ACTA DIDACTICA NAPOCENSIA
Volume 18, Number 2, 2025 – DOI: 10.24193/adn.18.2.10

SHAPING MATHEMATICS TEACHERS' COMPETENCIES AND INSTRUCTIONAL INSIGHTS THROUGH COMPUTATIONAL THINKING PROFESSIONAL DEVELOPMENT

Feyza KURBAN, Hüseyin Bahadır YANIK,
Ayşegül ERYILMAZ ÇEVİRGEN, Burcu DURMAZ

Abstract: This study explores the impact of a structured computational thinking (CT)-focused professional development (PD) program on mathematics teachers' competencies and instructional insights. Designed as a practice-oriented initiative, the PD program primarily employed block-based coding activities while also incorporating modeling-based, hybrid (block-text), and introductory text-based coding tasks to support progressive development of CT skills within mathematics education. A mixed-methods design was employed with 28 mathematics teachers attending a six-module program on integrating CT into instructional practice. Quantitative data were collected using three instruments: the Programming-Oriented CT Scale, the Teacher Beliefs About Coding and CT Scale, and the TPACK-Math Scale, administered before and after the training. Semi-structured interviews were conducted with 12 selected participants to examine how the training shaped their pedagogical competence, instructional approaches, and perceived barriers. Quantitative results showed gains in CT (conceptual knowledge, algorithmic thinking), self-efficacy, pedagogical efficacy, and technology-related competencies. Thematic analysis identified three core themes: the impact of professional development (e.g., overcoming initial fears, increased student engagement, personalized resource creation), strategies for successful integration (e.g., early CT exposure, stepwise coding progression, building teacher competence), and challenges to implementation (e.g., curricular limitations, insufficient resources, resistance to change, infrastructural barriers).

Keywords: Computational Thinking, Mathematics Teacher Education, Professional Development, CT Integration in Mathematics Education, Mathematics Teachers' Reflections on CT

Mathematics Subject Classification (2010): 97B50, 97C70, 97U70

1. Introduction

The idea of integrating computational thinking (CT) into mathematics throughout K–12 education is increasingly gaining support from teacher training programs, policymakers, curricula, and researchers, and is widely regarded as an exploratory and practical approach that enables students to expand their understanding of mathematical concepts and achieve a deeper, more nuanced mastery of the discipline (DeCoito, 2024; Kite & Park, 2022; Sands et al., 2018; Weintrop et al., 2016; Zhan et al., 2024). This integration relies on effective teacher preparation and professional development (PD) that equips future teachers with robust content knowledge, innovative pedagogical strategies, and frameworks for meaningful curricular integration, thereby empowering them to seamlessly infuse CT into subject-specific instruction and foster interdisciplinary learning (Yadav et al., 2017). Nevertheless, despite this growing consensus, recent findings (Kite & Park, 2022) reveal that significant barriers—stemming from insufficient teacher conceptualizations of CT, limited PD opportunities, and concerns about student preparedness—continue to hinder the meaningful adoption of CT in mathematics classrooms. In response to these challenges, this study investigates the outcomes of a hands-on coding program designed to leverage CT to enhance mathematical reasoning and connect theoretical principles to real-life applications by developing core CT skills such as decomposition, pattern recognition, algorithmic thinking, generalization, sequencing and logical reasoning through various coding activities.

Received April 2025.

Cite as: Kurban, F; Yanık, H. B., Eryılmaz-Çevirgen, & Durmaz, B. (2025). Shaping Mathematics Teachers' Competencies and Instructional Insights Through Computational Thinking Professional Development. *Acta Didactica Napocensia,* 18(2), 105-125, https://doi.org/10.24193/adn.18.2.10

Recognized as a transdisciplinary cognitive approach, CT equips learners with the ability to tackle complex problems, automate processes, and think systematically by formulating problems and expressing their solutions in forms that can be executed by an information-processing agent, such as a computer (Palop et al., 2025; Wing, 2011). These abilities not only enhance mathematical understanding, but also promote creativity, collaborative thinking, and flexible problem-solving across various domains of learning (Weintrop et al., 2016). Moreover, it functions as a boundary object that bridges abstract mathematical reasoning with concrete programming practices, enabling students to navigate between disciplines through meaningful and integrated problem-solving experiences (Ng, Leung, & Ye, 2023). Although CT is widely acknowledged for its pedagogical potential and cognitive value, its meaningful integration into mathematics classrooms depends largely on teachers' ability to grasp its core principles, tailor them to subject-specific contexts, and transform them into coherent and engaging instructional practices. The pivotal role of teachers in this domain has led researchers to investigate how PD initiatives influence CT-related teacher competencies—such as skills, beliefs, self-efficacy, and their capacity to effectively integrate CT into mathematics classrooms. However, the transition from theoretical value to classroom impact is far from automatic, as it requires intentional and structured professional development to equip teachers with the necessary tools, confidence, and pedagogical insight to integrate CT meaningfully into mathematics instruction.

While prior research has documented modest gains in teachers' CT awareness through short-term, often unstructured PD initiatives, such efforts have typically emphasized surface-level engagement with CT concepts, offering limited attention to pedagogical reasoning or instructional design (Bower et al., 2017; Yadav et al., 2017). In contrast, Jocius et al. (2020) implemented a more extensive and structured PD program for middle school mathematics teachers, combining coding tutorials, collaborative lesson design, and classroom-based applications with a block-based programming environment. This program contributed to the development of foundational CT skills and strengthened teachers' confidence in applying CT within discipline-specific instruction. Saxena and Chiu (2022) introduced a focused, curriculum-based PD initiative for in-service preschool teachers, which supported growth in CT knowledge, pedagogical beliefs, and teaching self-efficacy, shaped by contextual factors such as school culture and technological resources. Expanding on these approaches, Suters and Suters (2020) developed an intensive, multi-faceted PD experience for middle school mathematics teachers that integrated programming through Bootstrap Algebra and robotics with Lego Mindstorms. The program promoted improvements in mathematics-specific CT understanding and TPACK, while encouraging collaborative, hands-on engagement with CT-infused mathematical practices.

In alignment with these studies, the current research offers a more comprehensive contribution by examining how a structured, practice-oriented training program—designed to integrate CT into mathematics education—can illuminate shifts in teachers' programming-oriented CT skills, their beliefs about coding and CT (value, self-efficacy, and teacher efficacy), and their TPACK for Mathematics (technological, pedagogical, content, contextual, and integrated knowledge). What distinguishes this study from earlier work is its focus on teachers' post-training reflections, particularly their perceptions of pedagogical competence, the strategies they propose for classroom implementation, and the contextual challenges they anticipate. To address these aims, the study poses the following research questions (RQs):

RQ1. How does the training program impact participants' programming-oriented CT skills in terms of evaluation, conceptual knowledge, and algorithmic thinking?

RQ2. How does it affect their beliefs about coding and CT in terms of value, self-efficacy, and teacher efficacy?

RQ3. How does it influence their TPACK for Mathematics, particularly the technological, pedagogical, content, contextual, and integrated knowledge components?

RQ4. How has PD training influenced participants' perceptions of their pedagogical competence regarding the integration of CT into mathematics education?

RQ5. What strategies do participants suggest for embedding CT into mathematics instruction following the training?

RQ6. What factors do they identify as potential barriers to CT integration after the training?

2. Methods

Employing a mixed-methods research design, this study strategically integrated quantitative and qualitative data to capture both the measurable and interpretive dimensions of mathematics teachers' learning experiences (Doorenbos, 2014). Conducted with a single sample, the study enabled a comprehensive exploration of shifts in participants' CT-related competencies and pedagogical perspectives throughout the structured professional development program. The use of multiple data sources enhanced internal validity through triangulation, contributing to the overall coherence and trustworthiness of the findings (Carter et al., 2014; Schoonenboom & Johnson, 2017).

2.1. Participants

The study sample consisted of 28 mathematics teachers (18 female, 10 male) from middle and high schools in the same geographic region, all of whom participated in a structured PD program and demonstrated a strong willingness to integrate CT into their instructional practices, despite differences in professional background and technological proficiency. Using a purposeful sampling strategy (Patton, 2002; Palinkas et al., 2015; Robinson, 2014), 12 participants were selected for semi-structured interviews based on their voluntary involvement, variation in pre- and post-test score changes (ranging from significant improvement to little or no change), and diversity in teaching level, to ensure a balanced and nuanced representation of perspectives on the PD program.

2.2. Training program overview

The PD program was delivered through six structured modules, emphasizing Scratch-based block coding while integrating robotic, modeling-based, hybrid block-text, and text-based coding activities. It aimed to develop participants' CT skills by focusing on core components such as pattern recognition, algorithmic thinking, logical reasoning, and sequencing, embedding these skills into mathematics education through coding-based exploration. By integrating CT into mathematics instruction, the program provided participants with practical strategies to use coding as an effective teaching tool through hands-on engagement and structured learning approaches.

