Informatics in Education, 2025, Vol. 24, No. 2, 299–320

DOI: 10.15388/infedu.2025.12

Assessing Computational Thinking in Primary Education: A Cross-Sectional Study Using a Multidimensional Perspective

Javier DEL OLMO MUÑOZ[1*], Alicia JIMÉNEZ TOLEDO[2],
Sergio TIRADO OLIVARES[3], José Antonio GONZÁLEZ-CALERO[1],
Ramón CÓZAR GUTIÉRREZ[4]
[1]*Department of Mathematics, University of Castilla-La Mancha, Spain*
[2]*University of Castilla-La Mancha, Spain*
[3]*Department of Geography and Territorial Planning, University of Castilla-La Mancha, Spain*
[4]*Department of History, University of Castilla-La Mancha, Spain*
e-mail: javier.olmo@uclm.es, alicia.jtoledo@uclm.es, sergio.tirado@uclm.es, jose.gonzalezcalero@uclm.es, ramon.cozar@uclm.es
Received: December 2024

Abstract. The assessment of computational thinking (CT) is crucial for improving pedagogical practice, identifying areas for improvement, and implementing efficient educational interventions. Despite growing interest in CT in primary education, existing assessments often focus on specific dimensions, providing a fragmented understanding. In this research, a CT system of assessments for primary education was assembled and applied in a cross-sectional survey study with 1306 students from the 6th grade in a region of Spain. A three-way ANOVA and correlation analyses explored the effects of programming experience, educational context, and gender on CT skills and self-efficacy. Results highlighted a significant effect of programming experience but no significant effects of context or gender, alongside low overall correlations between CT skills and self-efficacy. These findings highlight the need to avoid focusing CT assessments on a single variable and support the combined use of multiple assessment instruments to measure CT accurately and effectively.

Keywords: computational thinking, system of assessment, cross-sectional study, primary education, programming experience, self-efficacy, problem-solving skills.

1. Introduction

Computational thinking (CT) is a set of cognitive and practical skills that enable people to solve problems systematically and efficiently. These problem-solving skills include,

* Corresponding Author:

among other aspects, breaking down complex situations into more manageable parts, identifying patterns and structures within those situations, abstracting irrelevant details to focus on the essentials, and formulating algorithms that provide step-by-step solutions to these problems (Barr & Stephenson, 2011; Grover & Pea, 2013).

Seymour Papert, a pioneer in the view that children can develop procedural thinking through programming, particularly through his work with the LOGO language, introduced the fundamental ideas of CT in his book *Mindstorms* (1980). In the eighties, those who foresaw the computer revolution made considerable efforts to integrate the so-called "computer literacy" – which includes understanding fundamental principles of computing – into teachers' curricula and subsequently into primary and secondary education (García-Vera, 1994). In Spain, the institutional incorporation of computing in schools began with *Project Atenea*, promoted by the Ministry of Education and Science in the mid-decade. This project already pointed out that "the possibilities of languages like LOGO seem great" (Arango, 1985, p. 8). However, not all these initiatives succeeded in consolidating or translating into a significant change in students' technological understanding (García-Vera, 1994).

Years later, at the beginning of the 21st century, the term "computational thinking" was popularized by Jeannette Wing (2006), who considered it a basic form of reasoning that should be taught alongside reading, writing, and arithmetic. Wing coined the term and defined it as a way of "solving problems, designing systems, and understanding human behavior by using fundamental concepts from computer science" (p. 33). Both Papert and Wing articulated the vision that everyone can benefit from learning to use principles, concepts, and approaches typical of computer science. Hence the importance of its integration into educational curricula.

Since then, the emergence of CT has marked a significant advance in awareness of the need to revisit some of these notions from the early educational stages, as reflected in the scientific production published since Wing's article (Piazza & Mengual-Andrés, 2020). This literature generally recognizes that CT is a set of cognitive processes mainly practiced in programming tasks, which favor new ways of thinking, communicating, and expressing ideas (Bers, 2021). However, these processes are also applicable to other disciplines (Barr & Stephenson, 2011; Mannila *et al.*, 2014). This speaks to the importance of incorporating CT beyond the technological field, as it is a key skill for anyone aspiring to fully participate in an increasingly technology-dependent and innovative society.

1.1. *Integration of Computational Thinking in Educational Curricula*

The integration of CT into school curricula has become a global priority. In America, the implementation of CT in primary schools is a growing concern. In the United States, the "Hour of Code" initiative was launched in 2013 by the non-profit organization Code.org (Code.org, 2024), aiming to disseminate computer science among all students. Additionally, in the United States, the "Computer Science for All" program seeks to offer all students the opportunity to participate in computer science and CT education at K-12 levels by training teachers, expanding access to high-quality teaching materials, and creating regional partnerships (Smith, 2016). In early primary education, the efficacy of brief intervention programs based on block-based programming, such as Coding as Another Language-ScratchJr, has been examined (Yang *et al.*, 2025). In other countries, such as Chile, Argentina, Uruguay, and Brazil, changes are being introduced in national curricula to include computing or digital technology skills, with many emphasizing CT skills specifically (Bers *et al.*, 2022; Brackmann *et al.*, 2016). In the Asia-Pacific region, countries with a strong information and communication technology industry, such as South Korea, Taiwan, Hong Kong, and China, have also implemented reforms to incorporate CT education from primary to secondary education, highlighting the importance of this skill in modern educational contexts (So *et al.*, 2020). In Japan, programming and CT education became mandatory in elementary schools in 2020, as part of a national strategy to enhance students' digital competencies and logical reasoning from an early age (Ohashi, 2024). Singapore has also made CT education compulsory at the upper primary level through the Code for Fun program, jointly developed by the Ministry of Education and the Infocomm Media Development Authority. Since 2020, all government and government-aided schools must offer this 10-hour program, which introduces students to computational thinking, coding, and emerging technologies such as artificial intelligence (Infocomm Media Development Authority, 2024).

In Europe, several countries, with Slovakia, Estonia, Finland, and the United Kingdom at the forefront, have reformed their compulsory education programs to include basic notions of computer science, as noted in the report *Reviewing Computational Thinking in Compulsory Education* by the Joint Research Centre of the European Commission (Bocconi *et al.*, 2022). In Spain, the formal integration of CT into the national curriculum came in 2022 with the Organic Law modifying the Organic Law of Education (LOMLOE), which emphasizes its incorporation from the early years of schooling (Organic Law 3/2020, 2020) and includes CT as a key competency to be developed by students. The aim is to ensure a solid foundation for building the development of CT concepts and processes throughout compulsory education and beyond. In this process, the evaluation of CT emerges as a crucial element to determine the effectiveness of its integration and implementation in educational curricula (Weintrop *et al.*, 2021).

