Technology, Knowledge and Learning (2025) 30:63–118
https://doi.org/10.1007/s10758-024-09765-y

ORIGINAL RESEARCH

Co-design Pedagogy for Computational Thinking Education in K-12: A Systematic Literature Review

Amos Oyelere Sunday[1] · **Friday Joseph Agbo**[2] · **Jarkko Suhonen**[1]

Accepted: 11 July 2024 / Published online: 6 August 2024

Abstract
The recent popularity of computational thinking (CT) and the desire to apply CT in our daily lives have prompted the need for a successful pedagogical technique for learning CT in K-12 education. The application of co-design pedagogical techniques has the potential to improve students' CT learning through knowledge sharing and the creation of ideas to solve problems and develop an artifact. However, there is a limited understanding of how co-design pedagogical techniques have been explored to foster CT learning, which could hamper the successful use of co-design as a pragmatic teaching approach. This study examined the ways in which co-design pedagogical techniques have been applied in CT education by implementing a systematic literature review (PRISMA protocol) to document the review analysis. A total of 26 articles that met the inclusion criteria for this study were reviewed. Findings in this study revealed that workshops are the most utilized co-design learning setting and, as expected, the collaborative technique is the co-design pedagogical technique most frequently adopted for implementing CT in K-12 education. NetLogo is the most frequently used co-design tool for teaching and learning CT in K-12 education, and an interdependence exists between NetLogo and the Common Online Data Analysis Platform. Co-design also helps teachers develop the ability to use co-design pedagogical techniques to learn, create content, and integrate CT into their various subjects. This study contributes to practical knowledge by unraveling and advocating the use of dialogical, prompting, framing, and game-based techniques as co-design pedagogical techniques for K-12 teachers and also helps teachers identify useful co-design tools for learning CT.

Keywords Computational thinking · Co-design pedagogy · Literature · K-12 education

✉ Amos Oyelere Sunday
amossun@uef.fi

Friday Joseph Agbo
fjagbo@willamette.edu

Jarkko Suhonen
jarkko.suhonen@uef.fi

[1] School of Computing, University of Eastern Finland, Joensuu, Finland

[2] School of Computing and Information Sciences, Willamette University, Salem, OR 97301, USA

1 Introduction

Following Wing's (2006) introduction of computational thinking (CT) concepts. There is a growing body of literature focusing on different ways of integrating CT into formal and informal education. For instance, many researchers (Kong, 2016; Yadav et al., 2016) have identified the benefits of teaching CT to young children and report the positive results of teaching CT in the classroom. Similarly, in higher institutions, CT has proven to be beneficial for preparing computer science students for programming courses (Agbo et al., 2019; Kin et al., 2021; Piedade et al., 2020). Recently, there has been renewed interest in the integration of CT into the primary and secondary curricula (Kelly et al., 2019; Waterman et al., 2020; Yang et al., 2021). K-12 teacher training programs, usually referred to as professional development programs or teacher development programs, have been organized to instruct teachers on ways they can use to introduce the CT concept, develop CT materials, create a CT curriculum, and integrate CT into their various classrooms (Ketelhut et al., 2020; Kong et al., 2020; Monjelat & Lantz-Andersson, 2020). Furthermore, a teachers' training model called PRADA has been built for the purpose of training teachers. PRADA stands for "Pattern Recognition, Abstraction, Decomposition, and Algorithms." The PRADA model has been developed to train non-computing teachers to comprehend CT core concepts and aid teachers' practical understanding of CT concepts. PRADA also improves teachers' capabilities to integrate CT into their existing teaching materials and enhances their self-confidence (Dong et al., 2019; Sunday, 2023).

The term "co-design" has been defined as the facilitation of a team-oriented process in which researchers, teachers, and developers collaboratively work hand-in-hand to define roles for building educative inventions, conceptualize a design, build a prototype, analyze the prototype, and re-design the prototype to meet the required educational needs (Roschelle et al., 2006; Sunday, 2023). This definition highlights the stakeholders and processes involved in using co-design for educational innovation creation. Equally, co-design is considered to be an important facet of the educational innovation-making process (Riikonen et al., 2018). Nonetheless, co-design as a pedagogical technique depends largely on the teacher's participation in the process of designing pedagogical inventions that involve technology as a major aid for practice, which results in the creation of curriculum materials in schools (Penuel et al., 2007; Roschelle et al., 2006).

Different pedagogical techniques have been adopted for teaching CT at various educational levels, including game-based techniques (Agbo et al., 2021a, 2021b; Kin et al., 2021) and programming (Lye & Koh, 2014; Tikva & Tambouris, 2021). However, the literature provides no clear understanding of how co-design pedagogy has been applied as a pedagogical approach to facilitate the teaching and learning of CT in K-12 education. Thus, this study aims to address this research gap. To our knowledge, this study is one of the first (SLR) studies to investigate co-design pedagogical techniques in CT education and reveal the state of the art of co-design pedagogy in CT K-12 education. The aim of this study is to investigate the literature to identify and understand how co-design pedagogical techniques have been applied in teaching and learning CT. Therefore, the objectives of this study include (1) identifying different co-design pedagogical techniques, learning activities, and learning settings used in teaching and learning CT in K-12 education; (2) classifying the different co-design tools and technologies based on their features and identifying the co-design CT learning and teaching mode in K-12 education; and (3) clarifying how studies that use co-design pedagogical techniques have contributed to CT education at the K-12 levels. Specifically, this study adds to existing knowledge and promotes the application of co-design pedagogical techniques to foster the learning of CT by revealing the implementation of co-design pedagogical approaches, the associated learning activities, and the tools or resources used in co-designing for teaching and learning CT. In addition, this study reveals the impacts of co-design pedagogical approaches in the literature in terms of tools or resource development, curriculum design, building innovation or creativity, building ideas or concepts, and applying programming to learning CT through co-design pedagogy. The following research questions were formulated to achieve the research aims.

RQ1: How has the co-design pedagogy been implemented for computational thinking education at K-12 levels?
RQ2: What tools and technologies have been used for co-design pedagogy in computational thinking education at K-12 levels?
RQ3: How have studies on co-design pedagogy contributed to computational thinking education at K-12 levels?

2 Theoretical Background

2.1 Computational Thinking Education

2.1.1 Definition of Computational Thinking

Recently, CT has been a major topic of discussion among researchers in K-12 education and higher institutions of learning (Angeli & Giannakos., 2020; Agbo et al., 2022). Various attempts have been made to define CT; however, no specific definition has been universally accepted. Wing (2006) defined CT by connecting its fundamental concepts to knowledge in computer science and a process that entails providing a solution to problems, developing a system, and comprehending human behavior. Agbo et al.,(2019) describe CT features as (a) expressing problems in a way that a computer or other tools can be used to solve the problems; (b) logically systematizing and analyzing the data; (c) formulating data through abstractions in simulation and model form; (d) using algorithmic ideas to automate a problem's solution; (e) discovering, evaluating, and applying potential solutions with the aim of obtaining the most efficient blend of phases and resources; and (f) organizing and applying the problem-solving steps to problems in other fields. In addition, CT is considered to be the fundamental background of computer science, which means it represents the main scientific thinking of the digital age (Agbo et al., (2019); Shute et al., 2017;. Therefore, CT can be assumed to be a required skill for providing an efficient solution to real-world problems. In addition, several CT concepts have been identified including algorithmic thinking, recursive thinking, abstraction, pattern recognition, decomposition, and problem-solving.

Apiola & Sutinen presented CT in terms of three forms of thinking in engineering education: engineering thinking, scientific thinking, and mathematical thinking. The main target of engineering thinking is innovative production, with a focus on changing the world through building and implementing inventive artifacts that solve a particular problem and consider reliability, utility, and usability (Apiola & Sutinen, 2021). Scientific thinking focuses on discoveries about the world that involve working with programs, numeric techniques, and models to create theoretical and abstract applications, such as computational model phenomena and artificial intelligence, to understand the world's complexity and solve global problems such as climate. Mathematical thinking uses algorithms and theoretical structures to redefine, extend, and solve complex problems through data structures and algorithms and create a concrete, theoretical structure with accuracy and coherency (Apiola & Sutinen, 2021). Additionally, CT has demonstrated strengths in providing solutions to diverse problems faced by humans (Li, 2016). CT has shown its potential in advancing human–computer interaction and association perceptions (Li, 2016; Shute et al., 2017). Further, CT is not limited to programming or coding capabilities but rather involves the ability to think in various ways for problem-solving (Wing, 2006). Several studies have provided different models for CT, for example, Denning and Tedre (2021) presented the old or beginner CT model, which focused on stimulating and arousing students' interest in computing. The beginner's CT problem-solving definition involves problem abstraction, decompositions, or recursion, which could be the first techniques utilized in solving a problem. These techniques can be followed by creating a step-by-step algorithm design for the problem-solving, which may lead to programming or another form of design depending on the kind of problem to be solved, and finally obtaining the real solution.

However, CT is evolving and moving towards new dimensions by embracing problem-solving concepts for professional circles to proffer solutions to different real-world problems (Denning & Tedre, 2021). Shin et al. (2021) recently presented five new steps for a CT problem-solving model. These include problem decomposition for a computational solution (CT1); using algorithmic thinking to create computational artifacts (CT2); data generation, organization, and interpretation (CT3); evaluation, testing, and debugging (CT4); and creating iteration and refinements (CT5). Nevertheless, this study considers the combination of the beginner CT model (Denning & Tedre, 2021) and CT simulation model (Shin et al., 2021) to provide a renewed CT framework for efficient problem-solving across the board. This framework includes solving the problem by first applying either decomposition, abstraction, or recursive thinking depending on the problem at hand; followed by applying recursive thinking where necessary and algorithmic thinking; then data generation, organization, or interpretation; followed by evaluation, testing, and debugging; which are finally followed by iterating and refining to provide an efficient solution (See Fig. 1).

As presented in Fig. 1, CT is gradually moving towards data-driven and machine learning (ML) approaches, as several ML and data-driven concepts have gradually been incorporated into CT (Tedre et al., 2021). The application of CT has revealed its diverse enhanced dimensions (Agbo et al., 2021a, b; García-Peñalvo et al., 2016; Jormanainen & Tukiainen, 2020). Similarly, the professional cycle are specialization fields, which include the field of sciences, mathematics, design, and engineering, where concepts such as automation and machine controls, software development and design thinking, and systemic thinking have also been proposed as vital concepts for CT (Apiola & Sutinen, 2021; Denning & Tedre, 2021). These new concepts incorporated into CT have primarily been applied for problem-solving in their respective professional fields. Moreover, these concepts have also been seen as relevant as part of the CT framework due to their proficiency and applicability in solving different complex problems. In addition, these concepts can help students recognize the crucial paths to follow to become a computing professional (Denning & Tedre, 2021; Tedre et al., 2021).

Several literature reviews on CT have been conducted to identify and provide the reader with an understanding of state-of-the-art CT in relation to the respective research topic (Palmatier et al., 2018). Bati (2022) conducted a systematic review of the literature on CT and programming in early childhood education. In his study, Bati identified that one of the most important factors for early childhood learning and CT is age; the application of unplugged activities was more efficient in improving children's programming and CT skills, thus, highlighting the power of tangible experiences for the students. The study also found no evidence of gender performance differences in CT (Bati, 2022). Similarly,

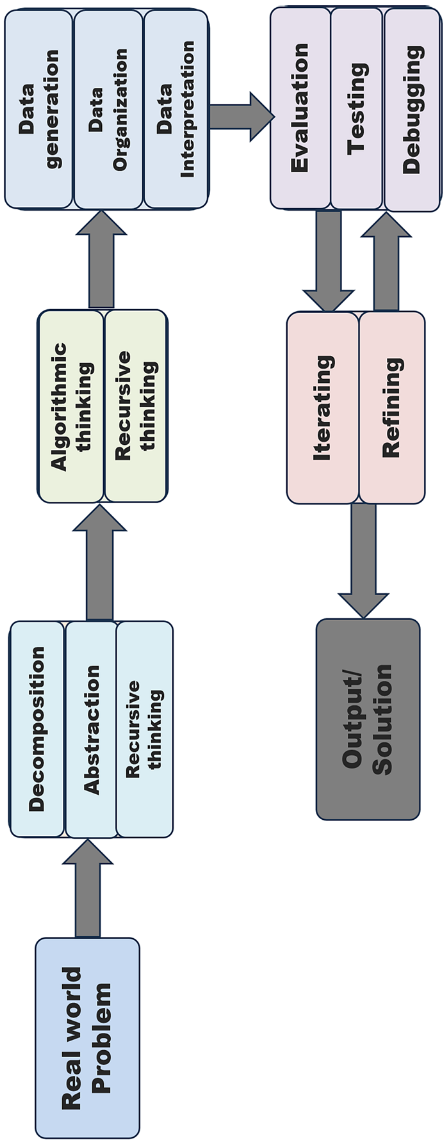


Fig. 1 Computational thinking problem-solving framework

Hsu et al. (2018) conducted a literature review to understand the application and development of CT in education that spanned ages, subjects, learning strategies, programming languages, and groups. The results show that the major CT learning strategies adopted, as identified in the study, were problem- and project-based strategies. A formal course setting was the most frequently adopted type of setting for learning CT. The study also revealed that the researchers and educators employed different learning strategies to aid and enhance student learning abilities and performance through diverse CT activities. Further, programming subjects were identified as the subjects that most frequently applied CT, followed by computer science subjects and mathematics. Teaching instruments for CT activities were frequently used in programming design courses. Scratch was the most frequently used programming language application adopted for learning CT (Hsu et al., 2018). Further, Papadakis also carried out an SLR to investigate the educational value of using coding applications for building young children's CT and computational fluency. The outcome of the study revealed that the evaluated applications supported and positively affected children's CT skill development, while none of the applications conclusively aided students' computational fluency. However, the ScratchJr application, which uses a "sandbox" approach (is an approach that provides a better structured programming space and style in Scratch) aided children in expressing themselves (Papadakis, 2021).

However, the implementation of CT into STEM education has encountered several barriers (Peel et al., 2023). These barriers include computing and CT not being included in pre-service teacher courses, teachers lacking adequate backgrounds in computer science, and limited resources for integrating CT, which result in a low sense of confidence in teaching and integrating CT in STEM education (Aljowaed & Alebaikan, 2018; Peel et al., 2023; Penuel et al., 2011; Sands et al., 2018; Yadav et al., 2016; Yoon et al., 2020). A recent study on practicing teachers of K-12 STEM subjects revealed teachers' perceptions of the difficulties facing CT and its implementation in K-12. These difficulties included assertions by teachers such as "I don't know how to align CT to my subject," "I don't have much understanding of CT," "my students are not academically ready for CT-infused subjects," "I have no access to the relevant technology," and "administrators do not support CT" (Yoon et al., 2020). Having identified teachers' distress and impediments in implementing CT in K-12 education (Yoon et al., 2020)**,** it is important to develop approaches for integrating CT that will attract the support of teachers and aid their confidence in teaching and integrating CT in their STEM classes (Peel et al., 2023).

2.1.2 Tools, Pedagogies, and Technologies for Computational Thinking Education

Several studies have adopted different tools for teaching and learning CT, including Code Master, Dr. Scratch, and LOGO (Nombela et al., 2018; Weng, et al., 2024; Kong et al., 2020). Nombela et al. (2018) presented Code Master as a tool that can be used by students to assess their learning performance in a project and receive instant feedback. Equally, Code Master can be used by teachers to evaluate and grade all their student's class projects at the same time in a comprehensive examination (Nombela al., 2018). Kong et al. (2020) used CT tools (Scratch) with the TPACK framework to train K-12 in-service teachers on programming basics for building CT skills. The results show that the participants' capacities for CT and its concepts were enhanced by these tools. Studies have revealed that LEGO can be used to advance student technology skills through CT (Weng et al., 2024). In addition, Weng et al. (2024) demonstrated how university students can use robotics kits (LEGO) to build their CT skills. Vallance and Towndrow (2016) also developed students'

CT skills by training students to develop a 3D virtual space (LEGO). Their study demonstrates how Japanese undergraduate students studied the feasibility and practicalities of using programmable robots to react to the ongoing Fukushima Daiichi nuclear power plant breakdown (in 2011) with CT. The results revealed that the students successfully identified problem areas for themselves and were also able to proffer solutions using IT, both in real-time and locally through collaboration with other students located in the United Kingdom.