Table 1. *Objectives and core areas of the teaching modules*

Modules	Description
Module 1:	A presentation on CT covered its fundamental concepts, processes, and applications in mathematics education. This was followed by unplugged coding instruction, which included both training and hands-on activities such as Pixel Coding and The Treasure Hunt, emphasizing and applying core coding principles.
Module 2:	A presentation on the integration of digitalization into mathematics education through coding was delivered, followed by an introduction to Scratch's code blocks, where participants explored their functions, including variables, before designing the "Know the Result" game using block-based coding.
Module 3:	Participants engaged in block-based coding, incorporating event-driven coordination in A Story from the History of Mathematics, where they utilized looks blocks along with the text-to-speech extension. They also worked with variables, lists, loops, conditionals, and motion blocks in Pattern Generator and Spirograph Design activities. They then transitioned to modeling-based coding, integrating similar CT concepts in Tower and Well Constructions, strengthening the connection between coding, engineering, and mathematical design.
Module 4:	Participants carried out Music and Cryptology and Visualizing Sound Waves activities in a block-based coding environment, assigning musical elements to sprites and utilizing the pen extension to represent sound waves. Later, in hybrid block-text coding, they engaged in "Which Polygon?" and "Is This Number Divisible?", applying the same CT concepts and processes from Spirograph Design and "Know the Result," while transitioning between block-based and text-based programming.
Module 5:	Participants designed line, bar, and circle chart drawing programs using block-based coding and

	familiar code block functions from previous activities. They then learned specific text-based coding techniques and transferred their block-based coding knowledge to text-based coding, applying these skills in the "How Should I Invest My Money in the Bank?" and "Saplings to Trees" activities.
Module 6:	Participants developed a block-based program to simulate the changing water level in a pool, where a faucet at the top fills the pool while another faucet at the bottom drains it, and to graphically represent these changes. This task involved coordinating sprites and constructing complex algorithms, creating a structured and dynamic design.

2.3. Data collection instruments and procedures

Quantitative data were collected using three established scales—first two with 33 items and the last one with 59—administered before and after the training to measure changes in participants' CT skills, coding-related beliefs, and technological pedagogical content knowledge in mathematics instruction, aiming to provide a comparable evaluation of their evolving perceptions, teaching confidence, and competence in integrating technology.

The Programming-Oriented CT Scale (P-CTS), consisting of three subscales, examined participants' self-perceived CT skills (Kılıç et al., 2021). The Evaluation (Ev) subscale identified participants' ability to analyze, debug, and optimize code by detecting errors and critically comparing solutions. Conceptual Knowledge (Ck) explored their understanding of fundamental programming principles, including loops, conditionals, functions, and data structures. Algorithmic Thinking (Al) emphasized structured problem-solving skills, focusing on decomposition, pattern recognition, abstraction, and algorithm design.

The Teacher Beliefs About Coding and CT Scale (TBaCCT) captured participants' perspectives on coding and CT across three subscales (Rich et al., 2021). The Values (Va) subscale focused on beliefs regarding coding's role in early education and curriculum integration. Self-Efficacy (Se) reflected participants' confidence in computing concepts, debugging, pattern recognition, logic, and algorithmic design, even without prior experience. Teacher Efficacy (Te) highlighted their ability to teach coding, support debugging, integrate coding into curricula, and design instruction aligned with real-world applications.

The Technological Pedagogical Content Knowledge (TPACK) scale for mathematics teachers (hereafter, TPACK-Math) was used to assess participants' knowledge across the scale's subscales (Önal, 2016). The Technology (T) subscale examined troubleshooting abilities, educational technology use, and hardware/software management. Pedagogy (P) explored instructional strategies, classroom management, assessment techniques, and the ability to adapt teaching to diverse learners. Content (C) evaluated proficiency in mathematical concepts, problem-solving, curriculum planning, and real-world applications. Technological Pedagogical (Tp), Technological Content (Tc), Pedagogical Content (Pc), and Technological Pedagogical Content (Tpc) subscales addressed the integration of technology, digital tools, and pedagogical strategies to enhance mathematics instruction. The Contextual (Cx) subscale considered teachers' awareness of classroom environments, student demographics, and technological accessibility.

To complement the quantitative data, qualitative data were collected through semi-structured interviews, which allowed researchers to keep the conversations focused on specific topics while also enabling participants to share unexpected insights. The interview protocol was carefully designed to examine mathematics teachers' perceptions of CT integration into mathematics education, along with its pedagogical, curricular, and practical implications. Over the course of the study, ten interview questions were posed to twelve participants, with each session lasting between 20 and 40 minutes. The interviews explored the impact of PD training on teachers' perceived competence (e.g., How has the training influenced your perspectives on integrating CT into mathematics classrooms?), effective strategies for CT integration (e.g., In what ways do you think CT can be embedded in mathematics instruction?), and barriers to implementation (e.g., What challenges might limit the integration of CT into mathematics education?). Furthermore, the interviews facilitated a deeper exploration of emergent themes and uncovered new lines of inquiry, enriching the depth and quality of the research findings.

2.4. Data analysis

To evaluate the impact of the instructional intervention on participants' P-CTS, TBaCCT, and TPACK-Math scores, this study followed a structured process of parametric and non-parametric statistical analyses selected based on the distributional characteristics of the data. As a first step, the Shapiro–Wilk test—commonly used for its sensitivity in detecting deviations from normality in small to medium-sized samples (Razali & Wah, 2011) was conducted on all paired pre- and post-test datasets at the subscale level across the three instruments. In accordance with conventional practice, the threshold for statistical significance was set at $p < 0.05$, with values below this level indicating a violation of the normality assumption (Ghasemi & Zahediasl, 2012; Field, 2018).

The Wilcoxon Signed-Rank Test was used for all subscales of the P-CTS, as each paired dataset included at least one variable that violated the assumption of normality, even though Ev-pre, Ck-pre, and Al-post individually followed a normal distribution. For the TBaCCT, only the Va-pre and Va-post datasets met the normality assumption and were analyzed using the paired samples t-test. In contrast, the Se and Te subscales, which did not meet the normality assumption, were analyzed using the Wilcoxon Signed-Rank Test. In the TPACK-Math, the paired datasets T-pre & T-post, P-pre & P-post, and Tp-pre & Tp-post met the normality assumption and were assessed using the paired samples t-test. The remaining subscales (C, Tc, Pc, Tpc, and Cx), each involving at least one non-normally distributed variable, were examined using the Wilcoxon Signed-Rank Test.

While the t-test and Wilcoxon Signed-Rank Test provide a strong basis for revealing the effect of the educational intervention by focusing on mean or median changes, diverging column charts were used to show the direction and magnitude of individual score differences, and box plots were used to visualize the overall tendency of the group through median shifts, spread, and outliers. Together, diverging column charts and box plots provided multi-dimensional insight into the data by capturing both individual-level variation and group-level distributional patterns.

A systematic thematic analysis (Nowell et al., 2017; Lochmiller, 2021) began with an in-depth familiarization process in which interview transcripts were extensively reviewed to immerse the researchers in teachers' experiences with CT integration in mathematics education. Next, systematic coding was conducted to identify key concepts, and the coded segments were categorized into three overarching themes: the impact of PD, strategies for successful integration, and challenges in implementing CT in mathematics instruction. These themes were further refined into thirteen sub-themes (Table 2). Direct participant quotations were integrated to illustrate each sub-theme, enhancing the analytical depth and trustworthiness of the findings. This approach enabled a nuanced interpretation of how educators engage with CT, adapt it to their teaching contexts, and identify the systemic and pedagogical conditions required for meaningful integration.

Table 2. *Summary of the themes and sub-themes*

Theme/ Sub-theme:	**Description:**
The first theme: The impact of PD	
Overcoming Initial Fears	Training helped participants move from uncertainty and fear to confidence in CT integration.
Applying CT in Math Instruction	Teachers explored using CT tools like Scratch to support mathematical tasks and problem-solving.
Enhancing Student Engagement	CT-based activities increased student interest and reduced boredom in math lessons.
Creating Personalized CT Resources	Participants transitioned from using pre-made coding materials to designing customized resources for their classes.
The second theme: The strategies to be followed for successful integration	
Early Exposure to CT for Long-Term Benefits	Introducing CT in early ages to build foundational skills, enhance engagement, and empower students for future learning and applications.
Progressive Integration of CT	Emphasizing a structured progression from unplugged activities to

in Mathematics Instruction	block-based and text-based coding to facilitate deeper learning.
Overcoming Language Barriers in Coding	Overcoming language barriers and coding challenges through practice and support from computer teachers to foster independence.
Building Teacher Competency in CT and Coding	Equipping math teachers with CT and coding skills via university and in-service training to enhance competence and reduce reliance on traditional methods.
The third theme: The potential challenges in implementing CT in mathematics instruction	
Curricular Limitations	The current mathematics curriculum lacks the necessary structure and time allocation to integrate CT-based activities effectively.
Insufficient Digital Learning Resources	Existing coding-based math resources are limited, failing to cover all learning outcomes and support diverse student needs.
Teacher Resistance to Change	Some educators are reluctant to adopt new teaching approaches, relying on traditional methods despite curriculum updates.
Infrastructure and Technological Barriers	Lack of access to computers, internet, and digital tools in certain regions prevents effective CT integration.
Challenges of Large Class Sizes	Managing and monitoring students' coding progress is difficult in overcrowded classrooms, limiting individualized instruction.