1.2. *Assessment of Computational Thinking*

It is essential, therefore, to develop rigorous assessment tools that can accurately measure how students acquire and apply these skills in various contexts. To this end, in recent years, various methodologies and tools have been proposed to measure these abilities (Cutumisu *et al.*, 2019; Poulakis & Politis, 2021; Román-González *et al.*, 2019; Tang *et al.*, 2020). However, there are few widely accepted assessments that measure these skills in primary education (Cutumisu *et al.*, 2019), with a notable scarcity of validated instruments for assessing CT that are not associated with a specific learning environment or programming language (Zapata-Cáceres *et al.*, 2024). Additionally, Kampylis *et al.* (2023) emphasize the need for further investigation to understand better how issues related to gender, equity, and inclusion impact the quality of CT education integration. Moreover, given the multifaceted nature of CT, it is unlikely that a single instrument will be sufficient to record it exhaustively; instead, a system composed of several assessment instruments may be necessary (Grover, 2015; Guggemos *et al.*, 2023). In this regard, Kampylis *et al.* (2023) highlight the importance of combining different approaches, such as self-reporting tools and practical tasks, to ensure consistency and comprehensiveness in the evaluation of CT. Therefore, it is necessary to develop comprehensive assessment systems specifically designed for primary education.

Given the variety of existing types of CT assessment, different classifications have been proposed to organize them according to their approach (Guggemos *et al.*, 2023). One such classification is proposed by Román-González *et al.* (2019), where the following seven categories are identified: (1) *diagnostic tools,* that can be administered in pure pre-test situations or also post-test situations, to measure the level of CT aptitude; (2) *summative tools,* typically used as post-tests, intended to assess whether a person has gained sufficient content knowledge after instruction; (3) *formative-iterative tools*, which include tools aimed at providing usually automatic feedback to students, with the purpose of developing and improving their CT skills – this category mainly refers to CT skills acquired while programming since these tools are based on code diagnosis; (4) *data mining tools*, focused on the learning process and usually based on learning analytics, using data recorded on online educational platforms which teach CT through programming; (5) *skill transfer tools,* intended to assess how far students can transfer their CT skills across different types of problems, contexts and situations; (6) *perception and attitude scales*, aimed at assessing perceptions and attitudes about CT, but also related topics, such as computers, computer science, programming or digital literacy; (7) *vocabulary assessment*, which aims at measuring various elements and dimensions of CT using student verbal expressions.

Accordingly with this classification, assessment tools, which aim is measuring student levels of CT aptitude, constitute the most appropriate means to determine these levels of proficiency in isolation, as they involve performance tests that do not require specific prior knowledge such as a particular programming language (Guggemos *et al.*, 2023). On the other hand, skill transfer tools, such as Bebras tasks (Dagienė *et al.*, 2016), consist of real-life problems and, being independent of any learning environment, can also be administered to students without prior experience in the subject. Thirdly, perception and attitude scales, such as the self-assessment instrument Computational Thinking Scales (CTS) (Korkmaz *et al.*, 2017), also allow for the isolated assessment of students' cognitive ability and attitudes towards their CT skills and are suitable for large-scale assessment (Sun *et al.*, 2021a). These three types of tools have in common that they allow for the objective assessment and measurement of students' CT skills without requiring prior knowledge or experience.

Recent systematic reviews support this multidimensional perspective. Rao and Bhagat (2024), after analyzing 360 studies, highlight the coexistence of diverse assessment strategies – from self-reporting instruments to problem-solving tasks and automated tools – and emphasise that no single method can adequately capture the complexity of CT. Yeni *et al.* (2024) reach similar conclusions, noting that existing tools often prioritise attitudes or domain-specific outcomes over direct assessment of CT skills. While their emphases differ, both reviews highlight the limitations of using a single assessment approach and stress the importance of combining diverse tools to adequately evaluate CT in educational contexts.

1.3. *Influential Factors in the Development of Computational Thinking*

Research on students' development of CT shows that several factors can influence the acquisition and refinement of these skills. Among these factors are previous programming experience, which provides a solid foundation and familiarity with some basic concepts; gender, which may entail differences in access and motivation towards technological disciplines; and educational context, which ranges from teaching quality to available resources and institutional support. These factors are analyzed in greater detail below.

Firstly, students' prior programming experience can influence their self-efficacy and performance when solving CT skill transfer tasks. Specifically, previous studies have demonstrated positive associations between training or prior experience and levels of self-efficacy (Ineson *et al.*, 2013; Prieto & Altmaier, 1994), which in turn positively influences performance (Honicke & Broadbent, 2016; Richardson *et al.*, 2012). This relationship aligns with findings from Gümüş *et al.* (2024), who measured middle school students' digital literacy, programming self-efficacy, and CT self-efficacy using self-reported scales. While self-reported measures capture participants' perceptions of their skills, they may not always align with actual performance. Nonetheless, these perceptions play a critical role in shaping confidence and motivation, both of which are essential for skill development. Therefore, students previously exposed to programming should show a higher level of competence and confidence in their abilities.

Secondly, differences between boys and girls in education and digital skills are a widely debated topic. Women are underrepresented when choosing a degree related to computer science, and stereotypes about who is good at CT and programming begin to manifest at an early age (Bers, 2021). Previous research points to disparities between perception and academic performance among boys and girls in STEM (science, technology, engineering, and mathematics) subjects. For example, in mathematics, girls may perform as well or better than boys but are more likely to show less confidence in their mathematical abilities compared to their actual performance (OECD, 2023). On the other hand, boys tend to have higher mathematical self-efficacy, partly because they experience more positive feelings and higher cognitive self-esteem (Zander *et al.*, 2020). Regarding CT, studies like that by Kallia and Sentance (2018) suggest that boys usually feel more competent in computing than girls do besides making significantly more accurate predictions (better calibrated, according to the authors) about their performance in programming. In addition, Sun *et al.* (2022) found that boys have better attitudes towards programming, despite having lower CT skills than girls. Nevertheless, other studies have reported better CT skills in boys (Sun *et al.*, 2021b), while others have not found significant differences regarding gender (Zhong *et al.*, 2016).

Thirdly, concerning educational context, differences between urban and rural areas could influence students' CT skills levels. Simmonds *et al.* (2019) note that, in Latin America, such level differences are due to these programs rarely reaching rural areas, which means the digital gap between urban and rural areas only increases. In India, rural teachers have less training in computer science, making it difficult to teach computational thinking (Shah, 2019). On the other hand, the report *What Makes Urban Schools Different?* (OECD, 2013) notes that, in Spain, students in urban areas tend to perform better because schools are usually larger, enjoy a better socioeconomic status, and have greater autonomy in allocating educational resources. Although recent initiatives have sought to reduce these disparities and enhance rural education, significant challenges remain (Rodríguez *et al.*, 2023).

1.4. *Study Objectives*

Based on the above, the present study seeks to achieve the following objectives:

- **O1:** To examine whether there are significant differences based on previous programming experience, gender, and educational context in the levels of CT skills and CT self-efficacy among 6th-grade students.
- **O2:** To evaluate possible associations between 6th-grade students' CT skills and their self-efficacy based on the previous variables.

2. Methods

2.1. *Design*

This research is characterized as a cross-sectional and survey study (Creswell & Guetterman, 2019), in which the CT of the participants is quantitatively evaluated from two perspectives – self-efficacy and skills. The choice of a cross-sectional design is justified by its ability to capture the state of a given variable in a specific population at a specific moment, which is well-suited for a study that assesses the current level of abilities or attitudes of students quickly and efficiently (Cohen *et al.*, 2018).