The development and implementation of CT requires the application of different pedagogies. One form of pedagogy used for CT is unplugged pedagogies. Several studies have adopted unplugged pedagogies to implement the teaching and learning of CT. Bell and Vahrenhold (2018) revealed how unplugged pedagogy supports CT, which includes aiding and complementing computer programming in CT skill development; aiding teachers in learning CT; and evaluating CT in the form of Bebras puzzles. Huang and Looi (2021) stated that unplugged pedagogies can successfully deliver CT to K-12 students in outreach settings. The unplugged approaches can potentially provide the same instruction as formal education. Huang and Looi (2021) also conducted a literature review study, which revealed that unplugged pedagogies can support CT development by providing an easy way to implement teaching, learning, open exploration, and inquiry-based learning. Similarly, Kotsopoulos et al. (2017) revealed four pedagogies that can be used for learning CT, which include tinkering, making, and unplugged.

2.2 The Co-design Pedagogy in Computing Education

2.2.1 Definition of Co-design Pedagogy

Co-design pedagogy in computing education refers to the application of collaborative design processes and practices in the field of computing education. Schmidt et al. (2016) and Seitamaa et al. (2012) adopted co-design pedagogies to inspire students' engagement with problem-solving activities that relate closely to expert communities' activities. The activities were focused on students sharing ideas and discussion. Students were assigned to teams and subgroups with roles in order to provide solutions to a given task by sharing ideas from their knowledge, engaging in discussion, making necessary decisions together, and bearing responsibility for those decisions (Vartiainen et al., 2020; Seitamaa et al., 2010; Toivonen et al., 2020). Students were also encouraged to work with professionals to experience professional practices (Radu et al., 2023; Sanusi et al., 2023a, 2023b; Park & Schallert, 2019; Seitamaa et al., 2012). This kind of collaboration, which involves guidance or cognitive apprenticeship, is extremely vital. In such collaborations, professionals offer metacognitive help for students to provide solutions to problems or complete a task that they do not independently have the skills to solve (Collins et al., 2018; Matsuo & Tsukube, 2020).

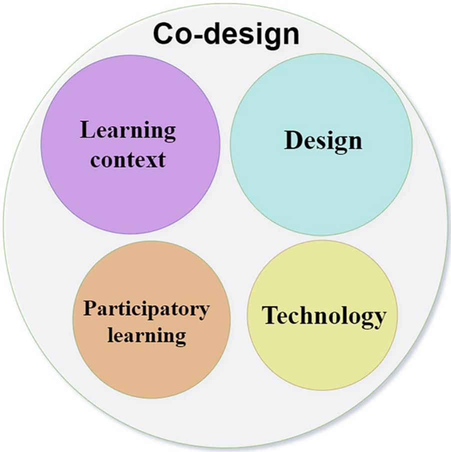
Co-design is a design string in which the main activities are founded on the power of the collaborative innovation of the designers with others not experienced in design and collaboration in the design structure processes, which range from the pre-design stage (problem scoping) to the post-design stage (after using the design object: Olesen et al., 2022; Sanders & Stappers, 2008, 2014). Similarly, co-design learning tools and technology represent platforms, applications, or software adopted in facilitating collaborative design procedures, which permit several stakeholders to participate in the design process (Olesen et al., 2022). In conducting a co-design study, several tools or technologies are adopted in co-designing to learn or teach several topics in different fields applicable to diverse researchers. Scratch, NetLogo, NetTango, and CODAP are examples of tools or technology used in co-designing for learning programming, algorithms, and chemistry (Giner Sanz et al., 2019; Real et al., 2020; Sathasivam & Fen, 2013; WaiShiang et al., 2017).

The term "design-oriented pedagogy" has been used to refer to co-design pedagogy (Vartiainen et al., 2012). Co-design pedagogy techniques serve an important role in facilitating collaboration and learning (Seitamaa et al., 2012; Vartiainen, 2022). Co-design pedagogy is an idea-creation process that elicits students' creative abilities by playing with different materials and interacting with one another through external and conceptual representation (Papert, 1980; Resnick, 2017). Through co-design pedagogy, the students gain more understanding of abstracted and complicated content by refining their ideas in association with the external representation (Kelter et al., 2021; Sanusi et al., 2023a, b). Similarly, Calvo and Sclater (2021) have emphasized that the co-design pedagogical process stresses the need for the authenticity of ideas and idea-focused activities. According to Sanders and Stappers (2008), co-design is a new landscape that promotes participatory design, while the co-design pedagogical process encourages collaboration between groups, teams, and peers. These are ideally heterogeneous and multi-age learner groups with diverse experiences who are faced with idea-focused activities in order to achieve a unique output. These processes create an avenue that allows students to learn from each other through participation. In addition, the co-design pedagogical solution empowers students to become creators, innovators, designers, and developers of artifacts and knowledge. Additionally, other techniques can be applied in conducting a co-design study, such as reflective (Wu et al., 2020), participatory (Alhumaidan, 2017; Ssozi-Mugarura et al., 2016; Wu et al., 2022), constructionist (Bain et al., 2020; Kelter et al., 2021), and game-based techniques (Turchi et al., 2019; Weintrop et al., 2016).

Furthermore, the co-design pedagogy relates to one of the CT framings termed "situated CT," as described by Kafai et al. (2020), in terms of collaboration to build knowledge through design. Kafai et al. (2020) present "situated CT" as a CT framing that involves building computational fluency in students via developing the programming of distributable digital artifacts. This idea is derived from connected learning and constructionist theories. Connected learning is a learning framework that involves support from peers through the social sharing of ideas and knowledge with feedback to empower an interest in gaining knowledge through academic design (Ito et al., 2013). The constructionist approach involves students learning by constructing knowledge or learning by creating (Papert & Harel, 1991). This implies that the situated computational thinking approach involves the development or the creation of knowledge through the sharing of ideas between peers to fuel personal interest in achieving educational goals and the development of artifacts. The situated approach can be achieved in a digital space or over social media. In situated CT, CT is referred to as a vehicle for interpersonal connection and expressions of interest, knowledge, and ideas that could lead to the creation of knowledge and the development of an artifact for academic and learning attainment.

Co-design learning entails involvement and collaboration between various educational stakeholders, including students, tutors, administrators, researchers, and other educational stakeholders, during the development, design, and implementation of a learning experience (Bovill, 2020; Durall, et al., 2019). In co-design learning, the collaboration that exists during the learning experience is an engagement that can occur in the form of partnership, mentorship, or apprenticeship (Bovill et al., 2016; Bovill, 2020; Sanusi, & Oyelere, 2020). The co-design pedagogical framework, as illustrated in Fig. 2, is divided into four pillars, which include the learning context, participatory learning, technology, and design. The major emphasis is on the learning context and design (which is why they are larger in size

Fig. 2 Co-design pedagogical framework (Adapted from Vartiainen et al., 2012)



in Fig. 2); however, the remaining two pedestals are also relevant and significant contributors to the achievement of the result. The design represents the outcome or result of all the activities accomplished by the collaboration that takes place within the three pedestals. In other words, design is the function of all the activities that take place within the co-design pedagogical framework. Participatory learning is an important learning concept that aids student learning through active participation and collaboration. Technology refers to the internet, which serves as the technological infrastructure that aids the learning process. The learning context refers to the situation surrounding the learning in which the student learning output is built, with the tutor supporting the students in improving their knowledge to achieve the learning goal (Collentine et al., 2004). The learning goal in this context is building knowledge through design.

Participatory learning involves students taking part in a learning process, research, and community development that highlights the need for personal learning. The technology represents the internet, which provides a technological platform to aid collaborative learning and serves as a foundation for personal learning. Students' own technologies such as mobile phones, tablets, and laptops provide tools or resources that aid learning across diverse contexts and support the collection of diverse empirical data. Social media offers platforms to enable student development, organization, knowledge sharing, and collaboration within and outside academic communities (Vartiainen, 2014). Moreover, co-design can be used as a pedagogical foundation for learning in academic institutions (Vartiainen et al., 2012). It represents the power behind the social initiatives of designed software and products (for example, the Linux phenomenon and Open Source; Vartiainen, 2014). Co-design promotes participation in community activities that encourages the implementation of members' practices in municipal development (Moser & Korstjens, 2022). Co-design stresses the need for socialized learning and promotes participants' successful participation in the activities (Schaper & Pares, 2021; Sunday, 2023). Students can be referred to as designers, developers, creators, architects, scientists, and makers based on the context of the co-design pedagogical technique. The context of the study is one of the major determinants in the co-design pedagogy (Kyza & Agesilaou, 2022; Sanusi & Oyelere, 2020).

2.2.2 Co-design Pedagogy Solutions

The co-design pedagogy has been used to provide solutions to diverse educational challenges. Studies have revealed that there are a handful of challenges to be solved in science education (Aksela, 2019). These challenges have led to the application of a co-design approach in science education to provide solutions to the numerous challenges existing in this field. The study shows how co-design pedagogy was applied with many stakeholders, consisting of students, teachers, scientists, industry specialists, and teacher educators, to address the 21st-century challenges of the science education LUMA2020 program, which occurred in Finland. A co-design approach was also applied with design-based research to facilitate the study of inventive pedagogy through a digital learning platform and a face-to-face communication workshop for social interaction and inventive thinking. The co-design approach was used to develop a framework through the Edelson models. The results revealed that the co-design approach helped in making decisions, advancing teachers' professional development, and providing a well-organized ecosystem for science education, including the development and improvement of the school curriculum (Aksela, 2019). Similarly, a co-design pedagogical approach was adopted in a participatory design project in the UK to ascertain how student-focused study can aid the implementation of a technology-aided classroom plan (Nicholson et al., 2022). The study presented co-teaching as a unique form of co-design practice used for developing educational technologies. In this co-design process, teachers and researchers engage in a study from the beginning (ideation level) to the end (evaluation level), during the study they co-teach in the classroom with the designed tool for total integration into classroom practice. The results of the study demonstrated how educational technology-centered design researchers aid the development and implementation of technologies in the K-12 classroom. The result also include how the co-design pedagogy provides support for long-term implementation, thereby creating experimentation, pedagogical alignment with technology design, and highlighting the value of co-teaching for researchers (Nicholson et al., 2022).

Furthermore, a co-design pedagogy was used to develop an embedded framework for introducing hardware and software courses to university students (Biswas & Johnson, 2015). The study used a co-design pedagogy with project-based learning for teaching and learning hardware and software in 21st-century computer engineering undergraduate courses (Biswas & Johnson, 2015). Co-design also promotes profound learning achievement and enhances student competence in programming and digital systems through the active participation of students that the co-design pedagogy encourages (Matutino et al., 2020). Consequently, the study results revealed that the students were able to consolidate their hardware and software skills and aggregate both perspectives (Matutino et al., 2020). Moser (2016) revealed that co-design can serve as an agent of transformation that can be used to transform a variety of systems. Additionally, Vázquez et al. (2021) presented a study on the application of co-design for teaching high-level programming to reduce the complexity of robot programming for novice learners. The co-design process and functionality were integrated into the robot to aid users in the automatic design and implementation of robots and their controllers in the Robot Operating System. The co-designed robot contains the programming framework of an intelligent system. The co-design programming framework was converted into visual programming blocks to permit inexpert users to easily learn programming through a visual programming language. Bravo-Palacios (2020) also presented a study that used co-design to develop a versatile framework for solving problems in the face of uncertainty using stochastic programming. The results of the study revealed that the system was improved in terms of versatility, energy cost, and task completion times throughout multiple scenarios. Severance et al. (2016) adopted co-design for developing curriculum materials for science education. In another study, students engaged in a co-design study to develop curricula for public administration programs (Elliott et al., 2021). Van Brummelen and Lin (2020) also adopted co-design in developing an AI K-12 curriculum.

2.3 Relevance of Co-design Pedagogy for Computational Thinking Education

The co-design pedagogy has been relevant for different fields of study and has been successfully used to facilitate teaching and learning with diverse methods and tools. In addition, co-design has been explored as a pedagogical method and recently adopted to teach and learn different topics, subjects, and ideas, and promote the development of tools or resources for teaching and learning (Dickes et al., 2020; Peel et al., 2020; Zeregal et al., 2021). The co-design process has also been adopted to develop and evaluate materials to aid students' progress (Higgins et al., 2019; Penuel et al., 2007). However, the co-design process depends on teachers' active participation in the development of educational inventions for enhancing teaching practices (Higgins et al., 2019; Penuel et al., 2007). Several co-design studies have been conducted to train or advance teachers' knowledge to facilitate effective knowledge transfer in the classroom (Wu et al., 2020). In this regard, workshops have been one of the major ways of facilitating co-design studies to train teachers. The co-design workshops concentrate on either the pedagogy, materials, or tool development to allow teachers to gain more learning experience in implementing the new ideas in the classrooms (Bain et al., 2020; Higgins et al., 2019; Wu et al., 2020). In addition, a recent study has shown that co-design can also be conducted through an online environment. According to the Agbo et al., (2021a, 2021b), an online co-design process was conducted with computer science students at a Nigerian university to teach CT concepts. The results indicated that this process could potentially enhance students' learning experiences. Similarly, co-design has been applied as a pedagogical technique in teaching and learning CT (Wu et al., 2020) in an attempt to integrate CT into K-12 STEM education.

Further, Moser (2016) listed the benefits of co-design, which reveal the importance of its application to solving both educational and non-educational problems. These include improving understandings of topics and advancements in knowledge, collaborative research, communication, and interpersonal relationships, grounding in local realities, and improved researcher reflectiveness. The output produced cut across various audiences, thus, producing practical results. These benefits further revealed the capacity, functionality, and usefulness of co-design in profiling solutions for every sector including CT education.

3 Methodology

This literature review aims to identify how the co-design pedagogy has been applied in CT education. Therefore, an SLR was designed to obtain studies that adopted a co-design pedagogy in teaching CT in K-12 education, following the guiding principles of Kitchenham and Charters (2007). This review study encompasses three major phases, as presented in Fig. 3, and the articles for the SLR were retrieved from five different databases.

Five different databases were selected for this study (See Table 1). The rationale for selecting these five databases was that they are popular and index a large number of

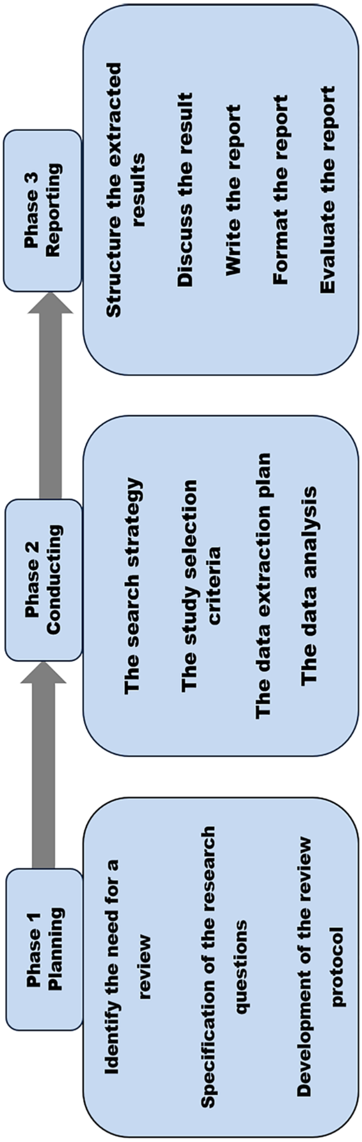


Fig. 3 Systematic literature review process (Kitchenham & Charters, 2007)

Table 1 Search structure

Academic databases	Search design
Scopus	Title, abstract, and keywords (TITLE-ABS-KEY)
Web of Science (WOS)	All fields
IEEE Xplore	All fields
Springer Link	All fields
ACM	All fields
Google scholar	

publications focused on the fields of design-oriented computing education with a high number of citations. The outcomes of the searches from the five databases were downloaded and an Excel worksheet was used for computing, manually sorting the duplicate articles, and organizing the collected data. The study followed the Preferred Reporting Items for Systematic Literature Review and Meta-Analysis (PRISMA), which is a widely used framework for systematic reviews in the social, education, and health sciences (Elmoazen et al., 2022). The selected papers include K-12 teachers training programs and teachers' professional development programs. The literature revealed that teacher training is one of the major means of preparing and improving K-12 teachers' knowledge of pedagogical skills, curriculum development and implementation, integration of technology, and classroom management (Coimbra 2020; Zhao & Liu, 2022). Teacher training provides K-12 teachers with comprehensive content knowledge and improves their expertise in teaching, application of new technology, and management of their classrooms (Green & Anid, 2013; Greenberg & Walsh, 2012). Hence, without studying the teacher training literature, we cannot successfully answer all the research questions in this study. This shows that the K-12 teacher training programs are very important to the implementation of K-12 education, as revealed in the articles we reviewed.