3. Findings

This section presents the integration of quantitative data derived from three instruments and qualitative insights obtained through thematic analysis of semi-structured interviews, offering a comprehensive account of participants' learning outcomes and pedagogical perspectives.

3.1. Quantitative Findings

Based on the outcomes of the Shapiro–Wilk normality tests, either paired-samples t-tests or Wilcoxon Signed-Rank Tests were employed to assess statistically significant changes in participants' self-perceived computational thinking skills, beliefs about coding and CT, and their technological pedagogical content knowledge in mathematics. To complement these inferential analyses, diverging column charts and box plots were used to visualize both individual- and group-level score variations, providing a more nuanced depiction of shifts across the subdomains of each scale.

3.1.1. Pre- and post-training comparison using the P-CTS

The Shapiro–Wilk normality assessment for the P-CTS indicated that Ev-pre, Ck-pre, and Al-post were normally distributed ($p > 0.05$), whereas the remaining distributions deviated from normality ($p \leq 0.05$). Therefore, Wilcoxon signed-rank tests were conducted to compare pre- and post-test scores across the P-CTS subscales. The analyses revealed significant improvements in Ev ($W = 1.5$, $p = 9.88e{-}06$), Ck ($W = 6.0$, $p = 1.08e{-}05$), and Al ($W = 2.0$, $p = 6.95e{-}06$), indicating that the training positively affected participants' programming-oriented CT skills.

Table 3. Shapiro-Wilk Statistics for *P-CTS*

Sections	Ev-pre	Ev-post	Ck-pre	Ck-post	Al-pre	Al-post
W	0.948	0.926	0.968	0.912	0.904	0.934
p	0.176	0.049	0.519	0.022	0.014	0.076

Table 4. Wilcoxon Signed Rank Test Results for P-CTS

Sections	Ev-pre & Ev-post	Ck-pre & Ck-post	Al-pre & Al-post
W	1.5	6.0	2.0
p	9.88e-06	1.08e-05	6.95e-06

A diverging column chart was created to provide a detailed analysis of participants' CT sub-components, revealing a generally positive change across all three sub-components—evaluation (25 participants), conceptual knowledge (24 participants), and algorithmic thinking (26 participants). While algorithmic thinking showed the greatest number of positive individual changes, a small number of participants in evaluation and conceptual knowledge exhibited slight negative changes. When changes were further examined using a predefined threshold (i.e., focusing on more substantial gains), the largest improvements were observed most frequently in conceptual knowledge (21 participants), followed by evaluation (19 participants) and algorithmic thinking (17 participants).

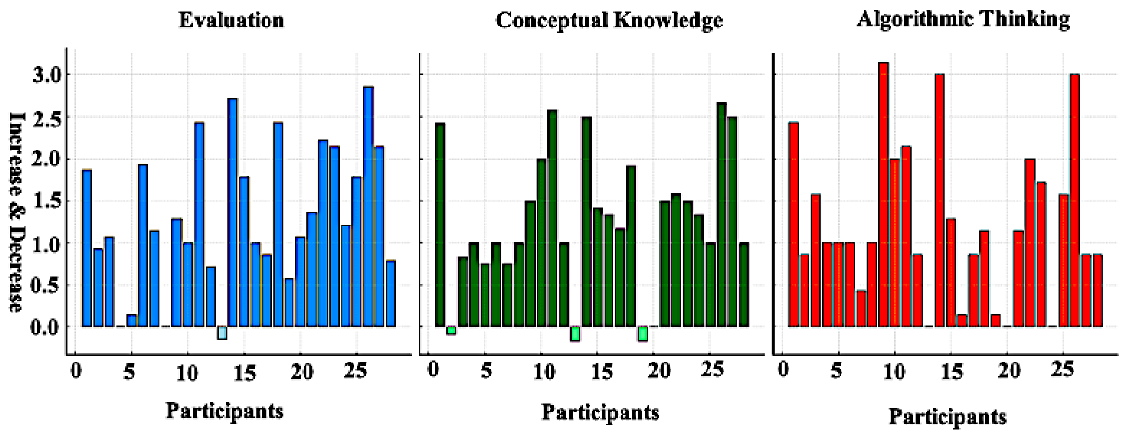

Figure 1. *Diverging column chart for P-CTS*

The box plots (Figure 2) show a clear upward shift from pre- to post-test across all three P-CTS subscales (Ev, Ck, and Al), as evidenced by higher post-test median values, which corroborates the overall positive direction observed in the diverging column chart. Pre-test distributions were comparatively more heterogeneous—most notably in Ev and Al—indicating greater variability in teachers' initial self-perceptions of evaluation and algorithmic thinking skills. Following the training, the post-test box plots appear more compressed (i.e., with narrower interquartile ranges and generally shorter whiskers), suggesting more consistent ratings and a reduced disparity among participants. In addition, the relatively more symmetric appearance of the Ck-pre distribution implies a more balanced clustering around the median prior to the intervention, whereas the upward shift in Ck-post indicates overall improvement in conceptual knowledge. Finally, the outlier visible in the Ck-post plot points to one participant reporting a markedly lower post-test score compared to the rest of the group, despite the general pattern of improvement.

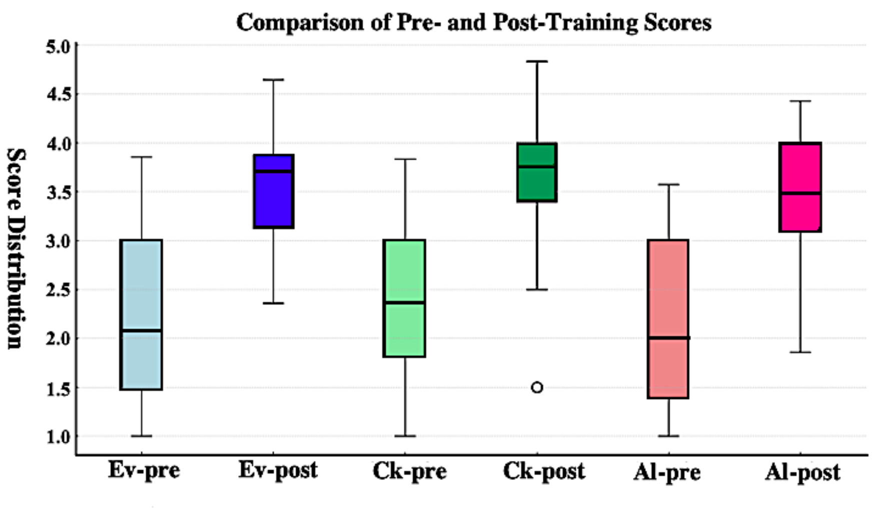

Figure 2. *Box plots for P-CTS*

3.1.2. Pre- and post-training comparison using the TBaCCT scale

The Shapiro–Wilk test indicated that Va-pre and Va-post were normally distributed ($p > 0.05$) and were therefore analyzed using a paired-samples t-test. In contrast, Se-pre and both Te-pre and Te-post deviated from normality ($p < 0.05$) and were analyzed using the Wilcoxon Signed-Rank Test. Although Se-post was normally distributed ($p > 0.05$), the paired dataset for self-efficacy was treated as non-normal due to the non-normal distribution of Se-pre.

Table 5. *Shapiro-Wilk Statistics for TBaCCT scale*

Sections	Va-pre	Va-post	Se-pre	Se-post	Te-pre	Te-post
W	0.9646	0.9359	0.9047	0.9820	0.8258	0.8759
p	0.4448	0.0872	0.0148	0.8950	0.0003	0.0033

The Wilcoxon Signed-Rank Test indicated statistically significant improvements in both self-efficacy ($W = 2.0$, $p \approx 0.00000688$) and teacher efficacy ($W = 1.0$, $p \approx 0.0000000149$), reflecting a strong impact of the training on participants' confidence in integrating coding and CT. In contrast, the paired t-test for value-related beliefs yielded a non-significant result ($p > 0.05$), suggesting that participants' attitudes, expectations, and perceived benefits regarding the role of computing in education remained largely unchanged.

Table 6. *Results of t-test and Wilcoxon Signed Rank Test for TBaCCT scale*

Sections	**Va-pre & Va-post**	**Wilcoxon Test**	**Se-pre & Se-post**	**Te-pre & Te-post**
t-statistic	1.180832840491446	W	2.0	1.0
p	0.2479671019445261	P	6.879521225403697e-06	1.4901161193847656e-08

Examining the Value diverging column chart reveals that 15 participants exhibited a positive change, slightly more than those who showed no change (4) or a negative change (9). While this suggests that the training positively influenced the Values of more than half of the participants regarding the integration of CT into classrooms, it also underscores that a considerable number remained unaffected or were negatively impacted by the intervention. Although only one participant experienced a decline in coding and CT self-efficacy, 96% of participants reported strengthened beliefs in their ability to teach coding and CT in classrooms following the intervention. When comparing the lengths of the bars above the reference line across the three graphs, it can be observed that even among participants for whom the intervention was effective, the increase in Values regarding the integration of CT into classrooms was lower than the increase in their self-efficacy and teacher efficacy for coding and CT integration. Moreover, the higher positive bars in Teacher Efficacy compared to Self-Efficacy indicate that the intervention contributed to the most notable improvement in participants' confidence in integrating CT into their teaching, particularly in lesson planning, explaining CT concepts, incorporating relevant resources, supporting students in debugging, and connecting CT principles to real-world applications.