2.2. *Participants*

The criteria for participant inclusion were that students were undertaking the last year of primary education in the autonomous community of Castilla-La Mancha and that they had express consent from their parents or legal guardians to carry out the evaluation. The selection of participants was based on the characteristics of the educational centers where they were taking their last year in primary education. Efforts were made to ensure that all chosen schools had a representative percentage of each type (rural/

Table 1
Distribution of participants by gender, educational context, and programming experience

Programming Experience	Rural		Urban		Total
	Girls	Boys	Girls	Boys	
None	57	56	60	75	248
Low	114	99	259	301	773
High	38	32	97	118	285
Total	209	187	416	494	1306

urban) concerning all centers in Castilla-La Mancha. In fact, to ensure representativeness and enable subgroup comparisons, a stratified sampling approach was adopted based on the educational context (urban vs. rural). The total population comprised 16,207 urban and 5,782 rural students at this grade. To determine the minimum required sample sizes for each stratum, we applied stratified sample size estimation formulas assuming a 95% confidence level, a 5% margin of error, and maximum variability (p = 0.5). For the rural population, the calculated minimum sample size was approximately 360 students; for the urban population, the required sample size was approximately 375 students.

To determine if a school was rural, those located in localities with a maximum population of 5,000 inhabitants were considered, following criteria used by Law 45/2007 (2007) to define small-sized rural municipalities.

In total, 1306 sixth-grade students participated. This grade was chosen as it marks a critical transition from primary to secondary education, where students are expected to develop foundational skills that will support their success in more complex learning environments (Ávila Francés *et al.*, 2022). Participants were distributed in 57 schools: 27 urban and 30 rural ones. Table 1 shows these students' distribution by gender (boys or girls), educational context (rural or urban), and previous experience in programming (none, low or high). As it can be seen in Table 1, 396 rural and 910 urban students participated, thus exceeding the required sample size in both strata.

2.3. *Instruments*

To measure the participants' CT, an evaluation system composed of two instruments was assembled. Based on Román-González *et al.* (2019), this evaluation system was used with an assessment approach. However, each of its areas has different characteristics that we detail below.

Before presenting the main body of the questionnaire, some identification questions were posed regarding the gender and previous programming experience of participants, among others. Programming experience was self-reported through a single multiple-choice question that asked whether students had any previous experience with programming, either on a computer or using a robot. The response options were: none, a little, or a lot, corresponding to the categories used in the analyses: none, low, and high. This categorization aimed to balance clarity and simplicity for young respondents, avoiding excessive granularity or cognitive overload. A "medium" category was intentionally excluded to reduce ambiguity and ensure that the distinctions remained accessible and meaningful for primary school students.

2.4. *Self-efficacy*

To assess students' self-efficacy towards CT, we used the self-efficacy assessment from *PESS (Programming Experience, Self-efficacy and Skills)* instrument designed by Mannila *et al.* (2020). The original instrument, written in English, was translated into Spanish by two native English-speaking lecturers at University of Castilla-La Mancha with academic training in Spanish language and education. The items in the self-efficacy section were general in nature and did not reference specific cultural or curricular elements, so no adaptation was required. The translated items were piloted in classroom contexts with students matching the study's target age group, confirming the clarity and appropriateness of the items. Regarding the validation of the instrument, Mannila *et al.* (2020) report that it was developed and piloted in two phases: first through an expert panel and then in classroom settings with 310 students. However, the authors do not provide information on the instrument's reliability. Nevertheless, the data collected in the context of our study yielded a Cronbach's alpha of 0.79, indicating acceptable internal consistency.

The original instrument has three areas: experience, self-efficacy, and skills. While the skills area adheres to the contents of the Swedish curriculum – the country in which Mannila *et al.*'s study is framed; the questions of experience and self-efficacy are more general in nature. On the other hand, given that the experience section of the original instrument consisted of 20 items, it was too extensive for the scope of our study, so only the self-efficacy section was used. According to the classification proposed by Román-González *et al.* (2019), this area of the instrument would be considered as a scale of perceptions and attitudes. With it, we evaluate eleven dimensions, categorized into six concepts (logical thinking, algorithms, decomposition, pattern recognition, abstraction, and evaluation) and five practices (tinkering, creating, debugging, persevering, and collaborating) (Barefoot CAS, 2016). In total, it has 22 items – two for each dimension – configured as a five-level Likert scale, so its scoring scale ranges from 1 to 5 (from "strongly disagree" to "strongly agree"). An example of the questionnaire format and the items incorporated is shown in Fig. 1.

		Strongly disagree	Disagree	Neither agree nor disagree	Agree	Strongly agree	Don't know / NA
05	I would rather solve one big problem than several small ones.						
06	It is easier to solve a problem if I first decompose it into smaller parts.						

Fig. 1. Sample self-efficacy items assessing the Decomposition dimension of CT.

2.5. *Skills*

We employed the instrument developed and psychometrically validated by Li *et al.* (2021), which was designed to assess the CT skills of primary education students, from 3rd to 6th grade (9 to 12 years old). This tool was selected due to its suitability for our target age group and because it is one of the few available instruments in this category that has undergone psychometric validation. Notably, the instrument demonstrated satisfactory internal consistency, with a Cronbach's alpha coefficient of 0.76, indicating reliable measurement (Li *et al.*, 2021). The instrument was translated into Spanish by the Confucius Institute at the University of Castilla-La Mancha, a center specialized in Chinese language and culture. Minor contextual adaptations were made, such as modifying the character names used in the word problems to align with names familiar to Spanish students. The structure, content, and difficulty level of the items were otherwise preserved. A review confirmed that no further cultural or contextual adjustments were required, as the items were designed to be independent of specific programming environments or local references. To ensure clarity and comprehension, a pilot implementation was conducted with students of the same age range in a regular classroom setting. The authors describe it as a summative and aptitude assessment of CT, similar to Bebras tasks (Dagienė *et al.*, 2016) or the *Computational Thinking test* (Román-González, 2015). According to the classification by Román-González *et al.* (2019), it can be classified as a skill transfer tool.

