

While this study provides valuable insights, it is not without limitations. First, the reliance on self-reported data for CT skills and metacognitive strategies may introduce bias, as participants' perceptions may not always align with their actual abilities. Future studies could complement self-reports with performance-based assessments to provide a more objective measure of CT proficiency. Second, the study was conducted within a single institution and with a predominantly Year 4 pre-service teacher sample, potentially limiting generalizability beyond the Canadian teacher education context and to earlier-year students with less CT exposure. Replicating this study in diverse educational contexts, including different countries or regions, would help determine the extent to which these findings apply across settings. Third, the study's reliance on time-on-task metrics (e.g., PG_Time, L_Time) as behavioural traces, while aligned with learning analytics, has limitations. These aggregates, captured via Qualtrics, may not distinguish between active problem-solving, idle time, or interface navigation, potentially confounding interpretations of cognitive engagement (Wise & Schwarz, 2017). For example, longer L_Time or high SDs (e.g., L_Time *SD* = 108.62) could reflect deliberate reflection or distraction during task engagement, limiting the granularity of our behavioural insights. Future studies could leverage finer-grained event logging (e.g., keystroke-level traces) or integrate advanced analytics, such as eye-tracking, to capture qualitative differences in engagement and enhance the precision of learning analytics applications. Finally, the cross-sectional design of the study precludes causal inferences about the relationship between coding experience, metacognitive strategies, and CT skills. Longitudinal studies tracking pre-service teachers' CT development over time, with a more than a single-task design, could provide deeper insights into how these factors interact and evolve. Additionally, experimental studies could test the effectiveness of specific interventions, such as metacognitive training modules or coding bootcamps, in enhancing CT skills. Exploring the role of cultural and contextual factors in CT development could also yield valuable insights, particularly in non-Western educational systems where CT integration may be at an earlier stage.

7. Conclusion

This study advances the field of learning analytics by providing a model for profiling and predicting CT skills among pre-service teachers. By extending Shute et al.'s (2017) CT model through the inclusion of metacognitive strategies and prior coding experience, the study uses LPA and LDA to identify three distinct CT profiles: Novice, Developing, and Proficient. Behavioural data from the *Programming Lamps* task mapped to Shute et al.'s CT components, with Proficient profiles showing greater task efficiency. These contributions align with research advocating for data-driven CT profiling and PD to enhance teacher readiness (Chichekian et al., 2025; Ramirez-Salgado et al., 2025).

Our findings further demonstrate how multimodal data (surveys, behavioural traces) can inform personalized pedagogical interventions, such as adaptive training modules or dashboards to visualize CT profiles and behavioural metrics for real-time feedback (Verbert et al., 2013). This model has potential for scalability to other educational contexts, including K–12 students and in-service teachers, where CT is increasingly vital. By bridging technical proficiency and reflective practice, this study supports analytics-driven teacher preparation, equipping educators to foster a computationally literate society in a technology-driven world (Ferguson, 2014).

Declaration of Conflicting Interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The publication of this article received financial support from the Social Sciences and Humanities Research Council (Grant # 430-2021-00707).

References

Aldemir, T., Moon, J., Bicer, A., Byun, G., Perez, C., & Sabanwar, V. (2025). How non-CS pre-service teachers conceptualize and engage with computational thinking through 3D game design: Insights from an XR-enhanced maker experience. *Computers & Education: X Reality, 6*, Article 100099. https://doi.org/10.1016/j.cexr.2025.100099

Argelagós, E., Garcia, C., Privado, J., & Wopereis, I. (2022). Fostering information problem solving skills through online task-centred instruction in higher education. *Computers & Education, 180*, Article 104433. https://doi.org/10.1016/j.compedu.2022.104433

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads, 2*(1), 48–54. https://doi.org/10.1145/1929887.1929905

Bers, M. U. (2017). *Coding as a playground: Programming and computational thinking in the early childhood classroom*. Routledge. https://doi.org/10.4324/9781315398945

Bower, M., & Falkner, K. (2015). Computational thinking, the notional machine, pre-service teachers, and research opportunities. In *Proceedings of the 17th Australasian Computing Education Conference (ACE 2015)* (pp. 37–46). Australian Computer Society.

Butler, D., & Leahy, M. (2021). Developing pre-service teachers' understanding of computational thinking: A constructionist approach. *British Journal of Educational Technology, 52*(3), 1060–1077. https://doi.org/10.1111/bjet.13090

Chaabi, H., Azmani, A., & Dodero, J. M. (2025). Bridging computational thinking research and practice of pre-service and in-service teachers. *Humanities and Social Sciences Communications, 12*, Article 666. https://doi.org/10.1057/s41599-025-04895-9

Chan, S.-W., Looi, C.-K., Ho, W. K., Huang, W., Seow, P., & Wu, L. (2021). Learning number patterns through computational thinking activities: A Rasch model analysis. *Heliyon, 7*(9), Article e07922. https://doi.org/10.1016/j.heliyon.2021.e07922

Cheng, Y.-P., Lai, C.-F., Chen, Y.-T., Wang, W.-S., Huang, Y.-M., & Wu, T.-T. (2023). Enhancing student's computational thinking skills with student-generated questions strategy in a game-based learning platform. *Computers & Education, 200*, Article 104794. https://doi.org/10.1016/j.compedu.2023.104794

Chichekian, T., Cutumisu, M., & Savard, A. (2025). Preservice teachers' computational thinking profiles. In A. Rajala, A. Cortez, R. Hofmann, A. Jornet, H. Lotz-Sisitka, & L. Markauskaite (Eds.), *Proceedings of the 19th International Conference of the Learning Sciences* (ICLS 2025), 10–13 June 2025, Helsinki, Finland (pp. 47–55). International Society of the Learning Sciences.