3.1 Planning the Review

3.1.1 Identifying the Need for a Review

As presented in phase 1 of Fig. 3, the first step was to strategically examine and identify articles that meet the review criteria as planned. The review plan begins with recognizing the major articles connected to CT and co-design pedagogical techniques relevant to K-12 education. This literature review is relevant to the research community and K-12 education stakeholders, as it sheds more light on the application and implementation of co-design pedagogical techniques for CT learning in K-12 education, the kind of tools used in applying those techniques to CT learning in K-12 education, and the contribution of the pedagogical techniques to teaching and learning CT in K-12 education. These identification processes preceded the other processes involved in the PRISMA protocol for SLR. This process promotes the proper identification of articles and facilitates the selection process.

3.1.2 Specification of the Research Questions

This study searched five databases without any specific time range for previous research articles. The research questions above were framed to classify the articles for data extraction and evaluation:

- The first research question investigates the selected literature by concentrating on identifying co-design approaches and how they have been implemented in teaching and learning CT in the literature. The first research question focuses on three aspects: pedagogy, learning settings, and activities.
- The second research question identifies and examines different kinds of tools and resources used in the selected literature. The second research question addresses three areas: tools or resources, tool features, and study mode (learning and teaching mode).
- The third research question identifies and analyzes the impacts of each selected study. The third research question is used to measure the use or contribution of the co-design approaches and tools utilized in the selected articles.

3.1.3 Development of the Review Protocol

The foundation for conducting a systematic review is provided by the review protocol (Oyelere et al., 2020). Specifying and divulging the approaches deployed in this study helps limit unintentional errors. To this end, this study adopted a review approach founded on the review protocol presented by Kitchenham and Charters (2007). In the course of planning this review, informal and formal searches were utilized to identify literature and gather pertinent information to address the research questions. The informal search involved identifying articles through sources other than those specified by the SLR protocols, while the formal search involves articles identified through those same protocols (Niazi, 2015).

3.2 Conducting the Review

This phase elucidates the procedures followed in conducting the review, as presented by Kitchenham and Charters (2007), which includes the search strategy, the literature selection criteria, the quality assessment of the study, the plan for extracting the data, and the analysis of the data.

3.2.1 The Search Keywords and Strategies

This study formulated the research strategy based on expounding the research questions. To minimize the number of irrelevant articles, the keywords were selected based on the research topic, study objective, and research questions.

The search keywords were formulated to gather literature about co-design and CT without any restrictions on the search criteria. The keyword search was focused on all metadata or fields of the articles in each of the databases. Vital information is contained in the metadata of an article, for instance, the keywords, abstract, and article title Agbo et al., (2019). Table 1 shows the structure of the search, and Table 2 presents the executed protocol for each respective database. The search keywords include "computational thinking" and "co-design."

Table 2 The executed protocol

Database	Search string	Results
ACM	*[All: computational thinking] AND [All: co-design*]*	386
IEEE Xplore	("All Metadata": computational thinking) AND ("All Metadata": co-design*)	5
Web of science	ALL = (computational thinking AND co-design*)	16
SpringerLink	'Computational thinking AND co-design*'	2654
Scopus	(TITLE-ABS-KEY (computational AND thinking) AND TITLE ABS-KEY (co-design*))	33

Table 1 presents the search design and academic databases in which each database search was carried out. In four of the databases, ACM, Web of Science, SpringerLink, and IEEE Xplore, all fields were searched without any restrictions to obtain all the relevant articles from the databases. However, the search in Scopus was not performed using "ALL" but rather on the abstract, title, and keywords because searching "ALL" in this database usually results in having more irrelevant results (Sanusi 2022). Meanwhile, Table 2 contains detailed records of the organized structure adopted in the literature search for each database. As presented, a single Boolean operator "AND" was utilized to link the search terms, which identify the alternatives and synonyms of the words. The results of the retrieved data from the five electronic resources were reduced using the exclusion and inclusion criteria that were formulated to meet the research objectives. The Web of Science and Scopus electronic databases provide relevant literature (Sanusi et al., 2021) as well as a functional robust search platform. In addition, relevant article reference lists were searched manually to retrieve more articles.

3.2.2 The Article Selection Criteria

Figure 4 shows the process involved in selecting the appropriate articles for this research, which includes the exclusion and inclusion criteria. The whole text of the selected papers was read to ascertain whether they successfully met the inclusion and exclusion criteria.

3.2.2.1 Inclusion Criteria The inclusion criteria included the following:

- Articles published in the English language.
- Articles focused on CT including poster papers.
- Articles published in peer reviewed journals or conferences.
- Papers that concentrate on co-design for teaching CT.
- Papers aimed at teaching and learning CT via co-design.
- Academic literature that discusses co-design tools, resources, and methods used for learning and teaching CT.

3.2.2.2 Exclusion Criteria The exclusion criteria consisted of the following:

- Literature not written in the English language was removed.
- Non-peer reviewed literature (books, audio, files, PPT, etc.) was not included.
- Articles that did not address CT were excluded.

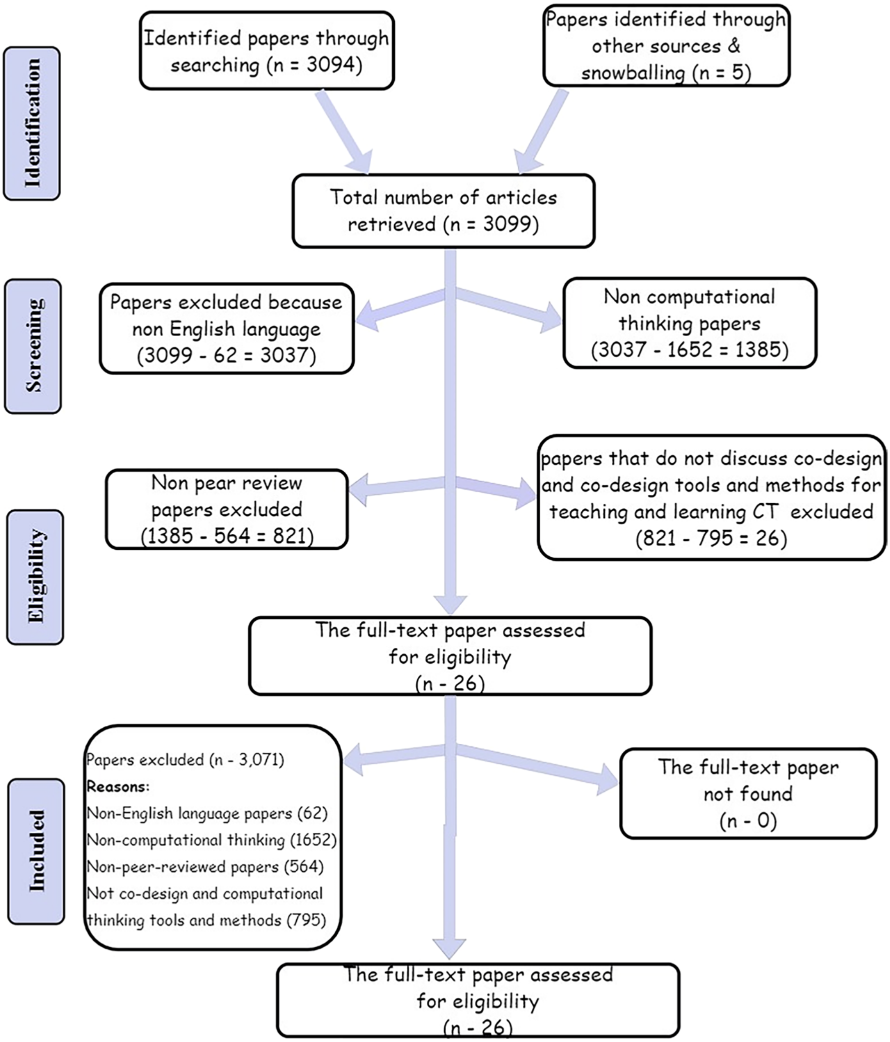


Fig. 4 Flow diagram of the review process (Adapted from Oyelere et al., 2020)

- Articles that did not include the keywords (in the metadata) were not included.
- Articles that did not address teaching and learning CT were not included.
- Articles that did not discuss different co-design tools, resources, and methods used for learning and teaching CT were excluded.

The inclusion and exclusion criteria determined the selection of the articles. In this study, a total of 26 articles met the inclusion criteria and were evaluated and analyzed. The 26 articles utilized in this study, which are listed in Appendix 1, were labeled S1 to S26, where S stands for "serial number."

3.3 The Data Extraction and Analysis

This study adopted an inductive analysis approach to derive the concepts from the data (Elo & Kyngäs, 2008; Sanusi et al., 2023a, 2023b). A coding scheme was built to guide the process of extracting the appropriate data from the literature with more relevance to the research question (Elo & Kyngäs, 2008; Sanusi et al., 2023a, 2023b). The coding scheme involved the following: titles of the articles, authors, pedagogical activities, methods, tools or resources, tool features, the impacts of the studies, and the modes. Figure 4 presents the PRISMA diagram showing the inclusion and exclusion process for data extraction. In the identification process in Fig. 4, there are articles identified through alternative sources, including searching through Google Scholar and references from other articles. The method of finding articles through article references is referred to as snowballing (Niazi, 2015; Oyelere et al., 2020). Snowballing refers to the process of using the reference list of an identified article or the citations of the paper to identify and select additional papers (Greenhalgh & Peacock, 2005). This was performed to mitigate the omission of relevant articles. Of the 3,099 articles retrieved, only 26 met the inclusion criteria for review. This represents a substantial proportion of excluded articles because most of the identified articles did not address CT and co-design at the same time and thus did not meet the inclusion requirement.

4 Results

This section presents the results of the SLR. The first sub-section shows the various implementations of the co-design pedagogy for CT. The second sub-section presents the identified co-design tools and resources. The third sub-section shows the categorizations relating to how co-design pedagogy has contributed to CT education at the K-12 levels (see Table 4). The selected studies are listed in Appendix 1.

4.1 Implementation of Co-design Pedagogy

This study identified how co-design pedagogical techniques were implemented for learning CT in K-12 education in a number of papers. This literature review revealed the learning settings involved in conducting a co-design study, the application of co-design pedagogical techniques, the implementation modes involved in applying these techniques, and various activities involved in the implementation of the techniques in teaching and learning CT.

4.1.1 Co-design Pedagogical Learning Settings

Different co-design pedagogical learning settings were identified in this study, as presented in Fig. 5. Bates (2015) refers to a learning setting as a location, context, and environment where learning takes place. In the context of this study, co-design pedagogical learning settings are the location, context, or environment that is used to accomplish or support teaching and learning using a co-design pedagogy. The identified co-design pedagogical learning settings include workshops, professional development (PD) programs, online, classrooms, and weekly clubs.

Figure 5 shows that the workshop is the most adopted learning setting for teaching and learning CT. The workshop was referred to as a structured event that enhanced learning via

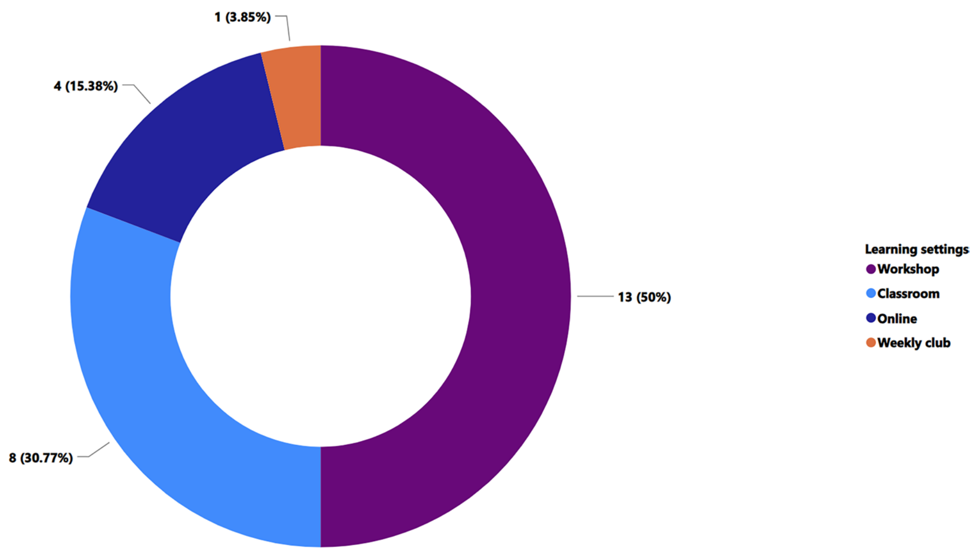


Fig. 5 Co-design pedagogical learning settings

the gathering of people with a clear objective to achieve a particular goal through collaboration. The event is facilitated by the workshop leader(s), who organizes and shares roles with each participant (Hamilton, 2016). From the analysis, 13 articles [S1, S3, S4, S7, S9, S13, S18, S19, S21, S22, S23, S24, S26] adopted workshops as a learning setting for co-design pedagogy for teaching and learning CT. For example, Kelter et al. (2021) conducted a four-week summer workshop with K-12 teachers, using five hours per day of physical attendance for three days a week. An online session was conducted on the remaining two days of each week. Bain et al. (2020) also conducted a four-week summer workshop for five hours a day. The teachers who participated worked in a school for 14 days and from home for the remaining six days by communicating online via email. Similarly, Biddy et al. (2021) conducted a three-day co-design workshop during the summer (see Appendix 1).

The classroom was also identified as a learning setting for a co-design pedagogy for CT (see Fig. 5). A classroom learning setting is described as an organized learning environment that involves individuals (the teacher and the students) in a room, whereby the teacher acts as the instructor or facilitator while the students participate as the learners. The participants interact to achieve socially determined goals (i.e., instructional goals or curriculum goals; Collins & Green, 1992). Eight research studies [S2, S5, S10, S11, S14, S15, S17, S20] adopted classroom learning settings to implement co-design pedagogies for CT. For example, Dickes et al. (2020) used classroom learning settings to enhance teachers' co-design pedagogical abilities to integrate CT into their various school curricula. This was conducted through a seven-month co-design lesson development program (a partnership between researcher and teachers), in which teachers served as learners and researchers as facilitators. The implementation of the study was further investigated with the teachers as they applied the ideas and knowledge gained in their various K-12 classrooms.

Further, an online environment was adopted in three [S6, S8, S16] different studies as a co-design pedagogical learning setting for CT (see Fig. 5). These studies involved using online platforms to coordinate teaching and learning. Online learning settings provide an avenue for facilitators or instructors and learners to meet online for collaboration and learning interactions. For example, Phuan et al. (2020) used online learning settings to learn CT through co-design to create storytelling with the Python programming concept. One article also used a weekly club as a learning setting for CT [S12] (see Fig. 5). The weekly club learning setting is a learning setting organized for students outside normal classroom hours in a high school, which is referred to as students' extracurricular activities hour (Cho et al., 2022; Lipscomb, 2007). This club hour is usually used for debates, arts, sports, and other extracurricular activities. It promotes student collaboration and allows like-minded students to relate and gain knowledge of new skills in preparation for a future outside the normal classroom (Cho et al., 2022; Lipscomb, 2007). Laato and Pope (2019) adopted weekly club learning settings to develop lightweight software with primary school students.