This instrument measures five dimensions of CT: abstraction, algorithmic thinking, decomposition, evaluation, and pattern recognition. It is assembled to be administered in a maximum time of 45 minutes and consists of 25 multiple-choice tasks with four

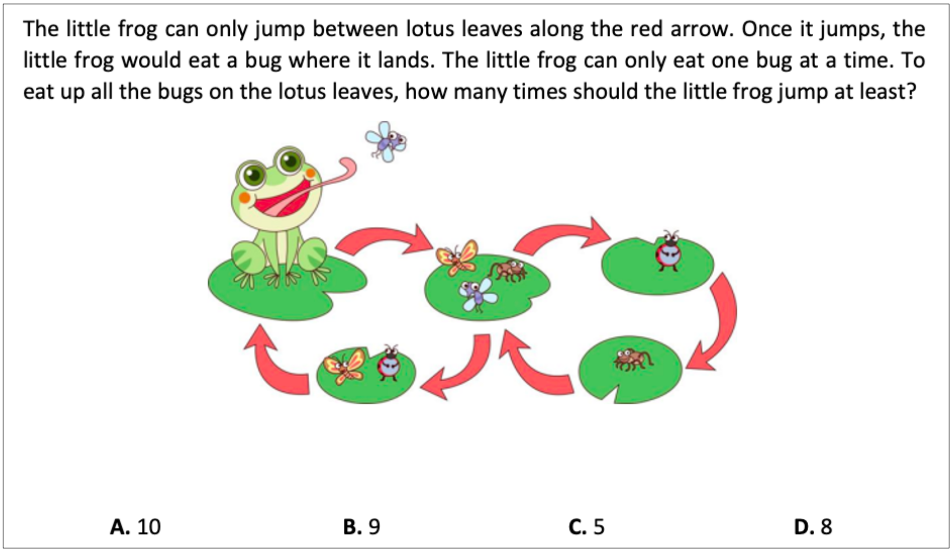

Fig. 2. Example of a skill item designed to assess the Decomposition dimension of CT.
Source: Li *et al.* (2021).

response options, contextualized in real-life situations. Each problem refers to at least a computational skill or concept (see Fig. 2 for an example).

2.6. *Procedure*

The study was conducted in accordance with the Declaration of Helsinki and was approved by the Social Research Ethics Committee (SREC) of the University of Castilla-La Mancha (Ethics Code: CEIS-714230-H0P3). In this context, all participants provided written informed consent signed by their parents or legal tutors prior to enrolment in the study. The evaluation system was presented to the students in a classroom-group context, avoiding taking them out of their usual environment. However, each student completed the questionnaire individually. The assessment was administered in paper format, allowing students to write their responses directly and enabling observation of intermediate steps in the skills section.

To guide the students on how to respond to each section of the questionnaire and control the maximum time for completion (60 minutes), a video was made. This video included two trial questions for the self-efficacy part and one for the skills part. The purpose of these questions was to graphically explain how to respond to the rest of the test without influencing their responses. With this video, it was also sought to ensure that all participants, regardless of their group, received information uniformly. Apart from these instructions, no additional explanations were provided, with the sole exception that teachers present in the classroom during the administration of the questionnaire were allowed to clarify the meaning of any word if a student indicated they did not understand it.

Regarding the order of tests, although they were all performed in the same block of time, the video first scheduled the experience part, followed by the self-efficacy part, and finally the skills part.

2.7. *Data analysis*

In line with the research aims, the statistical analysis focuses on identifying potential differences in the scores for CT skills and self-efficacy based on students' gender, educational context, and programming experience, as well as examining possible associations between these variables. For each participant, the CT skills score was calculated as the arithmetic mean of the scores obtained for each question in the skills section. Each response was coded in a binary manner, assigning one point for a correct answer and zero points otherwise. The total score obtained in the CT skills test was adjusted to a 10-point scale to enhance interpretability and to align with standard academic evaluation practices in the country in which the study was conducted. Similarly, the self-efficacy score was calculated for each participant as the average of their responses to each question in the self-efficacy section of the questionnaire. However, self-efficacy scores were retained on a Likert scale ranging from 1 to 5 points to preserve the interpretation of the original scale. Regarding the prior programming experience question, students' responses – originally "never," "a little," and "a lot" – were recategorized as "none," "low," and "high" to represent varying levels of experience more clearly.

Concerning the statistical analysis aligned with the first objective (O1), a three-factor ANOVA was conducted separately for CT skills and self-efficacy scores, considering three independent variables: prior programming experience (none, low or high), gender (boys or girls) and educational context (urban or rural). In cases of significant interactions between two variables, two-factor ANOVA tests were performed for those variables. If no significant interactions were observed, main effects were explored using one-way ANOVA. Prior to conducting the ANOVA analyses, the assumptions of normality and homogeneity of variances were evaluated. Given the large sample size, the normality of residuals was assessed graphically using histograms and Q – Q plots. Homogeneity of variances was also assessed using Levene's test and complementary procedures that support the use of parametric tests.

Furthermore, for the prior programming experience variable (which includes more than two levels), post-hoc pairwise analyses were conducted to identify potential differences between these levels. Effect sizes were interpreted using partial eta squared for the ANOVA analyses and Cohen's d for pairwise comparisons. To address the second objective (O2), Pearson correlation analyses were conducted for each independent variable to examine the relationship between CT skills and self-efficacy. Before performing Pearson correlation analyses, the required assumptions of normality, linearity, and the absence of significant outliers were examined.

3. Results

First, Table 2 presents a summary of the descriptive statistics for both CT skills and self-efficacy across the subgroups defined by the three independent variables: gender, educational context, and programming experience. Subsequently, to ensure a clear and structured analysis, the results are presented in alignment with the two research objectives outlined in the study.

3.1. *Results Related to O1*

In relation to the first objective (O1), the results of three-way ANOVA are summarized in Table 3. As shown, no significant interaction effects were found among the variables programming experience, educational context, and gender in predicting CT skills. However, significant main effects were observed for each of the three factors independently. Specifically, programming experience emerged as the most influential variable, followed by gender and educational context, although the effect sizes for the latter two were very small. Regarding CT self-efficacy, only programming experience showed a statistically significant main effect, with a moderate effect size. None of the

Table 2

Descriptive statistics by group

Group	$M_{CT\ skills}$	$SD_{CT\ skills}$	$M_{CT\ self\text{-}efficacy}$	$SD_{CT\ self\text{-}efficacy}$	n
Gender					
Girl	3.64	1.16	3.72	0.42	625
Boy	3.80	1.29	3.75	0.44	681
Educational context					
Rural	3.81	1.16	3.74	0.43	396
Urban	3.68	1.26	3.73	0.43	910
Programming experience					
None	3.62	1.20	3.64	0.45	248
Low	3.67	1.20	3.70	0.43	773
High	3.95	1.31	3.89	0.39	285

Table 3

Summary of three-way ANOVA results for CT skills and CT self-efficacy

Dependent variable	Factor(s)	$F(df)$	p	Partial η^2	Effect Size
CT skills	Programming experience	6.22 (2, 1294)	.0021	.010	Small
	Educational context	4.06 (1, 1294)	.0441	.003	Very small
	Gender	6.60 (1, 1294)	.0103	.005	Very small
	Prog. Exp. × Context	0.05 (2, 1294)	.9524	<.001	Negligible
	Prog. Exp. × Gender	0.07 (2, 1294)	.9311	<.001	Negligible
	Context × Gender	0.39 (1, 1294)	.5326	<.001	Negligible
	Prog. Exp. × Context × Gender	0.48 (2, 1294)	.6215	<.001	Negligible
CT Self-efficacy	Programming experience	25.87 (2, 1292)	<.0001	.040	Moderate
	Educational context	1.32 (1, 1292)	.2500	.001	Negligible
	Gender	1.92 (1, 1292)	.1670	.001	Negligible
	Prog. Exp. × Context	0.70 (2, 1292)	.4950	.001	Negligible
	Prog. Exp. × Gender	0.10 (2, 1292)	.9050	<.001	Negligible
	Context × Gender	0.78 (1, 1292)	.3790	.001	Negligible
	Prog. Exp. × Context × Gender	0.22 (2, 1292)	.8050	<.001	Negligible

other main effects or interaction terms reached significance, and their effect sizes were negligible.