Cutumisu, M., Adams, C., Yuen, C., Hackman, L., Lu, C., & Samuel, M. (2019). Callysto Computational Thinking Test (CCTt) Teacher Version [Measurement instrument]. https://callysto.ca

Dagienė, V., & Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. In R. T. Mittermeir & M. M. Sysło (Eds.), *Informatics education - Supporting computational thinking* (pp. 19–30). Springer. https://doi.org/10.1007/978-3-540-69924-8_2

Dong, W., Li, Y., Sun, L., & Liu, Y. (2024). Developing pre-service teachers' computational thinking: A systematic literature review. *International Journal of Technology and Design Education, 34*(1), 191–227. https://doi.org/10.1007/s10798-023-09811-3

Ferguson, R. (2014). Learning analytics: Drivers, developments and challenges. *Italian Journal of Educational Technology*, *22*(3), 138–147. https://www.learntechlib.org/p/183381/

Ferguson, S. L., Moore, E. W. G., & Hull, D. M. (2020). Finding latent groups in observed data: A primer on latent profile analysis in Mplus for applied researchers. *International Journal of Behavioral Development, 44*(5), 458–468. https://doi.org/10.1177/0165025419881721

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

Hoyles, C., & Noss, R. (2015). A computational lens on design research. *ZDM Mathematics Education, 47*(6), 1039–1045. https://doi.org/10.1007/s11858-015-0731-2

Jaipal-Jamani, K., & Angeli, C. (2017). Effect of robotics on elementary pre-service teachers' self-efficacy, science learning, and computational thinking. *Journal of Science Education and Technology, 26*(2), 175–192. https://doi.org/10.1007/s10956-016-9663-z

Knight, S., Wise, A. F., Ochoa, X., & Hershkovitz, A. (2017). Learning analytics: Looking to the future. *Journal of Learning Analytics, 4*(2), 1–5. https://doi.org/10.18608/jla.2017.42.1

Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A pedagogical framework for computational thinking. *Digital Experiences in Mathematics Education, 3*(2), 154–171. https://doi.org/10.1007/s40751-017-0031-2

Liao, C. H., Chiang, C.-T., Chen, I.-C., & Parker, K. R. (2022). Exploring the relationship between computational thinking and learning satisfaction for non-STEM college students. *International Journal of Educational Technology in Higher Education, 19*(1), Article 43. https://doi.org/10.1186/s41239-022-00347-5

Liu, Z., Gearty, Z., Richard, E., Orrill, C. H., Kayumova, S., & Balasubramanian, R. (2024). Bringing computational thinking into classrooms: A systematic review on supporting teachers in integrating computational thinking into K-12 classrooms. *International Journal of STEM Education, 11*(1), Article 51. https://doi.org/10.1186/s40594-024-00510-6

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), 61–76. https://doi.org/10.14742/ajet.3521

Pintrich, P. R., Smith, D. A. F., Garcia, T., & McKeachie, W. J. (1993). Reliability and predictive validity of the Motivated Strategies for Learning Questionnaire (MSLQ). *Educational and Psychological Measurement, 53*(3), 801–813. https://doi.org/10.1177/0013164493053003024

Prinsloo, P., & Slade, S. (2017). An elephant in the learning analytics room: The obligation to act. In *Proceedings of the Seventh International Learning Analytics & Knowledge Conference* (pp. 46–55). ACM Press. https://doi.org/10.1145/3027385.3027406

Ramirez-Salgado, A., Wusylko, C., Weisberg, L., Delgado, J., & Israel, M. (2025). Preparing elementary pre-service teachers to integrate computational thinking in the curriculum. *ACM Transactions on Computing Education, 24*(4), Article 42. https://doi.org/10.1145/3744751

Román-González, M., Moreno-León, J., & Robles, G. (2019). Combining assessment tools for a comprehensive evaluation of computational thinking interventions. In S.-C. Kong & H. Abelson (Eds.), *Computational thinking education* (pp. 79–98). Springer. https://doi.org/10.1007/978-981-13-6528-7_6

Sadik, O., Ottenbreit-Leftwich, A., & Nadiruzzaman, H. (2017). Computational thinking conceptions and misconceptions: Progression of pre-service teacher thinking during computer science lesson planning. In P. J. Rich & C. B. Hodges (Eds.), *Emerging research, practice, and policy on computational thinking* (pp. 221–238). Springer. https://doi.org/10.1007/978-3-319-52691-1_14

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*, 142–158. https://doi.org/10.1016/j.edurev.2017.09.003

Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. *Computers & Education, 148*, Article 103798. https://doi.org/10.1016/j.compedu.2019.103798

Verbert, K., Duval, E., Klerkx, J., Govaerts, S., & Santos, J. L. (2013). Learning analytics dashboard applications. *American Behavioral Scientist, 57*(10), 1500–1509. https://doi.org/10.1177/0002764213479363

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*(1), 127–147. https://doi.org/10.1007/s10956-015-9581-5

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35. https://doi.org/10.1145/1118178.1118215

Wise, A. F., & Schwarz, B. B. (2017). Visions of CSCL: Eight provocations for the future of the field. *International Journal of Computer-Supported Collaborative Learning, 12*(4), 423–467. https://doi.org/10.1007/s11412-017-9267-5

Wu, T.-T., Asmara, A., Huang, Y.-M., & Permata Hapsari, I. (2024). Identification of problem-solving techniques in computational thinking studies: Systematic literature review. *SAGE Open, 14*(2). https://doi.org/10.1177/21582440241249897

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*, 565–568. https://doi.org/10.1007/s11528-016-0087-7