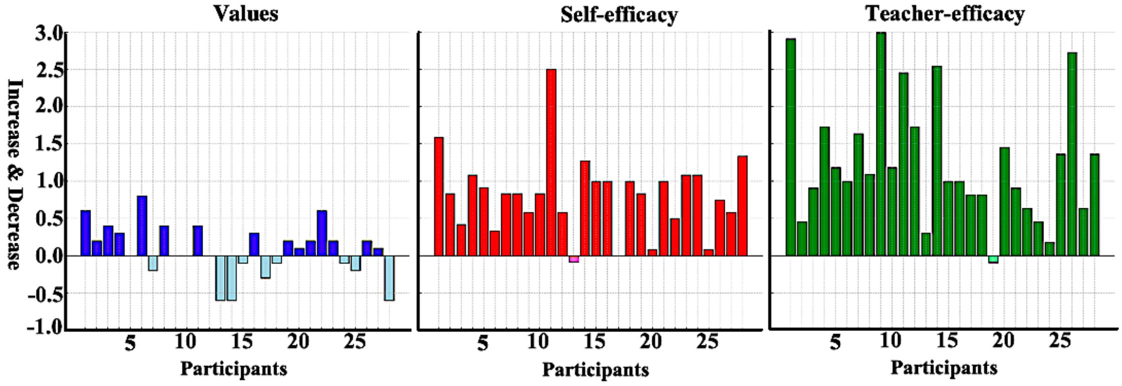

Figure 3. *Diverging column chart for TBaCCT scale*

In Va-pre and Se-pre, the box plots appear more compressed, with median lines positioned at the center of the boxes. In contrast, Va-post and Se-post show slightly wider distributions, where the median line in Va-post is closer to the lower bound, while in Se-post, it is near the upper bound. This indicates that before the training, participants' ratings on the value and self-efficacy of both coding and integrating CT into teaching were relatively uniform and homogeneously distributed. The unchanged median near the lower bound in Va-post suggests that the training had little overall impact on participants' perceived value of CT integration in teaching, while the slightly increased variability—evident from the wider interquartile range—indicates that individual responses differed, with some participants raising their ratings and others maintaining pre-training scores; the presence of an outlier above the upper whisker further suggests that a small number of participants experienced a substantial positive shift in their perception. The increase in median values in Se-post and Te-post compared to their pre-training counterparts indicates that the intervention positively influenced participants' self-efficacy and teacher efficacy beliefs in integrating coding and CT into teaching. The placement of the median line in the upper part of the box in Se-post, Te-pre, and Te-post indicates that participants' ratings were concentrated at higher levels. However, the greater distance between the median and the lower whiskers in Te-pre and Te-post suggests greater variability in the lower part of the data set, meaning that lower scores occupied a considerable portion of the overall distribution and that some individuals assigned notably low ratings. The outliers above the box in Se-pre indicate that certain participants assigned notably high ratings to the value and self-efficacy of integrating coding and CT into teaching. Conversely, the three lower outliers in Se-pre suggest that some participants had low self-efficacy in their coding knowledge and skills. Similarly, the lower outlier in Te-post implies that certain participants experienced a decline in their teacher efficacy, suggesting that the instructional intervention did not meet their expectations for improving their confidence in teaching coding and CT.

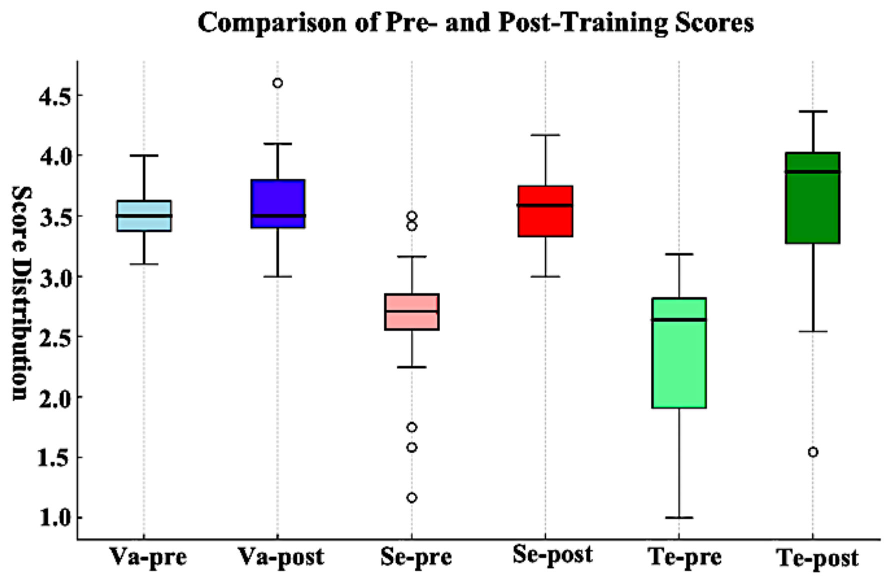

Figure 4. *Box plots for TBaCCT scale*

3.1.3. Pre- and post-training comparison using the TPACK-Math scale

The Shapiro-Wilk test showed that the T-pre, T-post, P-pre, P-post, Tp-pre, Tp-post, Tc-pre, and Tpc-pre datasets followed a normal distribution ($p > 0.05$). In contrast, the C-pre, C-post, Tc-post, Pc-pre, Pc-post, Tpc-post, Cx-pre, and Cx-post datasets did not ($p < 0.05$). As a result, paired t-tests were conducted for dataset pairs T-pre & T-post, P-pre & P-post, and Tp-pre & Tp-post, as they met the normality assumption. The Wilcoxon Signed-Rank Test was applied to the remaining dataset pairs, in which at least one dataset deviated from normality.

Table 7. *Shapiro-Wilk Statistics for TPACK-Math scale*

Sections	W	P		W	P
T-pre	0.978161	0.804304	Tc-pre	0.927667	0.053856
T-post	0.937772	0.097014	Tc-post	0.871220	0.002586
P-pre	0.927938	0.054706	Pc-pre	0.909085	0.018794
P-post	0.935280	0.083855	Pc-post	0.881275	0.004285
C-pre	0.887802	0.005996	Tpc-pre	0.942780	0.130077
C-post	0.799937	0.000106	Tpc-post	0.911184	0.021116
Tp-pre	0.934665	0.080896	Cx-pre	0.923761	0.043003
Tp-post	0.937358	0.094691	Cx-post	0.875098	0.003136

The t-test for the paired comparisons T-pre & T-post, P-pre & P-post, and Tp-pre & Tp-post yielded t-statistics with strong negative values and $p<0.05$, confirming the statistical significance of the instructional intervention's impact; these t-values indicate that the intervention significantly enhanced participants' knowledge in technology, pedagogy, and technological pedagogy, as post-test scores were notably higher than pre-test scores. The Wilcoxon test results indicate statistically significant improvements in technological, pedagogical content, technological pedagogical content, and contextual knowledge components ($p < 0.05$). The lowest W-value (12.0) and strongest significance ($p = 0.000079$) for Tpc-pre & Tpc-post suggest the most substantial improvement in Technological Pedagogical and Content Knowledge. In contrast, the highest W-value (51.5) for Cx-pre & Cx-post, despite being significant ($p = 0.045415$), indicates a relatively smaller shift in participants' ability to consider external classroom factors such as technological infrastructure, environmental conditions, societal context, and students' access to technology when teaching mathematics.

Table 8. *Results of t-test for TPACK-Math scale*

Sections	t	p
T-pre & T-post	-5.888076	0.000003
P-pre & P-post	-3.009564	0.005612
Tp-pre & Tp-post	-5.967302	0.000002

Table 9. *Results of Wilcoxon Signed Rank Test for TPACK-Math scale*

Sections	W	p
C-pre & C-post	38.5	0.012884
Tc-pre & Tc-post	25.5	0.001034
Pc-pre & Pc-post	21.5	0.003080
Tpc-pre & Tpc-post	12.0	0.000079
Cx-pre & Cx-post	51.5	0.045415

The diverging column charts reveal a significant improvement in participants' technology-focused knowledge and skills, particularly in the Technological, Technological-Pedagogical, and Technological-Pedagogical-Content domains. For instance, in the Technological domain, 23 participants reported a positive change, with 14 of them showing a substantial increase of 1 or above. Likewise, the Technological-Pedagogical (23 positive, 11 at 1 or above) and Technological-Pedagogical-Content (21 positive, 13 at 1 or above) domains also showed notable gains. On the other hand, while the Pedagogical (20 positive, only 1 at 1 or above) and Pedagogical-Content (16 positive, 3 at 1 or above) domains displayed an overall positive trend, the proportion of participants making a substantial leap in these areas remained relatively small. In the Content (15 positive, 8 neutral, 5 negative, 1 at or below -1) and Contextual (14 positive, 8 neutral, 6 negative, 2 at or below -1) domains, the proportion of participants reporting neutral or negative shifts is higher compared to other components. In each dimension, there were more participants whose post-test scores exceeded their pre-test scores, indicating that the overall trend is still predominantly positive. Only a small number of participants did experience a marked decline below -1 (e.g., 1 in Content, 1 in Technological Pedagogical), and these instances remain relatively few.