To further explore the impact of programming experience, post-hoc analyses were conducted for both CT skills and CT self-efficacy. The results, presented in Table 4, indicate that students with high levels of prior programming experience significantly outperformed those with no or low experience on both outcome variables. In contrast, no significant differences were found between students with no experience and those with low experience. The effect sizes associated with the significant comparisons ranged from small to moderate, with stronger effects observed in the self-efficacy domain.

Table 4

Post-hoc comparisons for CT skills and CT self-efficacy by programming experience

Dependent variable	Comparison	Mean Difference	*p*	*Cohen's d*	95% CI for *d*	Effect size
CT skills	High vs. None	0.81	.0067	0.27	[0.09, 0.44]	Small – moderate
	High vs. Low	0.68	.0039	0.22	[0.09, 0.36]	Small
	Low vs. None	-0.13	.8332	-0.04	[-0.19, 0.10]	Negligible
CT self-efficacy	High vs. None	0.24	<.0001	0.57	[0.40, 0.75]	Moderate
	High vs. Low	0.18	<.0001	0.43	[0.29, 0.57]	Moderate
	Low vs. None	-0.06	.1160	-0.15	[-0.29, -0.00]	Small

In summary, these results suggest that previous programming experience has a significant impact on CT self-efficacy scores, whereas educational context and gender do not have a significant influence, either independently or in interaction with other factors. The post hoc comparisons indicate that individuals with high programming experience have significantly higher CT self-efficacy scores values compared to those with low or no experience, underscoring a positive relationship between programming experience and performance in this self-efficacy measure.

3.2. *Results Related to O2*

In relation to the second objective (O2), Pearson correlation analyses were performed to examine the association between students' CT skills and CT self-efficacy across different subgroups (Table 5), considering previous programming experience (none, low or high), educational context (urban or rural), and gender (boys or girls). Statistically significant but very weak correlations were observed among students with low programming experience, those attending urban schools, and male students. In contrast, the associations were not significant in the remaining subgroups, and effect sizes across all cases were minimal.

Table 5

Pearson correlations between CT skills and CT self-efficacy across subgroups

Subgroup	*r* (Pearson)	*p-value*		Interpretation
High Programming Experience	0.00	.9510		None
Low Programming Experience	0.09	.0094	*	Very weak
No Programming Experience	0.09	.1440		Very weak
Urban Context	0.10	.0034	*	Very weak
Rural Context	0.07	.0692		Very weak
Boys	0.10	.0089	*	Very weak
Girls	0.07	.0758		Very weak

4. Discussion

The assessment system presented in this study directly addresses the demands identified in the literature regarding the scarcity of comprehensive systems for assessing CT (Grover, 2015; Guggemos *et al.*, 2023; Rao & Bhagat, 2024). Previous studies have examined how programming instruction enhances CT skills, with a focus on various programming modalities and tools. For example, Cetin *et al.* (2023) investigated the impact of different programming instruction modalities (mBlock, Scratch, and Python) on sixth-grade students. Their findings demonstrated the benefits of using constructionist block-based and robotics programming environments, although no significant differences were observed in the development of CT skills across the groups. On the other hand, the systematic review by Fagerlund *et al.* (2021) on the use of Scratch in primary schools concluded that this block-based programming language allows students to interact with programming content that fosters CT skills. However, the authors emphasized the need for more concrete and comprehensive assessment methods to accurately measure CT skill development.

Specifically, our study contributes important nuances by directly responding to the need for assessments that are not tied to a specific learning environment or programming language (Zapata-Cáceres *et al.*, 2024) by integrating both practical tasks and self-report tools, in line with the recommendations of Kampylis *et al.* (2023) and Yeni *et al.* (2024) for more comprehensive and consistent methods to measure CT. In addition, issues related to gender, equity, and inclusion were carefully considered during the analysis of the results, adhering to calls for greater attention to these factors to ensure quality and equity in CT education (Kampylis *et al.*, 2023).

Regarding the results obtained, we analyzed differences in CT skills and self-efficacy scores among 6th-grade students based on gender, educational context and previous programming experience (O1), and examined potential associations between these variables (O2).

For O1, no statistically significant effects were observed for CT skills or self-efficacy with respect to gender, educational context, or programming experience in the two-way and three-way interactions. However, a significant main effect of programming experience on both CT skills and self-efficacy was evident, extending findings from previous studies on the influence of academic self-efficacy on academic performance (Ineson *et al.*, 2013; Honicke & Broadbent, 2016). Specifically, there is a significant difference between those students who reported to have high programming experience compared to those with low experience or no experience. This suggests that a greater exposure to programming positively influences students' CT, a finding consistent with Sun *et al.* (2022), who highlighted that programming experience contributes to CT development, particularly when combined with positive attitudes toward programming.

Despite the presence of certain gender stereotypes concerning CT (Bers, 2021), our study reveals no significant differences either on CT skills or self-efficacy regarding gender. Although some research has reported higher CT skills in girls (e.g., Sun *et al.*, 2022) and others in boys (e.g., Sun *et al.*, 2021b), our findings align with those which did not identify significant gender differences (e.g., Zhong *et al.*, 2016). Regarding self-efficacy, previous research often reports higher scores for boys (Kallia & Sentance, 2018; Sun *et al.*, 2022). However, in our study, no significant gender-related differences were detected.

Even if there is a lack of research on CT differences between urban and rural areas, urban schools in Spain have historically tended to achieve better academic results, which can be attributed to a broader range of resources and the urban socioeconomic context (OECD, 2013). However, recent years have seen efforts to address these disparities, particularly through the inclusion and promotion of projects and initiatives aimed at improving rural education, though significant challenges remain (Rodríguez *et al.*, 2023). Similar trends are observed in other regions of the world, such as Latin America and India, where a digital gap has been identified (Simmonds *et al.*, 2019; Shah, 2019). Nevertheless, our study did not identify significant differences between urban and rural students, which can be attributed to the equitable provision to digital resources and teacher training implemented by the regional government.

Regarding O2, we identified statistically significant but very weak correlations between CT skills and self-efficacy in the case of students with low experience, those belonging to urban schools and boys. Consequently, the overall results suggest a low association between CT skills and CT self-efficacy in the studied sample. This seems to indicate that students' practical CT skills are not a direct reflection of their confidence in these skills. This finding contrasts with previous studies, such as that of Kallia and Sentance (2018), which suggest that, in terms of gender, boys typically perceive themselves as more competent in computing than girls and make significantly more accurate predictions – better calibrated, according to the authors – about their programming performance, which is related to CT.