4.1.2 Application of Co-design Pedagogical Techniques and the Implementation Mode

This section presents identified studies that applied and implemented co-design as a pedagogical technique for learning and teaching CT and the type of learning activities adopted in the implementations. Pedagogical techniques are the set of techniques, methods, and instructions that permit successful teaching and learning, and create opportunities for gaining knowledge to build and improve competencies and attitudes within specific material or social contexts (Chartier & Geneix, 2007). This definition implies that pedagogical techniques are the approaches and practices aimed at imparting knowledge to learners and encouraging learners to obtain knowledge. Each technique determines the materials or instruments to be used for learning and can be student- or teacher-centered. Several pedagogical techniques exist such as inclusive, constructivist, problem-based, flipped, and co-design techniques. Moreover, the co-design pedagogical technique is a student-centered teaching technique that involves teamwork, which is facilitated (by teachers or researchers) based on the role assigned to each participant, with the main goal of building educational artifacts (Sunday, 2023). This section also presents the analysis of implementation modes. The implementation mode is relevant as it reveals the kinds of activities adopted for learning CT in the identified studies. The types of activities include plugged, unplugged, and hybrid activities. Unplugged activities refer to activities used in learning CT that do not involve the use of digital devices (such as computers, smartphones, and tablets). For example, cards, logic games, flipcharts, cardboard handling, or physical movements are used to represent and understand different concepts in computer science (Brackmann et al., 2017; Zeregal et al., 2021). Conversely, plugged activities refer to activities conducted with the use of digital devices; for example, Bee-Bot, Codequit, and micro:bit (Saxena et al., 2020; Morales et al., 2022). Hybrid activities involve the combination or application of both plugged and unplugged activities in a particular study.

This study identified several co-design pedagogical techniques adopted in CT K-12 education in the literature. Figure 6 shows that 18 articles [S1, S2, S4, S5, S6, S7, S8, S9, S11, S12, S13, S14, S16, S17, S18, S19, S21, S22] applied collaborative techniques as a pedagogical method for CT, which implies that a co-design study cannot be successful without some form of collaboration. However, there are collaborative studies that do not lead to design, which indicates the importance of this finding regarding collaborative techniques (Karumbaiah et al., 2019; Phuan et al., 2020). Within the 18 articles, 14 [S1, S4, S5, S6, S7, S8, S9, S11, S12, S17, S18, S19, S21, S22] adopted hybrid activities, five articles [S2, S6, S13, S14, S16] adopted only plugged activities, while only one article adopted [S8] unplugged activities. Collaborative techniques have been referred to as an instructional approach that involves a group of students learning through activating, sustaining, and regulating their thoughts, behaviors, emotions, and motivation to achieve their collective objectives through the social sharing of ideas and knowledge in the learning process (Järvelä et al., 2020; Lahann & Lambdin, 2020). These collaborative techniques motivate and enhance students' learning abilities through ideas and knowledge sharing between and within groups. Examples of how collaborative techniques were applied in plugged activities include social media and intelligent resources (storytelling system; Phuan et al., 2020) and a synchronous online forum for collaboration (Karumbaiah et al., 2019). In unplugged activities, collaborative techniques are applied by using non-digital device activities for collaboration in learning. They involve students sitting and working together on a non-digital task to brainstorm and share knowledge and ideas when solving the problem. Students use pens, papers, and sticky notes to share ideas and collaborate. For example, Apostolellis et al. (2014) developed a board game named RaBit EscApe that encapsulates CT for collaborative play and learning.

Reflective techniques were used in 14 articles [S1, S3, S4, S7, S9, S13, S14, S19, S20, S21, S23, S24, S25, S26] Ten [S1, S3, S4, S7, S9, S19, S21, S23, S24, S25] of the 14 articles adopted hybrid activities, while two [S13, S14] articles adopted only plugged activities, and two [S20, S26] utilized unplugged activities. Reflective techniques are instructional techniques engaged in by students concerning their experiences during or after learning, which enhance self-assessment, critical thinking, and introspection (Alt & Raichel, 2020; Guo 2022). This approach helps in fostering students' metacognitive self-consciousness and the

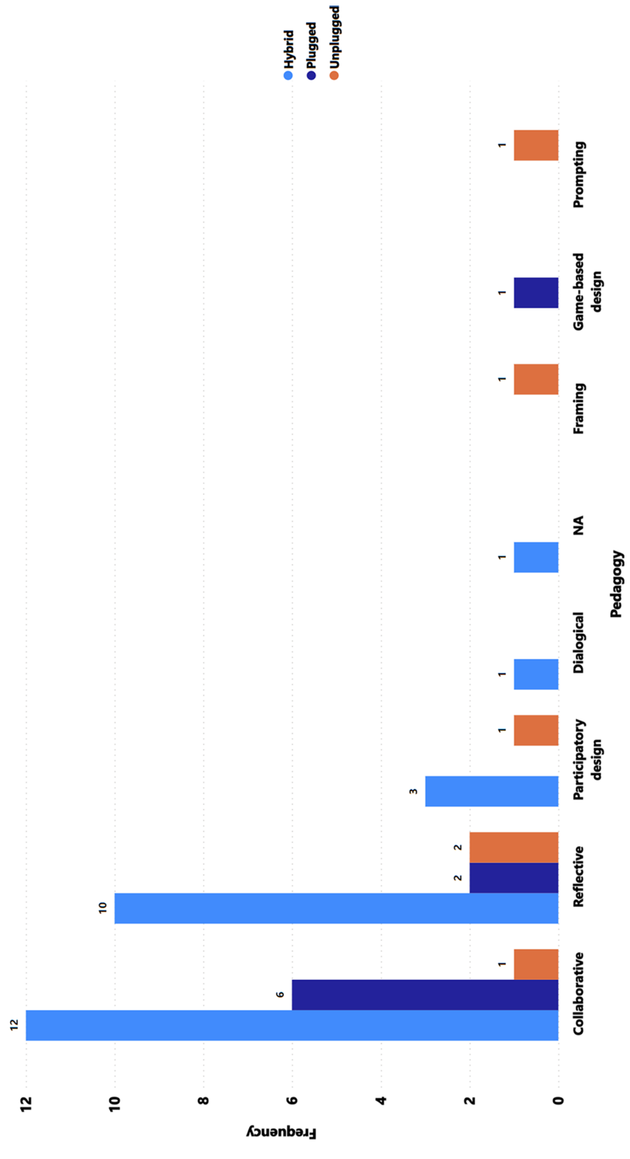


Fig. 6 Collected studies on co-design pedagogy and learning activities (Note: Hybrid = Combination of plugged and unplugged activities)

ability to interconnect new and existing knowledge (Alt & Raichel, 2020; Guo 2022). This technique also helps students to critically think about the new knowledge gained, evaluate themselves, and introspect on their learning experiences. In a co-design pedagogy, the reflective approach can also be implemented by seeking facilitators' reflections on the workshop design through interviews to ascertain the challenges faced and how to improve the effectiveness of the approach (Voigt et al., 2019). For example, Addone et al. (2021) applied the reflective approach by gathering feedback from the educators and users of the designed prototype based on their experience of using the prototype to rebuild, revise, and restructure the features of the tool. Plugged activities using the reflective approach involve reflection on activities performed during a study or class that utilizes digital devices, which can also be completed through online or digital devices. Unplugged activities using the reflective approach involve reflecting on activities conducted during a study or class that does not use digital devices. Rich et al. (2020) applied the unplugged reflective approach in the classroom by posing a question to students on the topic discussed in the classroom and requiring a verbal and/or demonstrative response. Similarly, unplugged activity reflection can be conducted via an audio discussion, interview, writing, and drawing (Van Mechelen et al., 2018; Morales et al., 2022). Examples of plugged activities for the reflective approach are reflective online essays, reflective online artifacts, online Google forms, and reflective online forms shared via email (Moudgalya et al., 2021; Van Mechelen et al., 2019; Voon et al., 2023).

Participatory techniques were adopted by four [S8, S12, S23, S25] articles. Three [S12, S23, S25] of the four articles employed hybrid activities, and only one [S8] article employed unplugged activities. The participatory technique is an instructional technique that involves collaboration, critical thinking, and the participation of learners in a teaching and learning process that emphasizes student-centered learning (Priya et al., 2020; Glenn, 2003; Reynolds-Cuéllar & Delgado 2020). This implies that participatory techniques encourage students' active participation in the learning process through collaboration and critical thinking to contribute to the learning process. For example, Karumbaiah et al. (2019) applied the co-design pedagogy to learn CT in an intensive 30-week participatory study for in-service teachers with researchers, which allowed teachers to introduce their classroom experience and collaboratively share ideas, with the common goal of teacher development and the creation of CT tools. Furthermore, one article [S16] adopted a game-based technique with only plugged activities. The game-based technique is an innovative, instructional approach that focuses on teaching through combining education and entertainment to enhance learning and relying on students' emotional and motivational engagement in the learning process (Hartt et al., 2020; Oyelere et al., 2023). This implies that students can learn when they are having fun. For instance, a co-design pedagogical approach was adopted by Breien and Wasson (2022) and Turchi et al. (2019). Similarly, Agbo et al., (2021a, 2021b) adopted a co-design pedagogical approach to teach CT through an online mini game design that empowered students to learn CT by collaboratively designing an educational mini game and peer reviewing each other's game prototype concepts.

Furthermore, framing, prompting, and dialogical techniques were adopted in a single article with unplugged activities, and one article does not specify which design-oriented pedagogical techniques were adopted. Dialogical techniques refer to how ideas are co-created by connecting different people's statements, utterances, and voices, and considering their various styles of speaking (Beeler & Lecomte, 2017). For instance, Dickes et al. (2020) implemented dialogue as a design-oriented pedagogy for teaching and learning CT through paying attention to different ideas and thoughts to formulate an effective CT curriculum for learning science. Framing is a way of presenting information to the public to influence people's perceptions (Scheufele & Tewksbury, 2007). For example, framing was applied as a design-oriented pedagogy for learning CT through teachers framing a particular idea that they wanted to teach the student and requiring a response from the student through reflection (Rich et al., 2020). Prompting is the act of stimulating, encouraging, promoting, or influencing another's action or reaction (Baxter et al., 2000). For instance, prompting was applied as a co-design pedagogy for learning CT through teachers prompting students to collaboratively work and respond to a particular concept, idea, or question that they wanted the students to learn and consider through the reflective method (Rich et al., 2020).

4.2 Co-design Tools and Resources

The identified tools and resources used in implementing a co-design pedagogy are presented in this section as tools and resources applied to support co-design pedagogies in CT education. The environment and mode of co-design pedagogy are also detailed.

4.2.1 Tools and Resources Applied to Support Co-design Pedagogies in Computational Thinking Education

The identification of tools and resources used in supporting co-design pedagogical techniques was inspired by Sanusi et al., (2021) and García-Peñalvo et al. (2016). The learning platforms and tools identified in this study were sorted into the four categories shown in Table 3. The table is structured into categories in conjunction with the corresponding references, as presented in Appendix 1. In this study, the tools and resources have been complementarily classified based on the type of activities, tools, and computer language used (García-Peñalvo et al., 2016; Sanusi et al., 2021). Some of the tools or resources fall under more than one category. For instance, NetLogo, ViMAP, and NetTango fall under both collaborative and programming tools. In the collaborative tools and resources, six different articles used NetLogo, followed by five articles using NetTango. Other collaborative tools and resources such as the CT Pathways Toolkit, ExposAR, a maglev train storyline, TAPASPlay, and Novelette were each used by one article. The collaborative tools category represents tools or resources used for collaboration during the co-design study for learning CT.

Similarly, the programming tools category includes tools and resources used in learning CT through programming with co-designing. Examples of programming tools include NetLogo, which was used by six articles, followed by NetTango, which was used by five articles, while micro:bit and Scratch were each used by two articles. Other programming tools including FunPlay Code, Construit, ViMAP, KEIRO software, and Codequit were each used by one article. Only two analysis tools were identified in this study. The analysis tools are tools and resources used for analyzing data during co-design study for learning CT in K-12 education. The identified tools were spreadsheet software, which was used by only one article, and CODAP, which was used by six different articles. Furthermore, the unplugged activity tools category includes the tools or resources used for learning CT through co-design without the use of electronic devices such as tablets, mobile phones, computers, or others (Kirçali & Özdener, 2023). Five unplugged activity tools were identified in this study. Sticky notes and paper were used by two articles, while other tools including pens, cardboard handling, and the ERP robot's physical buttons were each used by one article.

4.2.2 Environment and Study Mode Adopted for Co-design Pedagogy

Figure 7 presents the study mode and environment adopted for the co-design pedagogy in the research articles selected for this study. This study classified the study mode into

Table 3 Categorization of tools and resources

Categories	Tools and resources
Collaborative tools	CT Pathways Toolkit [S10] ExposAR [S11] Harmony Hippo [S12] Unity 3D platform [S12] CT-Integration Cycle [S13] Fablab@School project [S25] TAPASPlay [S16] NetLogo [S1, S3, S4, S7, S22, S23] NetTango [S1, S3, S4, S7, S22] CT Screener and Enhancer tools (lesson plans) [S26] ViMAP [S15] Novelette [S14] Bebras challenge [S9] The maglev train storyline [S13] ct4edu.org/resources [S26] The Engino Robotics Platform (ERP) [S18]
Analysis/visualization	Spreadsheet software [S13] CODAP [S1, S3, S4, S7, S22, S23]
Programming	FunPlay Code [S6] Construit (Javescript) [S12] ViMAP [S15] KEIRO software [S18] https://calliope.cc/en [S19] Codequit [S21] micro:bit [S13, S24] Scratch [S21, S24] Makecode [S24] NetLogo [S1, S3, S4, S7, S22, S23] NetTango [S1, S3, S4, S7, S22]
Unplugged activity	Pen [S12] Paper [S18, S12] Cardboard handling [S19] Sticky notes [S21, S24] ERP robot's physical buttons [S18]

Note: See Appendix 1 for studies (S)

synchronous, asynchronous, and hybrid. Synchronous mode refers to simultaneous interaction across devices during studies (Lunding et al., 2022). Synchronous mode promotes real-time interaction on a device, learning or teaching platform, or environment during studies. Unfortunately, none of the reviewed articles adopted only the synchronous mode. Most of the studies adopted an asynchronous mode, which represents sequential interactions across devices during studies (Lunding et al., 2022). The asynchronous mode does not permit real-time interaction on a device, platform, or in a learning environment during studies. For example, Phuan et al. (2020) adopted an asynchronous mode for a CT-based collaborative storytelling platform for learning CT with Python, which permits collaboration in an asynchronous mode, whereby participants enter their contributions, and they appear sequentially. The hybrid mode refers to the amalgamation of both synchronous and asynchronous modes for teaching and learning CT. Lunding et al.'s (2022) study is an example that used the hybrid mode; the study combined synchronous and asynchronous modes for teaching and learning CT. The tool developed by Lunding et al. (2022) provides

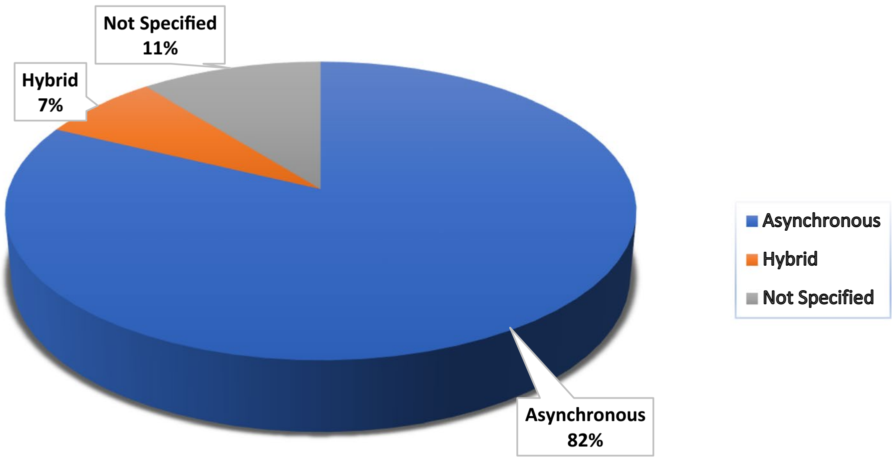


Fig. 7 Environment and mode of co-design

a sandbox environment that enables collaboration in real-time and the creation of the application. The tool permits collaboration based on the goal of the interaction and facilitates either close communication (synchronous interaction) or loose interaction (asynchronous interaction). The overall results show that 82% of the articles used asynchronous mode, while only 7% used hybrid mode. However, 11% of the articles did not specify the mode adopted.

4.3 Impact of Co-designing on Teaching and Learning CT

The categorization of how co-design pedagogy has impacted CT education at the K-12 levels is presented in Table 4. This study defined impact based on the contributions of each article to motivating, promoting, and enhancing the successful and efficient teaching and learning of CT through a co-design pedagogy. These contributions could be in terms of the design and implementation of co-design CT artifacts, tools, or resources; the unraveling of innovative pedagogies for CT; the co-design of CT curriculum, development of co-design CT materials, and integration of co-design CT; the use of co-design concepts or ideas; co-designing to build CT innovation and creativity; and the application of co-design in coding and programming to teach and learn CT. The impacts identified in the reviewed articles are presented in Table 4.