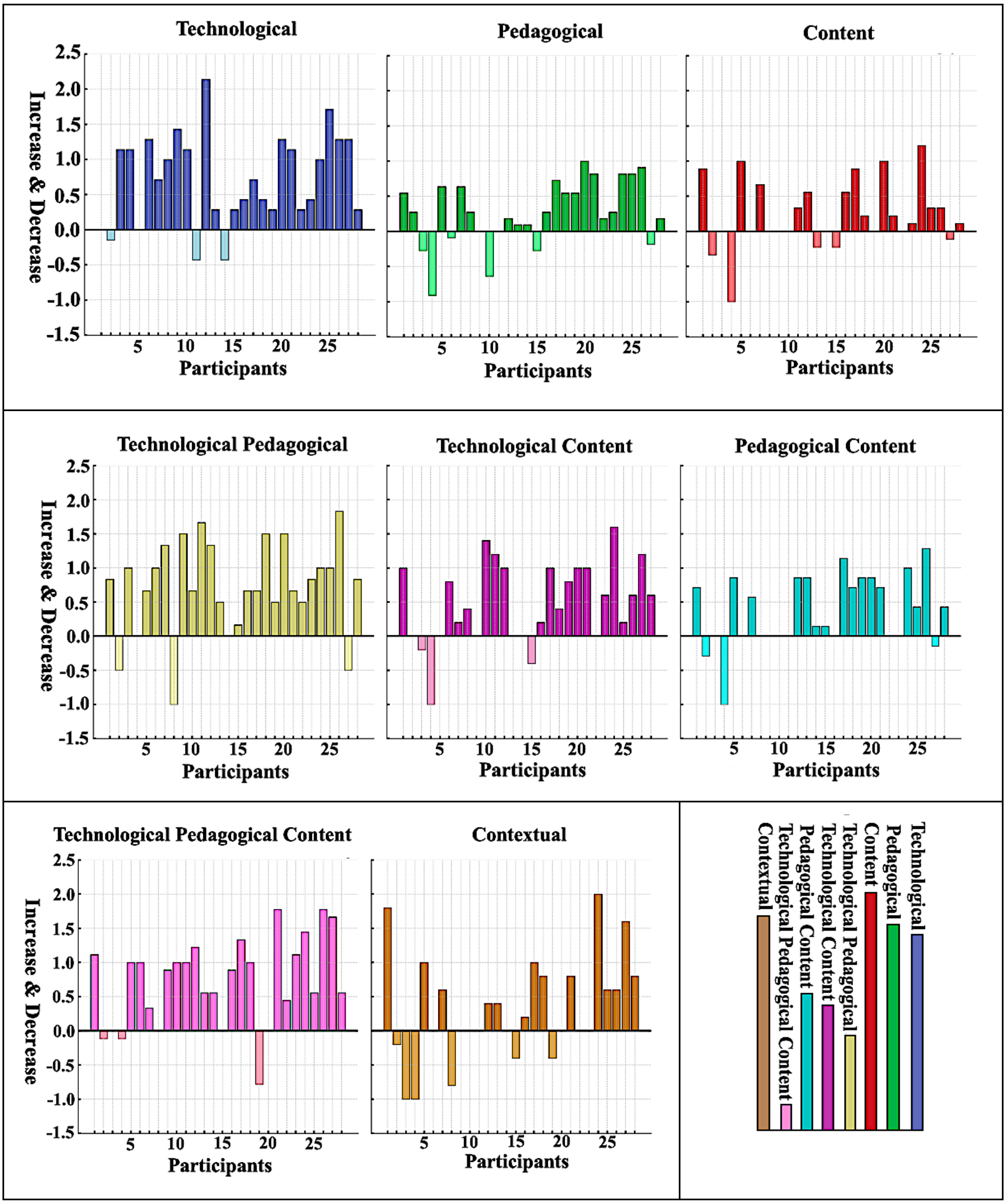

Figure 5. *Diverging column chart for TPACK-Math scale*

An examination of the box plots reveals that, apart from Cx-pre and Cx-post, the post-training median measurements generally exceed the matched pre-training measurements across most components. Although P-post, C-post, and Pc-post show relatively slight increases, the other components exhibit more pronounced improvements overall. The consistent box widths in T-post, Tpc-post, and Cx-post, when compared to their pre-instruction counterparts, suggest that the spread or dispersion of the data in these areas remained relatively unchanged after the intervention. In contrast, P-post and Tc-post boxes have slightly narrowed, suggesting more consistent ratings among participants, whereas C-post, Tp-post, and Pc-post display an expanded range, indicating a wider dispersion of scores in these domains. Regarding the distance between the upper and lower whiskers, T-pre, T-post, Tp-post, Tc-post, Tpc-post, Tpc-pre, Cx-pre, and Tp-pre exhibit notably longer whisker spans, suggesting that the highest and lowest measurements in these domains are more widely distributed. Additionally, Pc-post and Cx-post have only a short gap between the top of the box and the upper whisker, implying that the highest values in these domains cluster near the upper boundary of the box. The outliers in T-pre, Tc-post, Pc-pre, Tpc-pre, and Tpc-post—some appearing above the upper whisker, some below the lower whisker, and in certain cases both—indicate that, in these domains, a small subset of participants' ratings deviate significantly from the main distribution. A notable observation is that, across P-pre, P-post, C-pre, Tp-post, Tc-pre, Tc-post, Pc-post, Tpc-post, Cx-pre, and Cx-post, the upper whiskers—and in the case of C-post, the top edge of the box—extend to the maximum rating of 5, indicating that one or more participants in each domain felt fully competent and therefore assigned the highest possible scores.

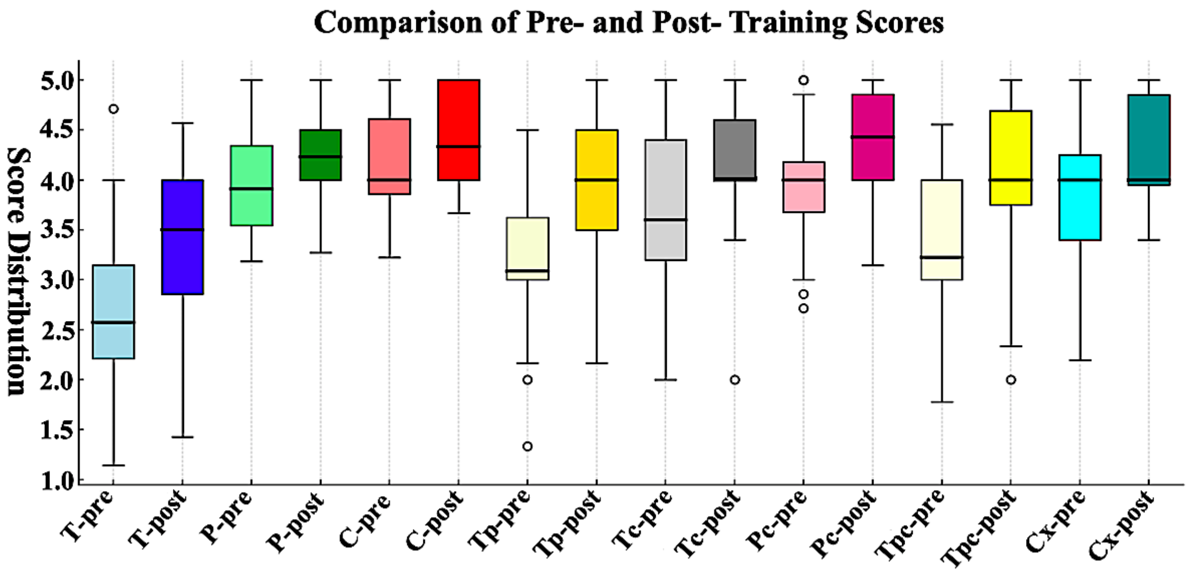

Figure 6. *Box plots for TPACK-Math scale*

3.2. Qualitative Findings

Thematic analysis, grounded in excerpts from teacher interviews, offered insights into their evolving perspectives, proposed instructional strategies, and anticipated challenges related to the integration of CT into mathematics education.

3.2.1. Impact of PD on Participants' Perceived Competence in Integrating CT into Mathematics Education

Participant P1's thinking process evolved from initial uncertainty and fear regarding CT integration in mathematics instruction to a state of confidence and proactive idea-generation. The structured and step-by-step nature of the training helped alleviate these fears, providing hands-on opportunities to build confidence. By the end, the participant not only felt capable of implementing CT but also began formulating new ideas on how it could be used effectively in various contexts.

P1: Before the training, yes, I could surmise from certain assumptions that there were useful activities and that we could use them in mathematics teaching… but, as for myself, I must say I had a fear like, "I might not be able to do it, I might not be able to explain it, I might not be able to learn it…" But, I mean, the training was so well structured. We first covered what CT is, then unplugged coding, block-based coding… all step by step, and it was really well organized, with each session contributing to both the previous and the next… And now, yes, I feel like I can do it, it might work, it might be usable… and I'm beginning to come up with ideas on how it could be used in certain contexts.