The lack of strong associations could reflect several underlying factors. On one hand, it could indicate that students are still in the process of developing and consolidating these skills, which aligns with the initial stage of the gradual integration of CT into school curricula in Spain at the time of the development of the study. On the other hand, this, coupled with the high scores observed in CT self-efficacy, could point to a possible overestimation of abilities, a common phenomenon at these ages, where children may exhibit unrealistic optimism about their skills (Wu *et al.*, 2021).

5. Conclusion

The last year of primary education marks the transition to secondary education and represents a pivotal stage where foundational skills such as CT play a critical role in shaping students' academic trajectories. This study contributes to understanding the factors influencing these skills. The lack of significant effects of gender and educational context on CT skills and self-efficacy, as well as the absence of significant effects of interactions between the different groups, contrary to the significant main effect of programming experience on these scores, may underline the importance of programming in developing CT skills in the school. Specifically, the difference between students with high programming experience and those with low or no experience indicates that intensive programming interventions, which work on several CT aspects, have a positive impact on CT performance. In other words, although basic or superficial programming interventions may have some impact on coding abilities, they fail to address the deeper components of CT, as previously found in Yang *et al.* (2025). Therefore, it is necessary to implement more comprehensive programming and CT practices in the classroom to fully develop CT.

These findings also provide a nuanced perspective on previous research regarding self-assessment measures. For example, Gümüş *et al.* (2024) observed significant relationships between programming self-efficacy and CT self-efficacy in middle school students aged 10–13, based on self-reported measures. While their work emphasizes the connections between self-efficacy constructs, our study extends this understanding by focusing on the relationship between CT self-efficacy and actual CT skills, where no significant correlation was found. This divergence may highlight the difference between self-perceptions and objectively assessed competencies, underscoring the need for educational practices that nurture both aspects. Self-efficacy remains critical, as it directly influences motivation and engagement, and fostering students' confidence alongside technical skills is essential for their balanced development in CT.

The variability in the observed correlations between CT skills and self-efficacy, as well as the contextual, prior experience, and gender differences that do not follow a clear pattern, highlight the complexity of effectively measuring these skills. While the lack of significant effects of gender and educational context on CT skills and self-efficacy may suggest these variables play a secondary role compared to programming experience, it also underscores the need for more nuanced assessment methods. By using both problems to assess specific CT skills and questionnaires to measure self-efficacy, the study has been able to evaluate various facets of students' CT competence. However, the low correlation between these measures confirms that each instrument measures substantially different aspects of these skills.

This disparity highlights the previously stated need in the literature (Grover, 2015; Guggemos *et al.*, 2023) for a more comprehensive approach to CT assessment that does not rely solely on one type of instrument, as integrating multiple forms of assessment can help provide a more complete picture of students' skills.

5.1. *Limitations and Future Guidelines*

Despite the conclusive results obtained in this study, some limitations can be identified. On the first place, the use of a paper-based questionnaire prevented reaching more participants and complicated the data management process. However, this choice was made intentionally, as it allowed participants to write intermediate steps when solving problems, providing deeper insights into their thought processes. Future research could explore digital alternatives that preserve this feature while facilitating broader participation and more efficient data handling.

Additionally, students' prior programming experience was self-reported, which may introduce certain biases, such as memory inaccuracies or social desirability effects. This is particularly relevant given that no specific question was included about unplugged programming experiences (e.g., tangible or screen-free activities), which are increasingly common in early CT education. Such activities may be harder for students to recognize or recall as "programming", compared to more explicit experiences involving robots or screens. As a result, the variable may not fully capture the diversity of students' prior exposure to programming-related practices. Nevertheless, given the lack of reliable and objective indicators – especially in countries where such experiences often take place in informal or extracurricular contexts – self-report remains a practical and widely used approach to gather background information in studies of this kind.

In addition, while this study analyzed the effects of personal, social, and contextual factors on CT skills and CT self-efficacy, as well as the association between these variables, a deeper analysis concerning CT dimensions could provide further valuable insights. Finally, future research could examine how educational interventions concerning CT and programming influence broader competencies, such as problem solving, among Primary Education students. As highlighted in this study, recognizing and embracing the complexity of CT's polyhedral nature is crucial, calling for educational strategies that respond to its varied cognitive and practical challenges.

Funding and Acknowledgements

This research has been conducted within the framework of the collaboration agreement between the Council of Education, Culture, and Sports of the Government of Castilla-La Mancha and the University of Castilla-La Mancha; the project TED2021-131557-B-I00 funded by MICIU/AEI /10.13039/501100011033 and by the European Union NextGenerationEU/ PRTR; the project PID2023-150960NB-I00 funded by MICIU/AEI/10.13039/501100011033 and by "ERDF A way of making Europe"; and the grant 2022-GRIN-34039 by the University of Castilla-La Mancha and the European Regional Development Fund (ERDF). We also extend our gratitude to the Confucius Institute of the University of Castilla-La Mancha for their collaboration in the translation of the CTA-CES instrument.

Declaration of conflicting interests:

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

Ethical considerations:

The study was conducted in accordance with the Declaration of Helsinki and was approved by the Social Research Ethics Committee (SREC) of the University of Castilla-La Mancha (Ethics Code: CEIS-714230-H0P3) on June 1st, 2023.

Consent to participate:

All participants provided written informed consent signed by their parents or legal tutors prior to enrolment in the study.

Data availability:

The datasets generated during and/or analyzed during the current study are not publicly available because they belong to a collaboration agreement between the Council of Education, Culture, and Sports of the Government of Castilla-La Mancha and the University of Castilla-La Mancha, but are available from the corresponding author upon reasonable request.

References

Arango, J. (1985). El Proyecto Atenea: un plan para la introducción nacional de la informática en la escuela. *Revista de Educación*, *276*, 5–12. https://www.educacionfpydeportes.gob.es/revista-de-educacion/numeros-revista-educacion/numeros-anteriores/1985/re276/re276-01.html

Ávila Francés, M., Sánchez Pérez, M. C., & Bueno Baquero, A. (2022). Factores que facilitan y dificultan la transición de educación primaria a secundaria. *Revista de Investigación Educativa, 40*(1), 147–164. https://doi.org/10.6018/rie.441441

Barefoot CAS. (2016). *The computational thinker: Concepts and approaches*. http://barefootcas.org.uk/

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12. *ACM Inroads*, *2*(1), 48. https://doi.org/10.1145/1929887.1929905

Bers, M. U. (2021). Coding, robotics and socio-emotional learning: developing a palette of virtues. *Pixel-Bit, Revista de Medios y Educación*, *62*, 309–322. https://doi.org/10.12795/pixelbit.90537

Bers, M. U., Strawhacker, A., & Sullivan, A. (2022). The state of the field of computational thinking in early childhood education. In *OECD Education Working Papers* (Issue 274). https://doi.org/https://dx.doi.org/10.1787/3354387a-en

Bocconi, S., Chioccariello, A., Kampylis, P., Dagiené, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M. A., Jasuté, E., Malagoli, C., Masiulionyté-Dagiené, V., & Stupuriené, G. (2022). *Reviewing Computational Thinking in Compulsory Education* (A. Inamorato Dos Santos, R. Cachia, N. Giannoutsou, & Y. Punie (eds.)). Publications Office of the European Union. https://doi.org/10.2760/126955