Table 4 reveals that the impacts of the studies were interrelated. In other words, some of the articles' contributions cover more than one category. For example, S1 contributes in terms of CT tool development concepts and skill learning development. Article S22 contributes in terms of designing tools, the application of co-design pedagogy, and CT concepts and skill development. Article S15 contributes by using programming, a co-design pedagogy, and building CT integration skills. Table 4 indicates that 12 of the 26 papers fall under category 4, "CT design implementation and integration skills," which implies that most of the articles concentrate on using the co-design approach to teach CT integration skills, develop a curriculum for integrating CT, develop curriculum materials to teach CT, or other related CT integration implementations. The first category (Tool or resource articles) encompasses

Table 4 Categorizations of the study impacts

SN	Categorization	Article serial number (Appendix 1)	Impact of the study
1	Tools or resources	S1, S3, S7, S10, S11, S14, S16, S22	This category represents developed or designed CT tools or resources and adopted tools or resources used for teaching and learning CT. Examples of tools and resources include augmented reality (AR), web-based platforms, and game-based learning platforms. The impact of this category is predicated on the design of co-design CT tools, resources, or artifacts for teaching and learning CT
2	Innovative pedagogies	S1, S2, S4, S5, S9, S11, S14, S15, S18, S19, S20, S22, S24, S25, S26	This category presents articles that utilized various pedagogical innovations in co-design for teaching and learning CT. For example, collaborative, constructionist, game-based, reflective, participatory, prompting, framing, and dialogical techniques. The influence of this category is predicated on the study's ability to apply a unique technique for teaching and learning CT via co-design
3	CT concepts, ideas, perspectives, and skills	S1, 16, S18, S22, S23	This category presents articles that focus on different CT concepts, ideas, perspectives, and skills. For example, algorithmic thinking, pattern recognition, decomposition, problem-solving, and abstraction. The impact of this category is predicated on the use of co-design to teach CT concepts, ideas, perspectives, and skills

Table 4 (continued)

SN	Categorization	Article serial number (Appendix 1)	Impact of the study
4	CT design implementation and integration skills	S4, S5, S8, S9, S12, S13, S14, S15, S17, S23, S24, S25	This category presents articles that implement the design and development of curriculum co-design for teaching and learning CT integration skills, CT integration curriculum development, CT integration curriculum materials, and other related CT integration implementation articles. For example, co-design workshops for integrating CT into STEM curricula, teachers' professional development programs for CT integration skills development, in-service teacher CT learning programs through co-design, and designing middle school CT integration with sensor technology. The impact of this category is predicated on using co-design to develop CT curricula, materials, and CT curriculum integration skills
5	CT innovation and creativity development via co-design	S6, S13, S14, S20, S21, S25	This category presents articles that cover creativity and innovation building through co-design for teaching and learning CT. For example, online co-design for developing a mini game, implementing CT co-design with students to build a robot, and innovation development through co-design. The impact of this category is predicated on using co-design to build CT's inventive advantage and innovation
6	Programming learning	S12, S15, S21, S25	This category presents papers that cover teaching and learning programming via co-design for CT or articles that teach CT concepts through co-design programming. For example, using programming to teach and learn CT integration, and using programming skills to develop and improve the student's knowledge of CT. The impact of this category is predicated on using programming and coding for learning CT through co-design

seven articles, which means that eight of the 26 papers address development or designing CT tools or resources for teaching and learning CT, including AR, web-based platforms, and game-based tools. In addition, the third (CT concepts, ideas, perspectives, and skills) and fifth (CT innovation and creativity development via co-design) rows in Table 4 contain six of the 26 articles each. The third (CT concepts, ideas, perspectives, and skills) row incorporates articles that present different CT concepts, ideas, perspectives, and skills. The fifth row (CT innovation and creativity development via co-design) includes scientific papers that contain creativity and innovation building through co-design for teaching and learning CT. The second row encompasses studies with innovative pedagogies for learning and teaching CT. In addition, the sixth (learning programming via co-design CT) row addresses papers that include a co-design approach for teaching and learning CT through programming or articles that teach CT concepts via co-design programming lessons.

5 Discussion

This research work investigated the existing literature to discover how co-design pedagogical techniques have been used in CT education at K-12 levels; what tools and technologies have been used for co-design pedagogical techniques in CT education; and how studies on co-design pedagogy have contributed to CT education. This section discusses the study findings and offers reflections to provide answers to the research questions.

RQ1 How has the co-design pedagogy been implemented for computational thinking education at K-12 levels?

Our study revealed that workshops are the most utilized learning setting in the co-design pedagogy for CT. This finding is in line with Wu et al. (2020), who demonstrated that teachers are able to understand CT through workshops. Similarly, Curzon et al. (2014) attested to the fact that a workshop is one of the best methods of teaching CT, as demonstrated in their study. Moreover, this study, as expected, identified that the collaborative technique is the most frequently used co-design pedagogical technique for teaching or learning CT. This finding is consistent with the study conducted by Echeverría et al. (2019), who found that collaborative techniques are used in education to enhance teaching and learning from a variety of fields. Echeverría et al. (2019) further stated that collaborative techniques influence learners' positive skill development and confidence building. Whereas CT is referred to as a skill required for solving problems (Agbo et al., (2019); Shute et al., 2017;, collaborative techniques are needed for learning CT. Similarly, Wu et al. (2019) also support our finding by referring to collaborative techniques as an efficient technique for learning and helping to build skill competence. Similarly, reflective learning techniques were also identified as one of the strong co-design pedagogical techniques for teaching and learning CT. Thus, this study's finding is also supported by Wu et al.'s (2020) findings, which demonstrated that teachers learned computational practices and tools through participatory reflective meetings. The reflective approach also helps learners appreciate the skills learned and increase their confidence, as argued by Daniels et al. (2020). However, in several studies, reflective techniques were combined with other techniques.

Our analysis also identified dialogical, prompting, and framing techniques as the least used co-design pedagogies in CT education. Dialogical techniques refer to the ways in which ideas are co-developed by connecting diverse people's speech, utterances, and voices, and considering their varied styles of speaking (Beeler & Lecomte, 2017). This implies dialogical techniques are heterogeneous statements or contributions by study participants that are gathered to form meaningful ideas in the learning process. Similarly, framing is a way of presenting information to the public to influence people's perceptions. Framing involves selecting a message and emphasizing a particular point to shape and obtain people's reactions. Framing as a co-design pedagogy was used by choosing and communicating a topic to the students and emphasizing a particular point of the topic in different forms to shape and improve the students' understanding of the topic and obtain their responses. Framing is mostly used in media, public relations, advertising, and politics (Chong & Druckman, 2007; Scheufele & Tewksbury, 2007) but has been adopted in co-design for learning CT. Prompting is the act of stimulating, encouraging, promoting, or influencing someone's action or reaction (Baxter et al., 2000). Prompting offers a means of gathering pragmatic data on values and theoretical viewpoints in the collaborative design process (Yoo et al., 2013). Prompting as a co-design pedagogy was used to stimulate students' reactions to a particular topic to trigger further discussion on the topic. The prompting and framing identified as co-design pedagogies in this study cannot be used alone without a means of receiving a reply. Thus, reflective techniques can be used as a means of feedback from participants. Furthermore, this study identified the hybrid mode (hybrid activities) as the most frequently adopted implementation mode in the co-design pedagogy for CT K-12 education. This finding corresponds to Caeli and Yadav's (2020) study, which asserts that CT skills should be taught to students using both plugged and unplugged activities to allow them to gain a better understanding of the CT concepts. Similarly, Saxena et al. (2020) support our finding by stating that unplugged activities can be used to complement plugged activities to improve students' understanding of CT skills and concepts.

RQ2 What tools and technologies have been used for co-design pedagogy in computational thinking education at K-12 levels?

This study classified the identified tools and resources into four main categories. The identified tools and resources were categorized into collaborative tools, analysis/visualization, programming, and unplugged activity. Some of the tools or resources identified occupy more than one category, for example, NetLogo, NetTango, and ViMAP. Notably, NetLogo was conspicuous among all the tools or resources utilized in the literature because it appears in six of the research papers. This tool is a free online programming modeling platform used by many (Wu et al., 2020). It is also referred to as a modeling platform utilized by teachers, students, and researchers. NetLogo can be used to teach and model different subjects such as biology, art, chemistry, computer science, and many others. Within those subjects, NetLogo can also be used to teach and learn different topics. For example, in biology, NetLogo can be used to teach students about blood sugar regulation, autumn, muscles, moths, and membrane. In chemistry, NetLogo can be used to learn about chemical equilibrium and simple kinetics. In physics, NetLogo can be used to learn about diprotic acid and Gaslab Adiabatic Pistons. In computer science, NetLogo can be used to learn about k-means clustering, Robotic Factory, Turing Machine 2D, and particle system fountains. In the analysis and visualization tool category, Common Online Data Analysis Platform (CODAP) was found to be an excellent analysis tool with features that can be used to view data in maps, slides, graphs, tables, and many others (Kelter et al., 2021; Peel et al., 2020). In CODAP, data are stored in tables with additional columns available for other entries such as formulas or results, as demonstrated by Bain et al. (2020). An interdependency between NetLogo and CODAP may exist, as all the articles that used NetLogo also used CODAP for analysis.

Furthermore, an asynchronous study mode was applied in most of the studies, which is in line with Malkin et al. (2018), who identified the asynchronous teaching mode as an efficient means of teaching in an online class discussion and expressed satisfaction with the students' performance. Similarly, Lytvyn et al. (2021) identified and enumerated the benefits of the asynchronous learning mode for making learning flexible, encouraging personal working pace, making learning accessible anytime and anywhere, and promoting individual time management and self-regulation. The hybrid mode identified in this study is the application of both synchronous and asynchronous modes in one study. The synchronous mode of learning was utilized in conjunction with the asynchronous mode to promote successful learning.

RQ3 How have studies on co-design pedagogy contributed to computational thinking education at K-12 levels?

Reflecting on the impact of co-design as a pedagogy on learning CT in this study, most of the reviewed articles focused on the teachers' development. Hence, this section discusses the effect of the categorized impact as presented in the results starting with the teacher and followed by the students.

The evaluation and assessment of the impact of co-design on learning CT revealed that teachers could use co-design as a pedagogical technique to learn, think, and re-think the creation of teaching content for learning CT. The teachers gained content creation abilities mostly during workshops and teacher training programs, which were adopted as methods and learning settings to enlighten teachers on developing CT materials. This was in line with Kelter et al. (2021), who identified PD programs as a means of equipping teachers with skills, knowledge, and ideas. Similarly, Kennedy (2016) confirms that knowledge and content creation abilities were mostly gained in PD programs, while Curzon et al. (2014) presented workshops as one of the best means of introducing teachers to CT (Curzon et al., 2014; Kelter et al., 2021; Kennedy, 2016). Similarly, this study shows that the teaching materials created by teachers were able to be integrated and implemented in the teachers' classrooms (Biddy et al., 2021; Rich et al., 2020). Additionally, this study shows that through the CT integration process, teachers were able to identify the challenges tutors generally faced while teaching CT, such as understanding how to connect the current pedagogy to the technical aspects of computer science such as algorithmic thinking (Mills et al., 2020).

Furthermore, the use of playful environments such as game-based environments and virtual reality could help in attracting, occupying, engaging, and improving student learning (Oyelere et al., (2023); Turchi et al., 2019;. The use of appealing visual displays in CT teaching and learning resources or tools empowers students to learn physical content (Bain et al., 2020). In addition, creating captivating contexts and content could attract and engage students to develop iteration and problem-solving skills (Voigt et al., 2019). The comparison of notes and ideas in collaborative learning can also awaken, improve, and shape students' creativity and innovation skills (Agbo et al., (2021a, 2021b; Peel et al., 2020). Collaborative learning motivates and influences students' participation in the learning process and develops CT skills (Zhang et al., 2019).

6 Study Limitations

This study is subject to the limitations common to review articles. This research is limited, firstly, by the number of databases explored. The search was restricted to only five databases (IEEE Xplore, Web of Science, ACM, SpringerLink, and Scopus), which may not include all the relevant literature. Furthermore, the search string in the study was limited by using two search terms, "co-design" AND "computational thinking," in all fields (metadata). In addition, the review did not include cross-cultural articles, such as articles not published in the English language and non-peer-reviewed literature. Non-CT articles were also excluded, along with articles that did not address teaching and learning CT.

Notably, this study found 26 relevant articles based on the defined search protocol and inclusion criteria. Previous research has reviewed articles by utilizing a smaller number of papers such as the work by Barreto and Benitti (2012), who conducted a review with 10 articles to identify the effectiveness and benefits of adopting robotics as a tool in education, and Su and Yang (2022), who carried out a review with 17 articles to identify the AI-related activities and tools used in early childhood education and the concepts embedded in the tools. Using other strategies, criteria, and expanding the search databases may have generated more articles. Thus, this article should be regarded as an attempt to probe into the learning and teaching of CT through co-design as a pedagogical technique, rather than an exhaustive overview.

7 Conclusion

Interest in the teaching and learning of CT has been mounting since its identification as a fundamental technique for developing problem-solving skills. Equally, the use of co-design has been identified as a pedagogical tool for improving the learning efficiency of students in developing CT skills. These factors have attracted researchers and educators to further use co-design as a pedagogy to introduce CT concepts, ideas, and practice; develop a curriculum for learning CT; integrate CT into STEM education; and develop materials for learning CT. Research has also presented co-design as a way to train teachers how to integrate and develop materials and curricula and teach CT in the classroom. Accordingly, this study found that workshops represent the most frequently adopted co-design pedagogical learning setting for learning CT in K-12 education. Prompting, framing, and dialogue were identified as unique and infrequent co-design pedagogical techniques for learning CT. The combination of plugged and unplugged activities is seen as a viable co-design activity for learning CT. The study also revealed that CODAP is the most frequently used visualization and analysis tool, while NetLogo is the collaborative tool most utilized for learning CT.

This study contributes to practical knowledge by unraveling the uniqueness of dialogical, prompting, and framing techniques as part of the co-design pedagogy. Our study also helps teachers to identify tools that could be used to co-design in CT K-12 education. These techniques could further be promoted and explored as a pedagogical technique in K-12 education. The study also revealed that game-based techniques were one of the most frequently used pedagogical techniques for CT education in K-12 levels. This study recommends that game-based techniques could further be used as a co-design pedagogical technique by teachers in K-12 education. Game-based techniques have been proven to be successful and efficient in stimulating learners' interest, sustaining their learning progress, and helping them achieve success in their learning pursuits (Hartt et al., 2020; Oyelere et al., 2023). This study forms a part of doctoral research aimed at designing and implementing co-design pedagogical scenarios for learning CT. This review has unraveled the co-design pedagogical techniques adopted for learning CT in K-12 and how the co-design process has been implemented to foster CT learning in K-12 education. Future studies will be directed at testing and evaluating existing co-design pedagogical tools, processes, resources, or methods with K-12 students and identifying their usability, impacts, and user requirements.

Appendix 1

Overview of data reviewed in this study.
S = Serial number, NA = Not applicable.