Participant P2 recognized that CT could be integrated into mathematics but felt unsure about how or where to begin for her own teaching. After seeing what tools like Scratch could do, she realized she could create function graphs (including logarithmic ones), as well as animations and games to enhance math lessons. This shift—from not knowing her starting point to beginning to envision programs tailored to various math-related tasks—demonstrates newly gained confidence in applying CT.

P2: I think it can be integrated, but to be honest, I didn't know where to start or what needed to be done about it. But now, yes, for example, I can do a lot with the Scratch program. If I really focus on it now, I believe I can integrate it nicely—at least that's what I'm thinking... Like, regarding graphics, for instance, function graphs, logarithmic graphs, maybe I could create some programs related to that... Apart from that, for example, we can develop animations... we can easily develop games.

Participant P3 underscores the pedagogical advantages of integrating CT into mathematics, especially to combat student boredom and enhance engagement. Having overcome the initial fear that it might be "impossible," P3 now feels confident about applying these new methods yet recognizes the need for additional inquiry to refine and expand their approach. By broadening their exploration and reaching higher levels of proficiency, P3 envisions offering more innovative and motivating learning experiences in math.

P3: Yes, from a pedagogical standpoint, I felt it was doable, and I think it can be implemented right now. In terms of getting the students' attention, our biggest problem is that they often get bored, so it's already difficult to engage them in the lesson, and even those who are interested eventually get bored. When they're confronted with different kinds of activities like this, they pay much more attention… As for teaching, I've now broken that 'it can't be done' mindset… I got over that fear, but I still need to do more research; after all, we're still at the beginning, but it can be looked into, and by researching further, we can move to higher levels and convey it to the students.

Participant P9 mentions previously relying on ready-made, coding-based digital resources from other teachers, which they had to use 'as is.' Consequently, these materials often didn't align with the exact level or questions needed for their own classes. After the training, P9 realized they could create such materials themselves, tailoring them more precisely to class needs. While acknowledging it requires effort, P9 now sees this as a feasible and effective way to integrate CT into math instruction, adapting activities to students' levels.

P9: I used to find these ready-made versions from our teachers and get support that way, using those. Now I've realized I can actually make them myself or adapt them, and for example, when I found prepared ones, sometimes they didn't have exactly the questions I wanted. Regarding the class level, it would be either too simple or more advanced… I couldn't strike that exact balance, nor could I find them for every topic. After this training, I've seen that I can do them on my own. It just takes a bit of effort.

3.2.2. Perspectives on effective ways to integrate CT into mathematics education

P10 underscores the value of introducing coding and CT instruction from an early age, observing that children already use technology before they even learn to read. P11 adds that by high school, such early exposure could enable students to develop their own programs and assist teachers in class, benefiting both student empowerment and teacher workload. Despite personal uncertainties about practical application, P10 believes that starting in preschool fosters a strong knowledge base, leading to more engaging, sustainable learning experiences and shaping students' future paths.

P10: It really needs to start at a younger age. Today's children are introduced to technology at a very early stage—they can easily use a tablet, phone, or the internet by age two or three, even before they learn to read. It's not that learning becomes harder as we get older, but since we never encountered anything like this before, it felt difficult to me at first. However, for a younger child, I'm sure it would be more fun, more attention-grabbing. If it begins in early childhood—preschool—then the child could have this knowledge base and even shape their future around it.

P11: If we start providing coding education at a young age, then by high school we might have a student who can design their own software or even assist me—a math teacher—with the lesson. For instance, if we assign a project at school and there's a student who truly knows programming, they *could design the topic we give them, then come and present it, actually using it in the classroom—so they'd both create it and implement it.*

Participant P6 outlines a progression for integrating coding into mathematics instruction: begin with unplugged activities (no computers), then proceed to block-based approaches, and ultimately advance to text-based languages. This step-by-step model is seen as a gradual way to build both student and teacher competence in coding for math. Participant P9 notes that, at the upper elementary or middle school level, block-based coding aligns with students' developmental stages. As a next step, P9 points out that some environments allow viewing both block-based and textual versions side by side—a form of block-text coding. As learners progress, P9 proposes transitioning to more complex languages like Python in higher grades, thereby deepening CT skills.

P6: We should start with unplugged coding and proceed step by step: first unplugged coding, then computer-based block-based coding, then block-based text coding.

P9: As they get older—for instance, block-based learning can be for middle school or maybe late elementary, like 3rd-4th grade, depending on their level, their developmental stage... As a next stage, for example, 'Hello World' pops up on its own. What was that? EduBlock, right? And you can see the coding and the text. You can see both the blocks and their code texts. Then we can move on to other coding types as well, like Python or similar, for upper levels, high school, I guess. I think it would be suitable there.

Participant P8, while confident in algebraic expressions, finds coding challenging due to language barriers in programming interfaces and unfamiliar "computer language." They highlight the need for computer teacher support in overcoming these obstacles. Despite initial struggles, P8 acknowledges progress through hands-on practice and anticipates greater independence in debugging and coding with continued effort.

P8: So, for example, if we're going to write a formula or cover something about a topic—like coefficient summation in algebraic expressions—that part is fine for me, but when it comes to coding, I might need help if I get stuck. Also, these programs use a lot of English, plus there's this 'computer language,' so because of some language barriers, it can get complicated. That's why a bit of support from a computer teacher would be reassuring, but I believe that as I keep using it and really settle those concepts, I'll be able to write code comfortably or spot my own mistakes. I still find issues sometimes—things not working or working incorrectly—so I still ask questions and there are things I can't figure out. Getting help in that sense is a relief.

Participant P1 highlights that comprehensive teacher training—whether through university-level courses or in-service programs—can prepare teachers with the knowledge and competencies to effectively integrate CT into mathematics lessons. P1 argues that many mathematics teachers default to traditional methods simply because they have not equipped themselves with the competencies required by digitalization, suggesting that contemporary training approaches should adopt coding and CT skills through which even non-specialist teachers can integrate CT into mathematics.

P1: In-service trainings can be conducted... In-service trainings, distance learning, interactions with different stakeholders... I thought that a mathematics teacher who lacks sufficient knowledge could be trained in this way. But of course, adequate training on this subject must be provided at the university level... Teachers resort to the traditional methods because they are not very competent in this area... That's why there should be serious training at the university level... Of course, a teacher might not have as much knowledge as a computer engineer or a computer teacher, but at least they can possess enough computational skills to use in mathematics

3.2.3. Participants' perspectives on barriers to CT integration in mathematics education

Participant P4 finds the current mathematics curriculum "inadequate" or "lacking," stressing that teachers cannot effectively incorporate new digital-oriented approaches or varied assessment methods without additional structure and time. P4 advocates expanding the curriculum—both in content and class hours—to better accommodate innovative teaching strategies, including activities requiring digital environments and CT.

P4: Our curriculum is very inadequate… In order for me to conduct different assessment and evaluation activities within my teaching process, I need to be able to do these kinds of coding applications... This also means I need the means to carry out these activities… Actually, let me not say 'inadequate'—I think it's lacking at this point... Because when we say 'inadequate,' it sounds like it exists but isn't enough… yet I don't really see these things in our curriculum

Participant P2 finds current code-based math resources "not very sufficient," pointing out that EBA (Education Informatics Network), a national educational platform, may offer an activity for "LCM" or rational number operations, yet other topics remain uncovered. As a result, a student who's already mastered rational numbers—or who struggles with a different concept—cannot easily find the coding support they need. P2 suggests that "every learning outcome" should have accessible, reliable, and enriched content so students can quickly locate the exact help they need without resorting to scattered or incomplete materials.

P2: It's not very sufficient… Right now, there's a coding activity on EBA related to LCM… Maybe it covers a single learning outcome… But if a student is struggling with another learning outcome and wants to quickly check it, like, 'How does this work?' that doesn't exist… There's only one activity, for example, about operations with rational numbers… Maybe the child has already mastered that, but is having trouble with something else in rational numbers, or another topic… Actually, for every learning outcome, there could be easily accessible, reliable, and enriched content. Yes, we see there's some, but it's not enough. I think every learning outcome should have an activity.

Participant P2 highlights a crucial challenge in educational reform: teacher resistance to change. P2 argues that simply updating the curriculum is insufficient for genuine digital transformation. Instead, the focus should be on breaking entrenched teaching habits, particularly amongst veteran educators who may resist deviating from established practices, regardless of curricular changes.

P2: I don't actually think our teaching programs or curricula are that bad. But we're not implementing them, or we're reluctant to do so. Or maybe it's just hard to change ourselves. For example, a teacher who's been at it for 20 years explains numbers—natural numbers—the same way they did 20 years ago. They explain fractions the same way, too. So even if you change the teaching program, there's no real effect, because the teacher just doesn't use it.

Participants P5 and P6 highlight infrastructure challenges that hinder the implementation of coding or CT-based activities in mathematics. One teacher (P6) underscores the need for a computer lab, while another (P5) points out wider technological gaps in certain regions, where students may lack basic internet access or devices.