Brackmann, C. P., Barone, D., Casali, A., Boucinha, R., & Munoz-Hernandez, S. (2016). Computational thinking: Panorama of the Americas. *2016 International Symposium on Computers in Education (SIIE)*, 1–6. https://doi.org/10.1109/SIIE.2016.7751839

Cetin, I., & Otu, T. (2023). The Effect of the Modality on Students' Computational Thinking, Programming Attitude, and Programming Achievement. *International Journal of Computer Science Education in Schools, 6*(2). https://doi.org/10.21585/ijcses.v6i2.170

Code.org. (2024). *Hour of Code*. https://hourofcode.com/

Cohen, L., Manion, L., & Morrison, K. R. B. (2018). *Research Methods in Education* (8th ed.). Routledge. https://lccn.loc.gov/2017015256

Cutumisu, M., Adams, C., & Lu, C. (2019). A Scoping Review of Empirical Research on Recent Computational Thinking Assessments. *Journal of Science Education and Technology*, *28*(6), 651–676. https://doi.org/10.1007/s10956-019-09799-3

Creswell, J. W., & Guetterman, T. C. (2019). *Educational research: Planning, conducting, and evaluating quantitative and qualitative research* (6th ed.). Pearson.

Dagienė, V., Sentence, S., Sentance, S., & Sentence, S. (2016). It's Computational Thinking! Bebras Tasks in the Curriculum. *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, *9973 LNCS*, 28–39. https://doi.org/10.1007/978-3-319-46747-4_3

Fagerlund, J., Häkkinen, P., Vesisenaho, M., & Viiri, J. (2021). Computational Thinking in Programming with Scratch in Primary Schools: A Systematic Review. *Computer Applications in Engineering Education, 29*(1), 12–28. `https://doi.org/10.1002/cae.22255`

García-Vera, A. B. (1994). Entre la cultura y la alfabetización informática. *Pixel-Bit. Revista de Medios y Educació*, *2*, 89–100.

Grover, S. (2015). "Systems of Assessments" for Deeper Learning of Computational Thinking in K-12. *Annual Meeting of the American Educational Research Association*.

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`

Guggemos, J., Seufert, S., & Román-González, M. (2023). Computational Thinking Assessment – Towards More Vivid Interpretations. *Technology, Knowledge and Learning*, *28*(2), 539–568. `https://doi.org/10.1007/s10758-021-09587-2`

Gümüş, M. M., Kukul, V., & Korkmaz, Ö. (2024). Relationships between middle school students' digital literacy skills, computer programming self-efficacy, and computational thinking self-efficacy. *Informatics in Education, 23*(3), 571–592. `https://doi.org/10.15388/infedu.2024.20`

Hansen, K., & Henderson, M. (2019). Does academic self-concept drive academic achievement? *Oxford Review of Education*, *45*(5), 657–672. `https://doi.org/10.1080/03054985.2019.1594748`

Honicke, T., & Broadbent, J. (2016). The influence of academic self-efficacy on academic performance: A systematic review. *Educational Research Review*, *17*, 63–84. `https://doi.org/10.1016/j.edurev.2015.11.002`

Ineson, E. M., Jung, T., Hains, C., & Kim, M. (2013). The influence of prior subject knowledge, prior ability and work experience on self-efficacy. *Journal of Hospitality, Leisure, Sport & Tourism Education*, *12*(1), 59–69. `https://doi.org/10.1016/j.jhlste.2012.11.002`

Infocomm Media Development Authority. (2024, November 28). *Code for Fun Programme*. `https://www.imda.gov.sg/how-we-can-help/code-for-fun`

Kallia, M., & Sentance, S. (2018). Are boys more confident than girls? In *Proceedings of the 13th Workshop in Primary and Secondary Computing Education* (pp. 1–4). ACM. `https://doi.org/10.1145/3265757.3265773`

Kampylis, P., Dagienė, V., Bocconi, S., Chioccariello, A., Engelhardt, K., Stupurienė, G., Masiulionytė-Dagienė, V., Jasutė, E., Malagoli, C., Horvath, M., & Earp, J. (2023). Integrating Computational Thinking into Primary and Lower Secondary Education: A Systematic Review. *Educational Technology & Society, 26*(2), 99–117. `https://www.jstor.org/stable/48720999`

Korkmaz, Ö., Çakir, R., & Özden, M. Y. (2017). A validity and reliability study of the computational thinking scales (CTS). *Computers in Human Behavior*, *72*. `https://doi.org/10.1016/j.chb.2017.01.005`

Ley 45/2007, de 13 de diciembre, para el desarrollo sostenible del medio rural, Boletín Oficial del Estado, 299, de 14 de diciembre de 2007. `https://www.boe.es/eli/es/l/2007/12/13/45/con`

Ley Orgánica 3/2020, de 29 de diciembre, por la que se modifica la Ley Orgánica 2/2006, de 3 de mayo, de Educación, Boletín Oficial del Estado, 340, de 30 de diciembre de 2020. `https://www.boe.es/eli/es/lo/2020/12/29/3`

Li, Y., Xu, S., & Liu, J. (2021). Development and Validation of Computational Thinking Assessment of Chinese Elementary School Students. Journal of Pacific Rim Psychology, 15, 183449092110102. `https://doi.org/10.1177/18344909211010240`

Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014). Computational Thinking in K-9 Education. *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference*, 1–29. `https://doi.org/10.1145/2713609.2713610`

Mannila, L., Heintz, F., Kjällander, S., & Åkerfeldt, A. (2020). Programming in primary education. *Proceedings of the 15th Workshop on Primary and Secondary Computing Education*, 1–10. `https://doi.org/10.1145/3421590.3421598`

OECD. (2013). What Makes Urban Schools Different? *PISA in Focus*, *28*, 1–4. `https://doi.org/10.1787/5k46l8w342jc-en`

OECD. (2023). *PISA 2022 Results (Volume I): The State of Learning and Equity in Education*. OECD Publishing. `https://doi.org/10.1787/53f23881-en`

Ohashi, Y. (2024). Online Survey on Changes of Teachers' Minds on Information and Communication Technology Use and Programming Education in Japanese Elementary Schools between 2017 and 2022. *Jour-*

nal of Information Processing, *32*, 93–102. https://doi.org/10.2197/ipsjjip.32.93

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

Piazza, A., & Mengual-Andrés, S. (2020). Computational thinking and coding in primary education: scientific productivity on SCOPUS. *Pixel-Bit, Revista de Medios y Educación*, *59*(59), 147–182. https://doi.org/10.12795/pixelbit.79769

Poulakis, E., & Politis, P. (2021). Computational Thinking Assessment: Literature Review. In *Research on E-Learning and ICT in Education* (pp. 111–128). Springer International Publishing. https://doi.org/10.1007/978-3-030-64363-8_7