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S1	Constructionist co-design: A dual approach to curriculum and professional development	Kelter et al. (2021)	Collaborative and reflective	Plugged and unplugged	Workshop	NetLogo, NetTango and CODAP	NetLogo contains features such as setup, go, model library, file new, export, choose file, command center, code, and model info, etc. NetTango is a blocked-based interface of NetLogo that can be embedded into a web-based NetLogo that makes it easy to create an agent-based model. CODAP is a data analysis platform whose features include Tables, Graphs, Map, Sliders, Calc, Text, Plugins, undo, redo, tiles, options, and help	Asynchronous	This study discovered that workshops could help teachers learn computational thinking practices and tools, through collaboration with researchers to co-design and generate a computationally improved STEM curriculum, which can engage teachers and K-12 students in the classroom. It further Identifies that constructionist co-design is a hopeful teaching technique that can aid both curriculum and professional development simultaneously, although there are challenges in an attempt to navigate and achieve the two goals concurrently	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S2	Resources for computational thinking: Co-designing with teachers	Mills et al. (2020)	Collaborative	Pluuged	Classroom	NA	The CT learning resources is a problem-solving computational system that contains a teacher's guide, probing questions, tools-related practices, and multiple assignment templates for students	NA	This study discovered the general challenge faced by tutors in teaching CT, which is understanding how to connect the pedagogy and the technical aspect of computer science such as algorithmic thinking	USA
S3	Back to computational transparency: Co-designing with teachers to integrate computational thinking in science classrooms	Bain et al. (2020)	Reflective	Plugged and unplugged	Workshop	NetLogo, NetTango, and CODAP	NetLogo contains features such as setup, go, model library, file new, export, choose file, command center, code, and model info, etc. NetTango is a blocked-based interface of NetLogo that can be embedded into a web-based NetLogo that makes it easy to create an agent-based model. CODAP is a data analysis platform whose features include Tables, Graphs, Map, Sliders, Calc, Text, Plugins, undo, redo, tiles, options, and help	Asyncroneous	The study revealed that students were able to use computational thinking tools to learn physics content	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S4	Workshops and co-design can help teachers integrate computational thinking into their k-12 stem classes	Wu et al. (2020)	Collaborative and reflective	Plugged and unplugged	Workshop	NetLogo, CODAP, NetTango, and NetLogo	NetLogo contains features such as setup, go, model library, file new, export, choose file, command center, code, and model info, etc. NetTango is a blocked-based interface of NetLogo that can be embedded into a web-based NetLogo that makes it easy to create an agent-based model. CODAP is a data analysis platform whose features include Tables, Graphs, Map, Sliders, Calc, Text, Plugins, undo, redo, tiles, options, and help	Asyncroneous	(1) help teachers develop an understanding of CT and (2) empower teachers to integrate and teach CT in their STEM courses	USA
S5	Computational Thinking Integration Design Principles in Humanities	Caskurlu et al. (2022)	Collaborative	Plugged and unplugged	NA	NA	NA	Asynchronous	The study aimed at supporting those who are interested in integrating computational thinking into Arts, Social Studies, and Language Arts in K-12 education, which includes teachers' professional development providers, and curriculum writers	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S6	Ct-based collaborative storytelling for learning programming concepts in python	PHUAN et al. (2020)	Collaborative and reflective	Plugged	Online	FunPlay Code Python-based Web application	It is a platform that can recognize programming logic, semantics, and pattern. It is restricted to running only python code. It permits users to make their individual digital stories and adapt and reuse the code. The platform also permits editing of the existing codes and functions such as comment, like, and share	Asynchronous	This study will stir up the creativity and innovative ability of participants via a combination of logic and design. The study aims to reduce mental blocks and open the minds of participants through co-design and appreciation of computer science and its significance in every area of human daily life	Hong Kong
S7	A case study of teacher professional growth through co-design and implementation of computationally enriched biology units	Peel et al. (2020)	Collaborative and reflective	Plugged and unplugged	Workshop	NetLogo, CODAP, NetTango, NetLogo Web	NetLogo contains features such as setup, go, model library, file new, export, choose file, command center, code, and model info, etc. NetTango is a blocked-based interface of NetLogo that can be embedded into a web-based NetLogo that makes it easy to create an agent-based model. CODAP is a data analysis platform whose features include Tables, Graphs, Map, Sliders, Calc, Text, Plugins, undo, redo, tiles, options, and help	Asynchronous	The work contributes to knowledge by helping tutors develop and implement computing-enhanced curricula for integrating computational thinking and science	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S8	Using participatory design to facilitate in-service teacher learning of computational thinking	Karumbaiah et al. (2019)	Collaborative and Participatory	Unplugged	online	NA	NA	Asynchronous	The facilitators empowered the teachers with their CT technological knowledge in other to improve the teacher's technological knowledge in aiding students learning of CT. Also, the study ensures that teachers learn technological content rather than building a tool	USA
S9	Bebras-inspired Computational Thinking Primary School Resources Co-created by Computer Science Academics and Teachers	Lehtimäki et al. (2022)	Collaborative and reflective	Plugged and Unplugged	Workshop	https://www.ics.ie/news/bebras_challenge_2020_success	NA	Asynchronous	The study participant facilitated the distribution of CT materials to 7,145 students. The generated feedback motivated the co-designing of the workbook and lesson plans for CT Berbras-style	Ireland

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S10	Developing Inclusive Computing with the CT Pathways Toolkit	Coenraad et al. (2022)	NA	Plugged and unplugged	Classroom	CT Pathways Toolkit	The toolkit guides districts through planning, building, and implementing their pathway while centering their district's equity needs	Asynchronous	This study enhances research on computing pathways at the district level particularly concentrating on encouragement of student inclusivity	USA
S11	ExposAR: Bringing Augmented Reality to the Computational Thinking Agenda through a Collaborative Authoring Tool	Lunding et al. (2022)	Collaborative	Plugged and unplugged	Classrooms	ExposAR	ExposAR is made up of AR mobile view (AR interface) and desktop view (model interface), it is divided into three phases; the authoring, player, and analysis phases. The authoring phase is used by users to build their AR game, the play phase is used by users to play the game, and the analysis phase is used by users to analyze and reflect on the game-played interactions	Hybrid	This study revealed ExposAR which is a collaborative AR prototype that engages students in AR ideas, perspectives, and procedures. It provides educational AR tools development principles that exposed users to the AR fundamental processes and abilities. ExposAR evaluation on design principles revealed CT future agendas for AR	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S12	A light-weight Co-construction activity for teaching twenty-first century skills at primary schools	Laato and Pope (2019)	collaborative and participatory design	Plugged and unplugged	Weekly club	Harmony Hippo, Unity 3D platform, Construit, GIMP	The Harmony Hippo tool is used for composing music with mathematics. Unity 3D platform was used for application development, Construit is a javascript online platform, GIMP is drawing programs app	Asynchronous	The study improved student desire and interest in programming, whereby the participating students learned contemporary skills and were able to write children-related codes. It also revealed an inspiring match to the Finnish overall curriculum plan	Finland
S13	Designing a middle school science curriculum that integrates computational thinking and sensor technology	Gendreau et al. (2019)	Collaborative and reflective	Plugged	Workshops	Spreadsheet software, sensor system	The sensor system was used to collect data and it contains an SD card where the data were stored. The important data were visualized and analyzed in the spreadsheet software (Google sheets)	Asynchronous	The study successfully developed a storyline unit that relatively meets several specifications constraints of the integration process, which involved teachers and researchers in the co-design process	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S14	Novelette, a Usable Visual Storytelling Digital Learning Environment	Addone et al. (2021)	Collaborative and reflective	Plugged	Classroom	Novelette	Novelette features include class management, an incipit mechanism, a digital storytelling editor, and a suggestion mechanism. The class management features contain the management dashboard where management can view, edit, or control the content, such as creating classes, account tabs, etc. The incipit mechanism is used by the teachers to introduce topics to students by giving the introduction of the story and encouraging the students to continue the story from where he stopped. The digital storytelling editor aids the users with the student role to author stories according to the workflow plan. The suggestion mechanism is a blank page that offers users suggestions mechanism of analogies, antonyms, synonyms, and rhymes depending on the user's defined word	Asynchronous	The outcome of the study revealed that the tool is usable by primary school students, and it helps them to build computational creativity in refining and rendering contextual and graphical stories	Italy

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S15	Sociomathematical Norms for Integrating Coding and Modeling with Elementary Science: A Dialogical Approach	Dickes et al. (2020)	Dialogical approach	Plugged and unplugged	Classroom	ViMAP (Sengupta et al., 2015)	ViMAP is an agent-based visual programming language, which contains, a command library, assessment agent-level attributes, modeling (increasing and decreasing functions and writing procedures), and measure commands	Asynchronous	This study identifies the relationships that exist between CT, mathematical thinking, and scientific modeling, when the computing medium is agent-based programming. Agent-based programming was used as a medium for building sociomathematical norms models across the mathematical quality of the model with the teacher's support as revealed by this research work. This study also revealed that teachers without computing background knowledge can help in integrating programming into science curricula that have been in existence	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S16	Fostering computational thinking through collaborative game-based learning	Turchil et al. (2019)	Collaborative and game-based design	Plugged	Online	TAPASPlay	It contains puzzle metaphor (for algorithm thinking) and puzzle pieces (for abstraction)	Asynchronous	This study showcases TAPASPLAY as a constructionist learning approach tool for learning CT via gameplay. It makes students learn CT through social and entertainment. It provides a playful environment for students to collaboratively learn and build CT skills	UK
S17	Computational Thinking in Preschool: Bridging Home and School	Kamdar et al. (2021)	Collaborative	Plugged and unplugged	Classroom	NA	NA	NA	This study revealed that CT integration into STEM courses was successful, the families' activities were useful to the families and the families decided to integrate the activities into their daily life routines	Germany

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S18	A Co-design Approach for Developing Computational Thinking Skills in Connection to STEM Related Curriculum in Swedish Schools (P156-159)	Zerega1 et al. (2021)	collaborative	Plugged and unplugged	workshop	The Engino Robotics Platform (ERP), KEIRO	KEIRO is a block-based programming software that is used for developing advanced algorithms EPR is a construction tool, it has a physical button used for basic function manual programming	Asynchronous	This study introduced students to CT concepts and practices which address the goals of Swedish Agency for Education. The study identified that having a strong collaboration between the researcher and the teachers to perform codesign educational activities that focus on constructionist and challenged-based learning can bring about efficient approaches for students' effective development of CT skills	Sweden
S19	Design Thinking with Children: The Role of Empathy, Creativity and Self-Efficacy	Christian Voigt et al. (2019)	Collaborative, reflective	Plugged and unplugged	Workshop	Calliope micro-board	Calliope micro-board contains the following features: input and output, sensors, extension possibilities, and system	Asynchronous	This study argued that making a fascinating context for keeping children engaged in problem-solving and performing iterations can help in solving the challenge of developing learning activities for the intersection of digital fabrication	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S20	Children's Assessment of Co-Design Skills: Creativity, Empathy and Collaboration	Van et al. (2019)	Reflective	Unplugged	Classroom	NA	NA	Asynchronous	This study revealed that participants acknowledge their present understanding and acknowledge the knowledge and skills (creativity, empathy, and collaboration) they acquire through reflections on the activities	USA
S21	Reimagining and Co-Designing with Youth an Hour of Code Activity for Critical Engagement with Computing	Morales et al. (2022)	Reflective and collaborative	Plugged and unplugged	Workshop	Scratch, sticky note, codequit	In Scratch, the following features were used: sticky note, CodeQuilt. Stickies (Physical and digital sticky notes) is used by the student to share their ideas and perspectives on coding and computer programming, while the CodeQuilt is a scratch feature that is used for collaboration during learning project, allowing the learner to take ownership of their work (stories about code) by building digital quilt patches used testing CodeQuilt, participants reflection on CodeQuilt and remixing CodeQuilt	Asynchronous	Participants in this study enumerate that coding offers people privilege to learn, create things, and provide solutions to complex problems, build games, software applications for companies, and institutions	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S22	Seeds of (r) Evolution Constructionist Co-design with High School Science Teachers	Kelter et al. (2020)	collaborative	Plugged and unplugged	Workshop	NetLogo, NetTango, and CODAP	NetLogo contains features such as setup, go, model library, file new, export, choose file, command center, code, and model info, etc. NetTango is a blocked-based interface of NetLogo that can be embedded into a web-based NetLogo that makes it easy to create an agent-based model. CODAP is a data analysis platform whose features include Tables, Graphs, Map, Sliders, Calc, Text, Plugins, undo, redo, tiles, options, and help	Asynchronous	This study revealed that teachers develop skills and confidence in computational thinking and computational tools. Teachers were also empowered to be constructionist evolutional agents. Likewise, it helps teachers increase their knowledge as educators and researchers and got more understanding of the sensitive challenges teachers faced in carrying out their duties	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S23	A Professional Development That Helps Teachers-Integrate Computational Thinking Into Their Science-Classrooms Through Codesign	Wu et al. (2022)	participatory, and reflective	Plugged and unplugged	Workshop	NetLogo, CODAP, and https://ct-stem.northwestern.edu/curriculum/preview/495/pem_code/HE28N68HDQQSZE9XWSPM,	The tool was used to introduce students to computational modeling, which involves the utilization, modification, and debugging of computational models which stimulated forest fire (NETLOGO). Common Online Data Analysis Platform (CODAP) was used by the student to collect data, evaluate and analyze the development	Hybrid	Teachers in this study gained more understanding of CT, discipline content, and co-design of curricula. Likewise, their interest, skills, and knowledge were generally improved	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S24	A Professional Development Model to Integrate Computational Thinking Into Middle School Science Through Codesigned Storylines	Biddy et al. (2021)	Reflective	Plugged and unplugged	Workshop	CT-Integration Cycle	NA	Asynchronous	The participant's ability was greatly increased after a deep engagement in CT and facilitating CT integration to students with thoughtful consideration using the professional development model	USA

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S25	Computational empowerment: participatory design in education. CoDesign 16, 1 (2020), 66–80	Dindler et al. (2020)	Participatory design, and reflective	Plugged and unplugged	NA	Fablab@School project	NA	NA	The study presented computational empowerment as a focus shift from programming skills to developing young children with important skills such as computational thinking, digital empowerment, digital design and design processes, and technological ability to actively engage in technology development. Also, the study revealed CT integrated course curriculum for STEM students, which will empower the student to be future technology builder	Denmark

ID	Title of articles	Authors and date	Pedagogy (RQ1)	Activities (RQ1)	Learning settings (RQ1)	Tools/ resources (RQ2)	Tools Features (RQ2)	Mode (RQ2)	Impact of study (RQ3)	Country
S26	Teacher implementation profiles for integrating computational thinking into elementary mathematics and science instruction	Rich et al. (2020)	framing, prompting, and inviting reflective	Unplugged	Workshop	ct4edu.org/resources	The resource features include home, team, educator resources, research, blog, and video	Asynchronous	This study discussed the three teaching strategies adopted by teachers to teach CT concepts, these strategies include: prompting, inviting reflection, and framing; furthermore, in the study, the teacher's implementation of the strategies was grouped into four, which include: (1) the presentation of CT as a universal strategy for problem-solving, (2) structuring lessons with CT, (3) stressing the importance of CT through prompting, and (4) adopting CT as a guide to teacher planning	USA

Funding Open access funding provided by University of Eastern Finland (including Kuopio University Hospital). Not applicable.

Data Availability The authors declare that the data supporting the findings of this study are available within the paper as appendices.

Declarations

Conflict of interest The authors declare that they have no competing interests.

References

Addone, A., De Donato, R., Palmieri, G., Pellegrino, M. A., Petta, A., Scarano, V., & Serra, L. (2021). Novelette, a usable visual storytelling digital learning environment. *IEEE Access, 9*, 168850–168868.

Agbo, F. J., Oyelere, S. S., Suhonen, J., & Adewumi, S. (2019). A systematic review of computational thinking approach for programming education in higher education institutions. In *Proceedings of the 19th Koli calling international conference on computing education research* (pp. 1–10).

Agbo, F. J., Sunday Oyelere, S., Suhonen, J., & Tukiainen, M. (2021). iThinkSmart: Immersive virtual reality mini games to facilitate students' computational thinking skills. In *21st Koli calling international conference on computing education research* (pp. 1–3).

Agbo, F. J., Oyelere, S. S., Suhonen, J., & Laine, T. H. (2021). Co-design of mini games for learning computational thinking in an online environment. *Education and Information Technologies, 26*(5), 5815–5849.

Agbo, F. J. (2022). Co-designing a smart learning environment to facilitate computational thinking education in the Nigerian context (Doctoral dissertation, Itä-Suomen yliopisto). https://erepo.uef.fi/handle/123456789/27287

Aksela, M. (2019). Towards student-centred solutions and pedagogical innovations in science education through co-design approach within design-based research. *LUMAT: International Journal on Math, Science and Technology Education, 7*(3), 113–139.

Alhumaidan, H. (2017). *Co-design of augmented reality textbook for children's collaborative learning experience in primary schools* (Doctoral dissertation, Loughborough University).

Aljowaed, M., & Alebaikan, R. A. (2018). Training needs for computer teachers to use and teach computational thinking skills. *International Journal for Research in Education, 42*(3), 237–284.

Alt, D., & Raichel, N. (2020). Reflective journaling and metacognitive awareness: Insights from a longitudinal study in higher education. *Reflective Practice, 21*(2), 145–158.