P5: I mean, first of all, students really need proper technological equipment. Maybe in the western regions we have fewer problems with that, but in the east there are a lot of difficulties… In terms of technology, not being able to access the internet… even today, there are towns and villages that still struggle with this.

P6: If we want to carry out this process with the students… we need a computer environment…. If such an environment is created, I think really great things could come out of working together in a setting that has a computer lab…. Let's say our school doesn't have a lab environment… But after I took this training…I started thinking of things like, 'You could help me with this, or we could do a project'.

Participant P7 emphasizes that in large classrooms, it becomes difficult to oversee everyone's coding progress or provide personalized guidance. Smaller groups, by contrast, allow the teacher to promptly detect and address errors within a single lesson. However, when the class size grows, time constraints and the demand for individual attention pose a major hurdle to ensuring each student gets the help they need.

P7: I think it could work, but I believe it would be feasible with smaller class sizes. In larger classes, it becomes more difficult to keep track, and helping those who can't do it is a bit more challenging. I mean, following the steps and all that, especially when we say fewer students... Exactly, so here's the thing: our class hours are already somewhat limited, so if the class is small, it's easier for me to check *everyone in one lesson period and spot their mistakes. But in a crowded class, of course, I think it would be much harder to do.*

4. Discussion and Conclusion

Structured, practice-based professional development that integrates CT into mathematics education has the potential to foster substantial growth in teachers' CT and TPACK competencies, as well as in their technical and pedagogical beliefs about coding and CT. Consistent with prior research (Kong et al., 2020; Leonard et al., 2017; Yadav et al., 2016), the present findings, interpreted in relation to the study's RQs, suggest that well-designed training can both reshape teachers' perspectives and make visible the structural conditions that enable or constrain implementation. In this respect, meaningful CT integration cannot be achieved solely by revising lesson content or aligning instructional practices with CT principles; it also depends on coordinated support systems that promote curricular coherence, equitable access to technological resources (e.g., digitally equipped classrooms and manageable class sizes), and sustained capacity development across in-service and pre-service teacher education, including mathematics and primary education.

To address RQ1, the P-CTS outcomes reveal notable growth in participants' CT-related abilities, particularly in conceptual knowledge and algorithmic thinking. While the most pronounced improvement occurred in conceptual understanding, development in algorithmic thinking was also significant, despite some variability among participants. However, the evaluation component—which involves debugging and programming flexibility—lagged behind, highlighting an area of relative weakness. The enhancement in conceptual knowledge observed in this study mirrors findings by Molina-Ayuso et al. (2022) and Kong et al. (2020), who emphasized the pedagogical value of block-based CT platforms such as Scratch. The second key outcome—improvement in algorithmic thinking—corresponds with Tripon's (2022) conclusion that CT instruction significantly supports the development of algorithmic reasoning. In contrast, the modest gains in assessment skills, particularly in debugging and adaptive programming, signal the need for more comprehensive and targeted training in these domains.

Findings from the TBaCCT scale indicate a significant increase in participants' self-efficacy and teacher efficacy beliefs; however, their perceptions of the value of CT in classroom settings did not exhibit a comparable change. These results are in line with earlier studies by Leonard et al. (2017) and McGinnis et al. (2022). Leonard et al. showed that interactive, programming-rich environments—such as game design and robotics—can positively shape teachers' beliefs about CT. In a similar vein, Kong et al. (2020) argued that teacher education programs integrating programming knowledge, pedagogical frameworks, and classroom-based practice—supplemented with opportunities for reflection—can effectively enhance teachers' pedagogical efficacy in CT integration. Nevertheless, in their study with pre-service teachers, McGinnis et al. (2022) reported that although participants regarded CT as valuable in elementary education, they experienced conceptual confusion about its nature, which led to a superficial formation of value-related beliefs. This indicates that the limited development in value-related beliefs is not merely a result of insufficient attitudinal support, but reflects a lack of conceptual clarity regarding CT's teachability at early grade levels, its capacity to foster student engagement and enrich general learning, its relevance to future career readiness and 21st-century literacy, as well as its applicability across different educational levels, which relates directly to RQ2.

Results from the TPACK-Math scale indicate that participating teachers demonstrated substantial growth in technology-related domains—specifically in technological knowledge, technological pedagogical knowledge, and technological pedagogical content knowledge. These areas of growth—reflecting the knowledge components examined in RQ3—closely align with the three core TPACK competencies identified by Kong et al. (2020), who reported similar gains following CT-based training in (i) content knowledge related to CT, (ii) technological knowledge associated with block-based programming environments, and (iii) the pedagogical integration of these tools into classroom practice. However, our findings also revealed limited progress in the contextual knowledge domain, suggesting that teachers require more comprehensive support in addressing external factors—such as school infrastructure, student access to technology, and socioeconomic context—when designing and delivering mathematics instruction.

The training process revealed that teachers' initial uncertainty and anxiety regarding the integration of CT into mathematics instruction gradually transformed into self-confidence and a strong willingness to implement CT practices in their classrooms. Participants developed a strong belief that integrating CT tools—such as Scratch—into mathematics lessons could enhance students' interest and motivation, consistent with the perceived pedagogical competence examined in RQ4. Moreover, rather than relying solely on pre-made programs, teachers began to generate ideas and instructional scenarios for how CT could be meaningfully integrated into mathematics teaching at various student levels, even though they had not yet implemented these in real classroom settings. This progression from uncertainty to confident and reflective thinking aligns with the experiences of teacher educators reported in Rajapakse Mohottige et al. (2024), further affirming the potential of CT integration to reshape mathematics education. This shift suggests that teachers were not merely acquiring technical skills, but also becoming willing to meaningfully incorporate CT principles into their instructional thinking and pedagogical perspectives.

Although countries such as the United States and the United Kingdom have introduced CT into their national curricula, Voogt et al. (2015) emphasize that effective implementation requires more than policy—it depends on preparing teachers across disciplines to meaningfully integrate CT into their own subject areas. In line with this emphasis, RQ5 yielded several practical strategies identified by participants for embedding CT into mathematics instruction following the training: early introduction of CT to foster sustained engagement, gradual integration from unplugged activities to block- and text-based coding, addressing language-related coding challenges, and promoting ongoing collaboration with computer science educators. Taken together, these findings present long-term and collaborative strategies for meaningfully integrating CT into mathematics education, in alignment with curriculum goals and grounded in a practice-oriented approach.

Despite the gains achieved through training, participating teachers also identified several potential barriers to the integration of CT into mathematics classrooms—the focus of RQ6. These included curricular constraints, with many noting that the existing program lacks the flexibility and allocated time necessary for meaningful CT integration; the limited availability of coding-based resources that align with learning objectives and support differentiated instruction; and resistance to adopting new instructional approaches, particularly among colleagues who continue to rely on traditional methods. In addition, infrastructural disparities—such as insufficient access to computers, internet, and digital tools—were viewed as significant obstacles, especially in under-resourced schools. Finally, large class sizes were seen to further complicate the monitoring of students' coding progress and limit opportunities for individualized support. These challenges are consistent with prior research, which has also highlighted curricular constraints (Grover & Pea, 2013), infrastructural inequalities (Bocconi et al., 2016; Dagiene et al., 2024), teacher resistance to pedagogical change (Yadav et al., 2016), and the scarcity of aligned instructional resources (Voogt et al., 2015) as key obstacles to the effective integration of CT in educational contexts.

4.1. Limitations

This study, while offering valuable insights into the integration of CT into mathematics education, is limited by its relatively small sample size, due to the logistical challenge of the participation of in-service teachers in an intensive, face-to-face professional development program. The pedagogical design—focused on real-time, hands-on coding and interactive feedback—also constrained scalability and precluded the inclusion of a control group, limiting comparative evaluation. Nonetheless, to ensure internal validity and interpretive robustness, the study adopted a mixed-methods design with purposeful sampling, and relied on standardized instruments with demonstrated high validity and reliability, for which formal usage permissions were obtained. In addition to inferential tests (Wilcoxon, t-test), visual analytics such as diverging column charts and box plots were used to capture individual and group-level score shifts, reinforcing the methodological triangulation and enhancing the trustworthiness of the findings.

References

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing computational thinking in compulsory education: Implications for policy and practice* (EUR 28295 EN). Publications Office of the European Union. https://doi.org/10.2791/792158

Bower, M., Wood, L. N., Lai, J. W. M., Howe, C., & Lister, R. (2017). Improving the computational thinking pedagogical capabilities of school teachers. *Australian Journal of Teacher Education, 42*(3), 53–72. https://doi.org/10.14221/ajte.2017v42n3.4

Carter, N., Bryant-Lukosius, D., DiCenso, A., Blythe, J., & Neville, A. J. (2014). The use of triangulation in qualitative research. *Oncology Nursing Forum, 41*(5), 545–547. https://doi.org/10.1188/14.ONF.545-547

Dagienė, V., Kampylis, P., Giannoutsou, N., Engelhardt, K., Malagoli, C., & Bocconi, S. (2024). Fostering computational thinking in compulsory education in Europe: A multiple case study. *Baltic Journal of Modern Computing, 12*(2), 189–221. https://doi.org/10.22364/bjmc.2024.12.2.05

DeCoito, I. (2024). STEM education: Curriculum and pedagogy. In I. DeCoito, X. Fazio, & J. Gichuru, *Global perspectives on STEM education: Theory and practice* (pp. 51–72). Springer. https://doi.org/10.1007/978-3-031-60676-2_4

Doorenbos, A. Z. (2014). Mixed methods research: Integration of quantitative and qualitative data. In A. Z. Doorenbos (Ed.), *Nursing research using mixed methods: Qualitative designs and methods in nursing* (pp. 3–22). Springer Publishing Company.