Prieto, L. R., & Altmaier, E. M. (1994). The relationship of prior training and previous teaching experience to self-efficacy among graduate teaching assistants. *Research in Higher Education*, *35*(4), 481–497. https://doi.org/10.1007/BF02496384

Rao, T.S.S. & Bhagat, K.K. (2024). Computational thinking for the digital age: a systematic review of tools, pedagogical strategies, and assessment practices. *Education Technology Research and Development, 72*, 1893–1924. https://doi.org/10.1007/s11423-024-10364-y

Richardson, M., Abraham, C., & Bond, R. (2012). Psychological correlates of university students' academic performance: A systematic review and meta-analysis. *Psychological Bulletin*, *138*(2), 353–387. https://doi.org/10.1037/a0026838

Rodríguez, J., Marín, D., López, S., & Castro, M. M. (2023). Tecnología y Escuela Rural: Avances y Brechas. *REICE. Revista Iberoamericana Sobre Calidad, Eficacia y Cambio En Educación, 21*(3), 139–157. https://doi.org/10.15366/reice2023.21.3.008

Román-González, M. (2015). Computational thinking Test: Design guidelines and content validation. *7th Annual International Conference on Education and New Learning Technologies*. https://doi.org/10.13140/RG.2.1.4203.4329

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 Singapore. https://doi.org/10.1007/978-981-13-6528-7_6

Shah, V. (2019). CSpathshala. *Communications of the ACM*, *62*(11), 54–55. https://doi.org/10.1145/3343445

Simmonds, J., Gutierrez, F. J., Casanova, C., Sotomayor, C., & Hitschfeld, N. (2019). A Teacher Workshop for Introducing Computational Thinking in Rural and Vulnerable Environments. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 1143–1149. https://doi.org/10.1145/3287324.3287456

Smith, M. (2016). *Computer Science For All*. https://obamawhitehouse.archives.gov/blog/2016/01/30/computer-science-all

So, H.-J., Jong, M. S.-Y., & Liu, C.-C. (2020). Computational Thinking Education in the Asian Pacific Region. *The Asia-Pacific Education Researcher*, *29*(1), 1–8. https://doi.org/10.1007/s40299-019-00494-w

Sun, L., Hu, L., & Zhou, D. (2021a). Which way of design programming activities is more effective to promote K-12 students' computational thinking skills? A meta-analysis. *Journal of Computer Assisted Learning*, *37*(4), 1048–1062. https://doi.org/10.1111/JCAL.12545

Sun, L., Hu, L., & Zhou, D. (2021b). Single or combined? A study on programming to promote junior high school students' computational thinking skills. *Journal of Educational Computing Research, 60*(2), 283–321. https://doi.org/10.1177/07356331211035182

Sun, L., Hu, L., & Zhou, D. (2022). Programming attitudes predict computational thinking: Analysis of differences in gender and programming experience. *Computers & Education*, *181*, 104457. https://doi.org/10.1016/j.compedu.2022.104457

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

Weintrop, D., Rutstein, D., Bienkowski, M., & McGee, S. (2021). Assessment of Computational Thinking. In A. Yadav & U. Berthelsen (Eds.), *Computational Thinking in Education* (pp. 90–111). Routledge. https://doi.org/10.4324/9781003102991-6

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

Wu, H., Guo, Y., Yang, Y., Zhao, L., & Guo, C. (2021). A Meta-analysis of the Longitudinal Relationship Between Academic Self-Concept and Academic Achievement. *Educational Psychology Review*, *33*(4), 1749–1778. https://doi.org/10.1007/s10648-021-09600-1

Yang, Z., Blake-West, J., Yang, D., & Bers, M. (2025). The impact of a block-based visual programming curriculum: Untangling coding skills and computational thinking. *Learning and Instruction, 95*, 102041. https://doi.org/10.1016/j.learninstruc.2024.102041

Yeni, S., Grgurina, N., Saeli, M., Hermans, F., Tolboom, J., & Barendsen, E. (2024). Interdisciplinary Integration of Computational Thinking in K-12 Education: A Systematic Review. *Informatics in Education, 23*(1), 223–278. https://doi.org/10.15388/infedu.2024.08

Zander, L., Höhne, E., Harms, S., Pfost, M., & Hornsey, M. J. (2020). When Grades Are High but Self-Efficacy Is Low: Unpacking the Confidence Gap Between Girls and Boys in Mathematics. *Frontiers in Psychology*, *11*, 552355. https://doi.org/10.3389/fpsyg.2020.552355

Zapata-Cáceres, M., Marcelino, P., El-Hamamsy, L., & Martín-Barroso, E. (2024). A Bebras Computational Thinking (ABC-Thinking) program for primary school: Evaluation using the competent computational thinking test. *Education and Information Technologies, 29*(12), 14969–14998. https://doi.org/10.1007/s10639-023-12441-w

Zhong, B., Wang, Q., Chen, J., & Li, Y. (2016). An exploration of three-dimensional integrated assessment for computational thinking. *Journal of Educational Computing Research*, *53*(4), 562–590. https://doi.org/10.1177/0735633115608444

J. Del Olmo-Muñoz is an assistant professor in the Department of Mathematics at the University of Castilla-La Mancha, Spain. He holds degrees in Computer Science and Primary School Education, and completed a PhD on Computational Thinking Education in 2023. He is a member of the Spanish Society of Research in Mathematics Education (SEIEM) and of LabinTic – Laboratory for the Integration of ICT in the Classroom. His research interests include computational thinking, problem solving, and educational technology. ORCID: https://orcid.org/0000-0001-8754-0648

A. Jiménez Toledo is a researcher at the University of Castilla-La Mancha, Spain. She holds a degree in Primary School Education and a Master's degree in Bilingual Teaching and ICT for Kindergarten and Primary Education. Her research interests include educational technology, computational thinking, and teacher training in primary education. She is a member of LabinTic – Laboratory for the Integration of ICT in the Classroom.

S.Tirado-Olivares is an assistant professor at the University of Castilla-La Mancha, Spain. He holds a PhD focused on applied Learning Analytics in STEAM disciplines, completed in 2024. He also holds degrees in Primary Education and Educational Research and Innovation. His research interests include educational technology and learning analytics. He is co-responsible for LabinTic – Laboratory for the Integration of ICT in the Classroom. ORCID: `https://orcid.org/0000-0002-8557-5115`

J.A. González-Calero is an associate professor in the Department of Mathematics at the University of Castilla-La Mancha, Spain. He holds a PhD in Educational Mathematics and master's degrees in Secondary Education and Mathematics Didactics. His research focuses on educational technology, problem solving, and interactive learning environments in mathematics education. He is a member of SEIEM and co-responsible for LabinTic – Laboratory for the Integration of ICT in the Classroom. ORCID: `https://orcid.org/0000-0003-0842-8151`

R. Cózar-Gutiérrez is an associate professor at the Faculty of Education of the University of Castilla-La Mancha, Spain, where he coordinates the Master's Program in Educational Research and Innovation. His research interests include social science education, teacher training, and educational technology. He is also a member of LabinTic – Laboratory for the Integration of ICT in the Classroom. ORCID: `https://orcid.org/0000-0001-8255-6376`