Angeli, C., & Giannakos, M. (2020). Computational thinking education: Issues and challenges. *Computers in Human Behavior, 105*, 106185.

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

Apostolellis, P., Stewart, M., Frisina, C., & Kafura, D. (2014). RaBit EscAPE: a board game for computational thinking. In Proceedings *of the 2014 conference on interaction design and children* (pp. 349–352).

Bain, C., Anton, G., Horn, M., & Wilensky, U. (2020). Back to computational transparency: Co-designing with teachers to integrate computational thinking in science classrooms. In International *conference of the learning sciences* (No. Jun-2020).

Barreto, F., & Benitti, V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education, 58*, 978–988.

Bates, A. W. (2015). *Teaching in a digital age: Guidelines for designing teaching and learning*. BCcampus.

Bati, K. (2022). A systematic literature review regarding computational thinking and programming in early childhood education. *Education and Information Technologies, 27*(2), 2059–2082.

Baxter, S. D., Thompson, W. O., & Davis, H. C. (2000). Prompting methods affect the accuracy of children's school lunch recalls. *Journal of the American Dietetic Association, 100*(8), 911–918.

Beeler, B., & Lecomte, P. (2017). Shedding light on the darker side of language: A dialogical approach to cross-cultural collaboration. *International Journal of Cross Cultural Management, 17*(1), 53–67.

Bell, T., & Vahrenhold, J. (2018). *CS unplugged—how is it used, and does it work?. Adventures between lower bounds and higher altitudes: Essays dedicated to Juraj Hromkovič on the occasion of his 60th birthday*. Springer.

Biddy, Q., Chakarov, A. G., Bush, J., Elliott, C. H., Jacobs, J., Recker, M., & Penuel, W. (2021). A professional development model to integrate computational thinking into middle school science through codesigned storylines. *Contemporary Issues in Technology and Teacher Education, 21*(1), 53–96.

Biswas, I., & Johnson, M. C. (2015). A pedagogical framework to teach HW-SW co-design. In *ASEE IL/IN section conference*.

Bovill, C. (2020). Co-creation in learning and teaching: The case for a whole-class approach in higher education. *Higher Education, 79*(6), 1023–1037.

Bovill, C., Cook-Sather, A., Felten, P., Millard, L., & Moore-Cherry, N. (2016). Addressing potential challenges in co-creating learning and teaching: Overcoming resistance, navigating institutional norms and ensuring inclusivity in student–staff partnerships. *Higher Education, 71*, 195–208.

Brackmann, C. P., Román-González, M., Robles, G., Moreno-León, J., Casali, A., & Barone, D. (2017). Development of computational thinking skills through unplugged activities in primary school. In *Proceedings of the 12th workshop on primary and secondary computing education* (pp. 65–72).

Bravo-Palacios, G., Del Prete, A., & Wensing, P. M. (2020). One robot for many tasks: Versatile co-design through stochastic programming. *IEEE Robotics and Automation Letters, 5*(2), 1680–1687.

Breien, F., & Wasson, B. (2022). eLuna: A Co-design framework for narrative digital game-based learning that support steam. *In Frontiers in Education., 6*, 775746.

Caeli, E. N., & Yadav, A. (2020). Unplugged approaches to computational thinking: A historical perspective. *TechTrends, 64*(1), 29–36.

Calvo, M., & Sclater, M. (2021). Creating spaces for collaboration in community co-design. *International Journal of Art & Design Education, 40*(1), 232–250.

Caskurlu, S., Hu, A. D., Yadav, A., & Santo, R. (2022, March). Computational thinking integration design principles in humanities. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 2* (pp. 1120–1120).

Chartier, A. M., & Geneix, N. (2007). Pedagogical approaches to early childhood education. *Documento de referencia para el Informe de Seguimiento de la EPT en el Mundo*.

Cho, N., Shin, M., & Ahn, H. (2022). Psychosocial characters and their behavioural indexes for evaluation in secondary school physical education classes and sports club activities. *International Journal of Environmental Research and Public Health, 19*(11), 6730.

Chong, D., & Druckman, J. N. (2007). Framing public opinion in competitive democracies. *American Political Science Review, 101*(4), 637–655.

Coenraad, M., Burke, Q., Ruiz, P., Mills, K., & Roschelle, J. (2022, March). Developing inclusive computing with the CT pathways toolkit. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 2* (pp. 1089–1089).

Coimbra, C. L. (2020). Models of K-12 Teacher Training: Who do we train?. *Educação & Realidade*, *45*.

Collentine, J., & Freed, B. F. (2004). Learning context and its effects on second language acquisition: Introduction. *Studies in Second Language Acquisition, 26*(2), 153–171.

Collins, A., Brown, J. S., & Newman, S. E. (2018). *Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In Knowing, Learning, and Instruction*. Routledge.

Collins, E., & Green, J. L. (1992). *Learning in classroom settings: Making or breaking a culture. Redefining student learning: Roots of educational change*. Albex.

Curzon, P., McOwan, P. W., Plant, N., & Meagher, L. R. (2014). Introducing teachers to computational thinking using unplugged storytelling. In *Proceedings of the 9th Workshop in primary and secondary computing education* (pp. 89–92).

Daniels, E., Hondeghem, A., & Heystek, J. (2020). Exploring the outcomes of group reflective learning for school leaders. *Participatory Practice, 21*(5), 604–618.

Denning, P. J., & Tedre, M. (2021). Computational thinking: A disciplinary perspective. *Informatics in Education, 20*(3), 361.

Dickes, A. C., Farris, A. V., & Sengupta, P. (2020). Sociomathematical norms for integrating coding and modeling with elementary science: A dialogical approach. *Journal of Science Education and Technology, 29*, 35–52.

Dindler, C., Smith, R., & Iversen, O. S. (2020). Computational empowerment: participatory design in education. *CoDesign, 16*(1), 66-80.
Dong, Y., Catete, V., Jocius, R., Lytle, N., Barnes, T., Albert, J., & Andrews, A. (2019). PRADA: A practical model for integrating computational thinking in K-12 education. In *Proceedings of the 50th ACM technical symposium on computer science education* (pp. 906–912).
Durall, E., Bauters, M., Hietala, I., Leinonen, T., & Kapros, E. (2019). Co-creation and co-design in technology-enhanced learning: Innovating science learning outside the classroom. *Ixd&a, 42*, 202–226.
Echeverría, L., Cobos, R., & Morales, M. (2019). Improving the students computational thinking skills with collaborative learning techniques. *IEEE Revista Iberoamericana De Tecnologias Del Aprendizaje, 14*(4), 196–206.
Elliott, I. C., Robson, I., & Dudau, A. (2021). Building student engagement through co-production and curriculum co-design in public administration programmes. *Teaching Public Administration, 39*(3), 318–336.
Elmoazen, R., Saqr, M., Tedre, M., & Hirsto, L. (2022). A systematic literature review of empirical research on epistemic network analysis in education. *IEEE Access., 10*, 17330.
Elo, S., & Kyngäs, H. (2008). The qualitative content analysis process. *Journal of Advanced Nursing, 62*(1), 107–115.
García-Peñalvo, F. J., Reimann, D., Tuul, M., Rees, A. M., & Jormanainen, I. (2016). TACCLE 3, O5: An overview of the most relevant literature on coding and computational thinking with emphasis on the relevant issues for teachers. Belgium. https://doi.org/10.5281/zenodo,165123
Gendreau Chakarov, A., Recker, M., Jacobs, J., Van Horne, K., & Sumner, T. (2019, February). Designing a middle school science curriculum that integrates computational thinking and sensor technology. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education* (pp. 818–824).
Giner Sanz, J. J., García Gabaldón, M., Ortega Navarro, E. M., Shao Horn, Y., & Pérez Herranz, V. (2019). A NetLogo® model for introducing students to genetic algorithms. In *IN-RED 2019. V Congreso de Innovación Educativa y Docencia en Red* (pp. 88–101). Editorial Universitat Politècnica de València.
Glenn, J. C. (2003). Participatory Methods. *Futures Research Methodology, 2*, 1–32.
Green, S. L., & Anid, N. M. (2013). Training K-12 teachers in STEM education: A multi-disciplinary approach. In *2013 IEEE integrated STEM education conference (ISEC)* (pp. 1–4). IEEE.
Greenberg, J., & Walsh, K. (2012). What teacher preparation programs teach about K-12 assessment: A Review. *National council on teacher quality*.
Greenhalgh, T., & Peacock, R. (2005). Effectiveness and efficiency of search methods in systematic reviews of complex evidence: Audit of primary sources. *BMJ, 331*(7524), 1064–1065.
Guo Brennan, L. (2022). Making virtual global learning transformative and inclusive: A critical reflective study on high-impact practices in higher education. *Journal of Teaching and Learning, 16*(2), 28–49.
Hamilton, P. (2016). *The workshop book: how to design and lead successful workshops*. Pearson.
Hartt, M., Hosseini, H., & Mostafapour, M. (2020). Game on: Exploring the effectiveness of game-based learning. *Planning Practice & Research, 35*(5), 589–604.
Higgins, D., Dennis, A., Stoddard, A., Maier, A. G., & Howitt, S. (2019). 'Power to empower': Conceptions of teaching and learning in a pedagogical co-design partnership. *Higher Education Research & Development, 38*(6), 1154–1167.
Hsu, T. C., Chang, S. C., & Hung, Y. T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education, 126*, 296–310.
Huang, W., & Looi, C. K. (2021). A critical review of literature on "unplugged" pedagogies in K-12 computer science and computational thinking education. *Computer Science Education, 31*(1), 83–111.
Ito, M., Gutiérrez, K., Livingstone, S., Penuel, B., Rhodes, J., Salen, K., & Watkins, S. C. (2013). *Connected learning: An agenda for research and design*. Digital Media and Learning Research Hub.
Järvelä, S., Gašević, D., Seppänen, T., Pechenizkiy, M., & Kirschner, P. A. (2020). Bridging learning sciences, machine learning and affective computing for understanding cognition and affect in collaborative learning. *British Journal of Educational Technology, 51*(6), 2391–2406.
Jormanainen, I., & Tukiainen, M. (2020). Attractive educational robotics motivates younger students to learn programming and computational thinking. In *Eighth international conference on technological ecosystems for enhancing multiculturality* (pp. 54–60).
Kafai, Y., Proctor, C., & Lui, D. (2020). From theory bias to theory dialogue: Embracing cognitive, situated, and critical framings of computational thinking in K-12 CS education. *ACM Inroads, 11*(1), 44–53.
Kamdar, D., Grover, S., Vahey, P., Leones, T., & Dominguez, X. (2021). Computational thinking in preschool: Bridging home and School. In *Proceedings of the 15th International Conference of the Learning Sciences-ICLS 2021*. International Society of the Learning Sciences.

Karumbaiah, S., Dabholkar, S., Shim, J., Yoon, S., Chandy, B., & Ye, A. (2019). Using participatory design to facilitate in-service teacher learning of computational Thinking. In *Proceeding of 13th international conference on computer supported collaborative learning, CSCL.*

Kelly, N., Wright, N., Dawes, L., Kerr, J., & Robertson, A. (2019). Co-design for curriculum planning: A model for professional development for high school teachers. *Australian Journal of Teacher Education (online), 44*(7), 84–107.

Kelter, J., Peel, A., Bain, C., Anton, G., Dabholkar, S., Aslan, U., ... & Wilensky, U. (2020). Seeds of (r) Evolution: Constructionism Co-Design with High School Science Teachers. *Constructionism* 2020.

Kelter, J., Peel, A., Bain, C., Anton, G., Dabholkar, S., Horn, M. S., & Wilensky, U. (2021). Constructionist co-design: A dual approach to curriculum and professional development. *British Journal of Educational Technology, 52*(3), 1043–1059.

Kennedy, M. M. (2016). How does professional development improve teaching? *Review of Educational Research, 86*(4), 945–980.

Ketelhut, D. J., Mills, K., Hestness, E., Cabrera, L., Plane, J., & McGinnis, J. R. (2020). Teacher change following a professional development experience in integrating computational thinking into elementary science. *Journal of Science Education and Technology, 29*, 174–188.

Kin, J. H. Y., Yatim, M. H. M., Hoe, T. W., & Seng, W. Y. (2021). Developing a game-based learning assessment framework towards ubiquitous computational thinking among undergraduate students. *Journal of ICT in Education, 8*(4), 130–144.

Kirçali, A. Ç., & Özdener, N. (2023). A comparison of plugged and unplugged tools in teaching algorithms at the K-12 level for computational thinking skills. *Technology, Knowledge and Learning, 28*(4), 1485–1513.

Kitchenham, B., & Charters, S. (2007). *Guidelines for performing systematic literature reviews in* software *engineering* (Report No. EBSE-2007–01). Keele University & University of Durham.

Kong, S. C. (2016). A framework of curriculum design for computational thinking development in K-12 education. *Journal of Computers in Education, 3*(4), 377–394.

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

Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A pedagogical framework for computational thinking. *Digital Experiences in Mathematics Education, 3*, 154–171.

Kyza, E. A., & Agesilaou, A. (2022). Investigating the processes of teacher and researcher empowerment and learning in co-design settings. *Cognition and Instruction, 40*(1), 100–125.

Laato, S., & Pope, N. (2019). A lightweight co-construction activity for teaching 21st century skills at primary schools. In *Proceedings of the* 52nd *Hawaii international conference on system sciences* https://hdl.handle.net/10125/60214

Lahann, P., & Lambdin, D. V. (2020). Collaborative learning in mathematics education. *Encyclopedia of Mathematics Education*. https://doi.org/10.1007/978-3-030-15789-0_23

Lehtimäki, T., Monahan, R., Mooney, A., Casey, K., & Naughton, T. J. (2022). Bebras-inspired computational thinking primary school resources co-created by computer science academics and teachers. In *Proceedings of the 27th ACM conference on innovation and technology in computer science education Vol. 1* (pp. 207–213).

Li, Y. (2016). Teaching programming based on computational thinking. In *2016 IEEE frontiers in education conference (FIE)* (pp. 1–7). IEEE.

Lipscomb, S. (2007). Secondary school extracurricular involvement and academic achievement: A fixed effects approach. *Economics of Education Review, 26*(4), 463–472.

Lunding, M. S., Grønbæk, J. E. S., Bilstrup, K. E. K., Sørensen, M. L. S. K., & Petersen, M. G. (2022). Exposar: Bringing augmented reality to the computational thinking agenda through a collaborative authoring tool. In *Proceedings of the 2022 CHI conference on human factors in computing systems* (pp. 1–14).

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior, 41*, 51–61.

Lytvyn, V., Akimova, O., Kuznetsova, H., Zenchenko, T., Stepanenko, O., & Koreneva, I. (2021). The use of synchronous and asynchronous teaching methods in pedagogical education in COVID-19 terms. *International Journal of Health Sciences, 5*(3), 617–629.

Malkin, A., Rehfeldt, R. A., & Shayter, A. M. (2018). An investigation of the efficacy of asynchronous discussion on students' performance in an online research method course. *Behavior Analysis in Practice, 11*, 274–278.