Field, A. (2018). *Discovering statistics using IBM SPSS statistics* (5th ed.). SAGE Publications.

Ghasemi, A., & Zahediasl, S. (2012). Normality tests for statistical analysis: A guide for non-statisticians. *International Journal of Endocrinology and Metabolism, 10*(2), 486–489. https://doi.org/10.5812/ijem.3505

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

Jocius, R., Albert, J., Andrews, A., Joshi, D., & Robinson, R. (2020). Infusing computational thinking into disciplinary teaching: From professional development to classroom practice. In D. Schmidt-Crawford (Ed.), *Proceedings of the Society for Information Technology & Teacher Education International Conference* (pp. 1771–1775). Association for the Advancement of Computing in Education (AACE). Retrieved December 27, 2025, from https://www.learntechlib.org/p/215952

Kılıç, S., Gökoğlu, S., & Öztürk, M. (2021). *A valid and reliable scale for developing programming-oriented computational thinking*. *Journal of Educational Computing Research, 59*(2), 257–286. https://doi.org/10.1177/0735633120964402

Kite, V., & Park, S. (2022). Preparing inservice science teachers to bring unplugged computational thinking to their students. *Teaching and Teacher Education, 120*, 103904. https://doi.org/10.1016/j.tate.2022.103904

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*, Article 103872. https://doi.org/10.1016/j.compedu.2020.103872

Leonard, J., Barnes-Johnson, J., Mitchell, M., Unertl, A., Stubbe, C. R., & Ingraham, L. (2017). Developing teachers' computational thinking beliefs and engineering practices through game design and robotics. In E. Galindo & J. Newton (Eds.), *Proceedings of the 39th Annual Meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education* (pp. 1289–1296). Hoosier Association of Mathematics Teacher Educators.

Lochmiller, C. R. (2021). Conducting thematic analysis with qualitative data. *The Qualitative Report, 26*(6), 2029–2044. https://doi.org/10.46743/2160-3715/2021.5008

McGinnis, J. R., Hestness, E., Mills, K., Ketelhut, D. J., Cabrera, L., & Jeong, H. (2022). Preservice science teachers' beliefs about computational thinking following a curricular module within an elementary science methods course. *Contemporary Issues in Technology and Teacher Education, 22*(1), 63–95. Retrieved from https://citejournal.org/volume-22/issue-1-22/science/preservice-science-teachers-beliefs-about-computational-thinking/

Molina-Ayuso, Á., Adamuz-Povedano, N., Bracho-López, R., & Torralbo-Rodríguez, M. (2022). Introduction to computational thinking with Scratch for teacher training for Spanish primary school teachers in mathematics. *Education Sciences, 12*(12), Article 899. https://doi.org/10.3390/educsci12120899

Ng, O.-L., Leung, A., & Ye, H. (2023). Exploring computational thinking as a boundary object between mathematics and computer programming for STEM teaching and learning. *ZDM – Mathematics Education, 55*, 1315–1329. https://doi.org/10.1007/s11858-023-01509-z

Nowell, L. S., Norris, J. M., White, D. E., & Moules, N. J. (2017). Thematic analysis: Striving to meet the trustworthiness criteria. *International Journal of Qualitative Methods, 16*(1), 1–13. https://doi.org/10.1177/1609406917733847

Önal, N. (2016). Development, validity and reliability of TPACK scale with pre-service mathematics teachers. *International Online Journal of Educational Sciences, 8*(2), 93–107. https://doi.org/10.15345/iojes.2016.02.009

Palinkas, L. A., Horwitz, S. M., Green, C. A., Wisdom, J. P., Duan, N., & Hoagwood, K. (2015). Purposeful sampling for qualitative data collection and analysis in mixed method implementation research. *Administration and Policy in Mental Health and Mental Health Services Research, 42*(5), 533–544. https://doi.org/10.1007/s10488-013-0528-y

Palop, B., Díaz, I., Rodríguez-Muñiz, L. J., & Santaengracia, J. J. (2025). Redefining computational thinking: A holistic framework and its implications for K–12 education. *Education and Information Technologies*. Advance online publication. https://doi.org/10.1007/s10639-024-13297-4

Patton, M. Q. (2002). *Qualitative research & evaluation methods* (3rd ed.). Sage Publications.

Rajapakse Mohottige, N. U. S., Bjerke, A. H., & Andersen, R. (2024). Teacher education as stakeholder: Teacher educator perspectives on the integration of computational thinking into mathematics and science courses. *Teachers and Teaching, 30*(4), 437–451. https://doi.org/10.1080/13540602.2024.2313635

Razali, N. M., & Wah, Y. B. (2011). Power comparisons of Shapiro–Wilk, Kolmogorov–Smirnov, Lilliefors and Anderson–Darling tests. *Journal of Statistical Modeling and Analytics, 2*(1), 21–33. Retrieved from https://www.nrc.gov/docs/ml1714/ml17143a100.pdf

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. https://doi.org/10.1080/15391523.2020.1771232

Robinson, O. C. (2014). *Sampling in interview-based qualitative research: A theoretical and practical guide*. *Qualitative Research in Psychology, 11*(1), 25–41. https://doi.org/10.1080/14780887.2013.801543

Sands, P., Yadav, A., & Good, J. (2018). *Computational thinking in K–12: In-service teacher perceptions of computational thinking*. In M. S. Khine (Ed.), *Computational thinking in the STEM disciplines*. Springer. https://doi.org/10.1007/978-3-319-93566-9_8

Saxena, A., & Chiu, M. M. (2022). Developing preschool teachers' computational thinking knowledge, attitudes, beliefs, and teaching self-efficacies: A curriculum-based professional development program. *Frontiers in Education, 7*, Article 889116. https://doi.org/10.3389/feduc.2022.889116

Schoonenboom, J., & Johnson, R. B. (2017). How to construct a mixed methods research design. *Kölner Zeitschrift für Soziologie und Sozialpsychologie, 69*(Suppl 2), 107–131. https://doi.org/10.1007/s11577-017-0454-1

Suters, L., & Suters, H. (2020). Coding for the core: Computational thinking and middle grades mathematics. *Contemporary Issues in Technology and Teacher Education, 20*(3), 435–463. Retrieved from https://citejournal.org/volume-20/issue-3-20/mathematics/coding-for-the-core-computational-thinking-and-middle-grades-mathematics/

Tripon, C. (2022). Supporting future teachers to promote computational thinking skills in teaching STEM—A case study. *Sustainability, 14*(19), Article 12663. https://doi.org/10.3390/su141912663

Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies, 20*(4), 715–728. https://doi.org/10.1007/s10639-015-9412-6

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. (2011). Research notebook: Computational thinking—What and why? *The Link Magazine, Spring 2011*, 20–23. https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why

Yadav, A., Gretter, S., Hambrusch, S., & Sands, P. (2016). Expanding computer science education in schools: Understanding teacher experiences and challenges. *Computer Science Education, 26*(4), 235–254. https://doi.org/10.1080/08993408.2016.1257418

Yadav, A., Stephenson, C., & Hong, H. (2017). *Computational thinking for teacher education*. *Communications of the ACM, 60*(4), 55–62. https://doi.org/10.1145/2994591

Zhan, Z., Li, T., & Ye, Y. (2024). Effect of jigsaw-integrated task-driven learning on students' motivation, computational thinking, collaborative skills, and programming performance in a high-school setting. *Computer Applications in Engineering Education, 32*(1), 45–63. https://doi.org/10.1002/cae.22793

Authors

Feyza KURBAN, Ministry of National Education, Burdur (Turkey). E-mail: feyza.1526@gmail.com

Hüseyin Bahadır YANIK, Anadolu University, Eskişehir (Turkey). E-mail: hbyanik@anadolu.edu.tr

Ayşegül ERYILMAZ ÇEVİRGEN, Anadolu University, Eskişehir (Turkey). E-mail: aysegulec@anadolu.edu.tr

Burcu DURMAZ, Süleyman Demirel University, Isparta (Turkey). E-mail: drburcudurmz@gmail.com

Ethical Consideration

This study was conducted within the scope of a project supported by the Innovative Educational Practices Support Program of the Scientific and Technological Research Council of Turkey (Project/Grant No. 10231751; June 7, 2024). Necessary administrative permissions were obtained from the participants' schools prior to data collection. Participation was voluntary, informed consent was obtained from all participants, and all data were anonymized to ensure confidentiality.

Acknowledgement

This study was supported by the Innovative Educational Practices Support Program of the Science and Society Directorate of Turkey. The authors thank the mathematics teachers who participated in the study and the instructors who contributed to the professional development training.