Matsuo, M., & Tsukube, T. (2020). A review on cognitive apprenticeship in educational research: Application for management education. *The International Journal of Management Education, 18*(3), 100417.
Matutino, P. M., Dias, T., & Sampaio, P. (2020). Teaching hardware/software co-design using a project-based learning strategy. In *2020 XIV technologies applied to electronics teaching conference (TAEE)* (pp. 1–6). IEEE.
Van Mechelen, M., Schut, A., Gielen, M., & Klapwijk, R. (2018). Developing children's empathy in co-design activities: A pilot case study. In *Proceedings of the 17th ACM conference on interaction design and children* (pp. 669–674).
Van Mechelen, M., Schut, A., Gielen, M., & Södergren, A. C. (2019). Children's assessment of co-design skills: creativity, empathy and collaboration. In *Proceedings of the 18th ACM international conference on interaction design and children* (pp. 520–526).
Mills, K., Angevine, C., & Weisgrau, J. (2020). Resources for computational thinking: Co-designing with teachers. In *Proceedings of the* 51st *ACM technical symposium on computer science education* (pp. 1343–1343).
Monjelat, N., & Lantz-Andersson, A. (2020). Teachers' narrative of learning to program in a professional development effort and the relation to the rhetoric of computational thinking. *Education and Information Technologies, 25*(3), 2175–2200.
Morales-Navarro, L., Thompson, N., Kafai, Y., Shaw, M., & Pinkard, N. (2022). Reimagining and co-designing with youth an hour of code activity for critical engagement with computing. In *Proceedings of the 21st annual ACM interaction design and children conference* (pp. 288–296).
Moser, A., & Korstjens, I. (2022). Series: Practical guidance to qualitative research Part 5: Co-creative qualitative approaches for emerging themes in primary care research: Experience-based co-design, user-centred design and community-based participatory research. *European Journal of General Practice, 28*(1), 1–12.
Moser, S. C. (2016). Can science on transformation transform science? Lessons from co-design. *Current Opinion in Environmental Sustainability, 20*, 106–115.
Moudgalya, S. K., Yadav, A., Sands, P., Vogel, S., & Zamansky, M. (2021). Teacher views on computational thinking as a pathway to computer science. In *Proceedings of the 26th ACM conference on innovation and technology in computer science education V. 1* (pp. 262–268).
Niazi, M. (2015). Do systematic literature reviews outperform informal literature reviews in the software engineering domain? An initial case study. *Arabian Journal for Science and Engineering, 40*, 845–855.
Nicholson, R., Bartindale, T., Kharrufa, A., Kirk, D., & Walker-Gleaves, C. (2022). Participatory design goes to school: Co-teaching as a form of co-design for educational technology. In *Proceedings of the 2022 CHI conference on human factors in computing systems* (pp. 1–17).
Nombela, R., & von Wangenheim, C. G. (2018). On tools that support the development of computational thinking skills: Some thoughts and future vision. *Copyright 2018 The Hong Kong Jockey Club All rights reserved. ISBN: 978–988–77034–5–7*, 129.
Olesen, A. R., Holdgaard, N., & Løvlie, A. S. (2022). Co-designing a co-design tool to strengthen ideation in digital experience design at museums. *CoDesign, 18*(2), 227–242.
Oyelere, A. S., Agbo, F. J., & Oyelere, S. S. (2023). Formative evaluation of immersive virtual reality expedition mini-games to facilitate computational thinking. *Computers & Education: X Reality, 2*, 100016.
Oyelere, S. S., Bouali, N., Kaliisa, R., Obaido, G., Yunusa, A. A., & Jimoh, E. R. (2020). Exploring the trends of educational virtual reality games: A systematic review of empirical studies. *Smart Learning Environments, 7*, 1–22.
Palmatier, R. W., Houston, M. B., & Hulland, J. (2018). Review articles: Purpose, process, and structure. *Journal of the Academy of Marketing Science, 46*, 1–5.
Papadakis, S. (2021). The impact of coding apps to support young children in computational thinking and computational fluency. A literature review. *In Frontiers in Education, 6*, 657895.
Papert, S. (1980). *Children, computers, and powerful ideas*. Harvester.
Papert, S., & Harel, I. (1991). Situating constructionism. *Constructionism, 36*(2), 1–11.
Park, J. J., & Schallert, D. L. (2019). Talking, reading, and writing like an educational psychologist: The role of discourse practices in graduate students' professional identity development. *Learning, Culture and Social Interaction, 22*, 100243.
Peel, A., Dabholkar, S., Anton, G., Wu, S., Wilensky, U., & Horn, M. (2020). A case study of teacher professional growth through co-design and implementation of computationally enriched biology units. In *Proceedings of the 14th international conference of the learning sciences, ICLS, 2020* (pp 1950–1957).

Peel, A., Dabholkar, S., Anton, G., Horn, M., & Wilensky, U. (2023). Characterizing changes in teacher practice and values through co-design and implementation of computational thinking integrated biology units. *Computer Science Education*. https://doi.org/10.1080/08993408.2023.2265763

Penuel, W. R., Fishman, B. J., Haugan Cheng, B., & Sabelli, N. (2011). Organizing research and development at the intersection of learning, implementation, and design. *Educational Researcher, 40*(7), 331–337.

Penuel, W. R., Roschelle, J., & Shechtman, N. (2007). Designing formative assessment software with teachers: An analysis of the co-design process. *Research and Practice in Technology Enhanced Learning, 2*(01), 51–74.

Phuan, N. H. Y., Lee, C. S., & Ean-Huat, O. O. I. (2020). *CT-based collaborative storytelling for learning programming concepts in Python*. CoolThink@JC.

Piedade, J., Dorotea, N., Pedro, A., & Matos, J. F. (2020). On teaching programming fundamentals and computational thinking with educational robotics: A didactic experience with pre-service teachers. *Education Sciences, 10*(9), 214.

Priya, R. S., Shabitha, P., & Radhakrishnan, S. (2020). Collaborative and participatory design approach in architectural design studios. *Social Sciences & Humanities Open, 2*(1), 100033.

Radu, I., Yuan, J., Huang, X., & Schneider, B. (2023). Charting opportunities and guidelines for augmented reality in makerspaces through prototyping and co-design research. *Computers & Education: X Reality, 2*, 100008.

Real, E., Liang, C., So, D., & Le, Q. (2020). Automl-zero: Evolving machine learning algorithms from scratch. In *International* conference *on machine learning* (pp. 8007–8019). PMLR.

Resnick, M. (2017). *Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play*. MIT press.

Reynolds-Cuéllar, P., & Delgado Ramos, D. (2020). Community-based technology co-design: Insights on participation, and the value of the "co". In *Proceedings of the 16th participatory design conference 2020-participation (s)* otherwise-*Volume 1* (pp. 75–84).

Rich, K. M., Yadav, A., & Larimore, R. A. (2020). Teacher implementation profiles for integrating computational thinking into elementary mathematics and science instruction. *Education and Information Technologies, 25*, 3161–3188.

Riikonen, S., Seitamaa-Hakkarainen, P., & Hakkarainen, K. (2018). Bringing practices of co-design and making to basic education. Rethinking learning in the digital age. In *Proceeding of* 13th international conference of the learning sciences (ICLS), (pp. 248–255)

Roschelle, J., & Penuel, W. R. (2006). Co-design of innovations with teachers: Definition and dynamics. In *Proceedings of the 7th international conference on learning sciences* (pp. 606–612).

Sanders, E. B. N., & Stappers, P. J. (2008). Co-creation and the new landscape of design. *Co-Design, 4*(1), 5–18.

Sanders, E. B. N., & Stappers, P. J. (2014). Probes, toolkits and prototypes: Three approaches to making in codesigning. *CoDesign, 10*(1), 5–14.

Sands, P., Yadav, A., & Good, J. (2018). Computational thinking in K-12: In-service teacher perceptions of computational thinking. *Computational thinking in the STEM disciplines: Foundations and research highlights*, 151–164.

Sanusi, I. T., & Oyelere, S. S. (2020). Pedagogies of machine learning in K-12 context. In *2020 IEEE frontiers in education conference (FIE)* (pp. 1–8). IEEE.

Sanusi, I. T., Oyelere, S. S., Agbo, F. J., & Suhonen, J. (2021). Survey of resources for introducing machine learning in K-12 context. In *2021 IEEE frontiers in education conference (FIE)* (pp. 1–9). IEEE.

Sanusi, I. T., Omidiora, J. O., Oyelere, S. S., Vartiainen, H., Suhonen, J., & Tukiainen, M. (2023). Preparing middle schoolers for a machine learning-enabled future through design-oriented pedagogy. *IEEE Access, 11*, 39776–39791.

Sanusi, I. T., Oyelere, S. S., Vartiainen, H., Suhonen, J., & Tukiainen, M. (2023). A systematic review of teaching and learning machine learning in K-12 education. *Education and Information Technologies, 28*(5), 5967–5997.

Sanusi, I. T., Oyelere, S. S., Vartiainen, H., Suhonen, J., & Tukiainen, M. (2022). A systematic review of teaching and learning machine learning in K-12 education. *Education and Information Technologies, 28*(5), 5967–5997.

Sathasivam, S., & Fen, N. P. (2013). Developing agent based modeling for doing logic programming in hopfield network. *Applied Mathematical Sciences, 7*(1), 23–35.

Saxena, A., Lo, C. K., Hew, K. F., & Wong, G. K. W. (2020). Designing unplugged and plugged activities to cultivate computational thinking: An exploratory study in early childhood education. *The Asia-Pacific Education Researcher, 29*(1), 55–66.

Schaper, M. M., & Pares, N. (2021). Co-design techniques for and with children based on physical theatre practice to promote embodied awareness. *ACM Transactions on Computer-Human Interaction (TOCHI), 28*(4), 1–42.
Scheufele, D. A., & Tewksbury, D. (2007). Framing, agenda setting, and priming: The evolution of three media effects models. *Journal of Communication, 57*(1), 9–20.
Schmidt, J. P., Resnick, M., & Ito, J. (2016). *Creative learning and the future of work. Disrupting unemployment*. Springer.
Seitamaa-Hakkarainen, P., Kangas, K., Raunio, A. M., & Hakkarainen, K. (2012). Collaborative design practices in technology mediated learning. *The Journal of Design and Technology Education, 17*(1), 54–65.
Seitamaa-Hakkarainen, P., Viilo, M., & Hakkarainen, K. (2010). Learning by collaborative designing: Technology-enhanced knowledge practices. *International Journal of Technology and Design Education, 20*, 109–136.
Sengupta, P., Dickes, A., Farris, A. V., Karan, A., Martin, D., & Wright, M. (2015). Programming in K-12 science classrooms. *Communications of the ACM, 58*(11), 33–35.
Severance, S., Penuel, W. R., Sumner, T., & Leary, H. (2016). Organizing for teacher agency in curricular co-design. *Journal of the Learning Sciences, 25*(4), 531–564. https://doi.org/10.1080/10508406.2016.1207541
Shin, N., Bowers, J., Krajcik, J., & Damelin, D. (2021). Promoting computational thinking through project-based learning. *Disciplinary and Interdisciplinary Science Education Research, 3*(1), 1–15.
Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying Computational Thinking. *Educational Research Review, 22*, 142–158.
Ssozi-Mugarura, F., Blake, E., & Rivett, U. (2016). Supporting community needs for rural water management through community-based co-design. In *Proceedings of the 14th participatory design conference: Full papers-Volume 1* (pp. 91–100).
Su, J., & Yang, W. (2022). Artificial intelligence in early childhood education: A scoping review. *Computers and Education: Artificial Intelligence, 3*, 100049.
Sunday, A. O. (2023). Design and implementation of co-design pedagogical scenarios for learning computational thinking. In *Proceedings of the 2023 conference on innovation and technology in computer science education V. 2* (pp. 609–610).
Tedre, M., Toivonen, T., Kahila, J., Vartiainen, H., Valtonen, T., Jormanainen, I., & Pears, A. (2021). Teaching machine learning in K–12 classroom: Pedagogical and technological trajectories for artificial intelligence education. *IEEE Access, 9*, 110558–110572.
Tikva, C., & Tambouris, E. (2021). Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature review. *Computers & Education, 162*, 104083.
Toivonen, T., Jormanainen, I., Kahila, J., Tedre, M., Valtonen, T., & Vartiainen, H. (2020). Co-designing machine learning apps in K–12 with primary school children. In *2020 IEEE 20th international conference on advanced learning technologies (ICALT)* (pp. 308–310). IEEE.
Turchi, T., Fogli, D., & Malizia, A. (2019). Fostering computational thinking through collaborative game-based learning. *Multimedia Tools and Applications, 78*, 13649–13673.
Vallance, M., & Towndrow, P. A. (2016). Pedagogic transformation, student-directed design and computational thinking. *Pedagogies: an International Journal, 11*(3), 218–234.
Van Brummelen, J., & Lin, P. (2020). Engaging teachers to co-design integrated AI curriculum for K-12 classrooms. *arXiv preprint* arXiv:2009.11100.
Vartiainen, H. (2014). *Principles for design-oriented pedagogy for learning from and with museum objects* [Doctoral dissertation, Itä-Suomen yliopisto].
Vartiainen, H., Toivonen, T., Jormanainen, I., Kahila, J., Tedre, M., & Valtonen, T. (2020). Machine learning for middle-schoolers: Children as designers of machine-learning apps. In *2020 IEEE frontiers in education conference (FIE)* (pp. 1–9). IEEE.
Vartiainen, H. (2022). Cross-boundary co-design for learning machine learning. *AI, data science, and young people. Understanding computing education*.
Vartiainen, H., Liljeström, A., & Enkenberg, J. (2012). Design-oriented pedagogy for technology-enhanced learning to cross over the borders between formal and informal environments. *Journal of Universal Computer Science, 18*(15), 2097–2119.
Vázquez, A. S., Calvo, T., Fernández, R., & Ramos, F. (2021). A visual programming approach for co-designed robots. *Robotica, 39*(6), 1116–1139.
Voigt, C., Unterfrauner, E., Aslan, T., & Hofer, M. (2019). Design thinking with children: The role of empathy, creativity and self-efficacy. In *Proceedings of FabLearn 2019* (pp. 144–147).

Voon, X. P., Wong, S. L., Wong, L. H., Khambari, M. N. M., & Syed-Abdullah, S. I. S. (2023). Developing pre-service teachers' computational thinking through experiential learning: Hybridisation of plugged and unplugged approaches. *Research and Practice in Technology Enhanced Learning, 18*, 006–006.

WaiShiang, C., Nissom, S., Fong, S. S., & Khairuddin, M. A. (2017). Chemistry modelling and simulation through agent oriented modelling and Netlogo. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 9*(2–10), 145–149.

Waterman, K. P., Goldsmith, L., & Pasquale, M. (2020). Integrating computational thinking into elementary science curriculum: An examination of activities that support students' computational thinking in the service of disciplinary learning. *Journal of Science Education and Technology, 29*(1), 53–64.

Weintrop, D., Holbert, N., Horn, M. S., & Wilensky, U. (2016). Computational thinking in constructionist video games. *International Journal of Game-Based Learning (IJGBL), 6*(1), 1–17.

Weng, X., Ye, H., Dai, Y., & Ng, O. L. (2024). Integrating artificial intelligence and computational thinking in educational contexts: A systematic review of instructional design and student learning outcomes. *Journal of Educational Computing Research*. https://doi.org/10.1177/07356331241248686

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35.

Wu, B., Hu, Y., Ruis, A. R., & Wang, M. (2019). Analysing computational thinking in collaborative programming: A quantitative ethnography approach. *Journal of Computer Assisted Learning, 35*(3), 421–434.

Wu, S., Peel, A., Bain, C., Anton, G., Horn, M., & Wilensky, U. (2020). Workshops and co-design can help teachers integrate computational thinking into their K-12 stem classes. In *Proceedings of international conference on computational thinking education 2020*.

Wu, S. P., Peel, A., Zhao, L., Horn, M., & Wilensky, U. (2022). A professional development that helps teachers integrate computational thinking into their science classrooms through codesign. *Innovations, 7*(2), 1–11.

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends, 60*(6), 565–568.

Yang, D., Baek, Y., Ching, Y. H., Swanson, S., Chittoori, B., & Wang, S. (2021). Infusing computational thinking in an integrated STEM curriculum: User reactions and lessons learned. *European Journal of STEM Education, 6*(1), 4.

Yoo, D., Huldtgren, A., Woelfer, J. P., Hendry, D. G., & Friedman, B. (2013). A value sensitive action-reflection model: Evolving a co-design space with stakeholder and designer prompts. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 419–428).

Yoon, S. A., Miller, K., Richman, T., Wendel, D., Schoenfeld, I., Anderson, E., & Shim, J. (2020). Encouraging collaboration and building community in online asynchronous professional development: Designing for social capital. *International Journal of Computer-Supported Collaborative Learning, 15*, 351–371.

Zeregal, R., Hamidi, A., Tavajoh, S., & Milrad, M. (2021). A co-design approach for developing computational thinking skills in connection to STEM related curriculum in Swedish schools. In *The Fifth APSCE international conference on computational thinking and STEM education 2021; Singapore* (pp. 144–147). Asia-Pacific Society for Computers in Education.

Zhang, K., Chang, Y., Chen, M., Bao, Y., & Xu, Z. (2019). Computer organization and design course with FPGA cloud. In *Proceedings of the 50th ACM technical symposium on computer science education* (pp. 927–933).

Zhao, F., & Liu, X. (2022). From mutual creation to mutual benefit: China's national teacher training program between higher teacher education and K-12 teachers. *Journal of Contemporary Educational Research, 6*(9), 65–75.