Working Group Roundtable Title: Computational Thinking in Early Childhood: The State of Our Burgeoning Field

Abstract

While a number of researchers, curriculum designers, app developers, and other practitioners have made computational thinking (CT) a priority for learning during early childhood, we are still generating evidence as to how this early experience can support children's learning and development later in life. This roundtable workgroup is intended to foster a discussion of what we know about the development of CT in children, a brainstorm of gaps in our knowledge and the extent to which new projects are addressing those gaps, and a collaborative thinking session to identify critical next steps to further investigate the extent to which early experience supports CT later in life.

Session Summary

In the last several years, there has been a call to integrate CT into formal and informal learning environments for young children. Referenced as a critical life skill, the International Society of Technology in Education has released a set of Computational Thinking Competencies (ISTE, 2018) to support the incorporation of CT across the curriculum. Introducing CT into preschool learning in both formal and informal spaces has the potential to increase children's development across multiple domains of school readiness. Despite the growing consensus that early childhood is an ideal period for developing CT skills, our field is still in its infancy in synthesizing evidence as to how and why early childhood is the true "sweet spot" in the development of these skills. While many projects have researched the best instructional methods and experiences to support CT for young children and how early experiences are posited to support proficiency with coding and computer science in later childhood, our field is still seeking evidence to characterize the true value of these early activities.

The objective of this session is to convene practitioners and researchers to discuss the state of our field. Specifically, we would address the following agenda points during the session:

1. Brief orientation: what is recognized as consensus in our field as to the value of CT during early childhood (30 minutes- three papers)
2. Acknowledging diverging perspectives and approaches to how CT is fostered in preschool children: knowledge, skills, practices (20 minutes – two papers)
3. Group brainstorm & share out: what are the gaps in our knowledge and where do we lack developmental knowledge and evidence? (25 minutes)
4. Discussion: in what ways are we starting to do research or develop new content around some of those gaps? For any gaps that aren't being addressed, what is the pathway forward? (15 minutes)

Participants are asked to come to the working group with some experience either in the research of CT or in the development of learning resources. The session will be largely interactive with participants engaging in a discussion around how our collective work is creating new evidence for the field and working towards a deeper understanding of how to support CT in children.

Paper 1: Computational Thinking (CT) Meets Young Children: Critical Review of Research on CT in Early Childhood

Viewed as the foundational processes of computer science with broader application for learning, computational thinking (CT) is increasingly promoted and researched in early childhood (EC; ages 2-8) (Bers, 2018). In contrast to secondary education (the site of most current CT pedagogy and research), EC education prioritizes development of the whole child through integrated interdisciplinary pedagogy. There is a need for an informed debate about benefits and tensions of introducing CT in EC (Manches & Plowman, 2017). To address the need, we conducted a systematic review guided by the following questions: (1) What are the general characteristics of the existing studies on CT in EC? (2) How is CT defined and operationalized in EC? (3) How is CT promoted and supported in EC?

Following Alexander's (2020) guidance for systematic reviews, we employed a three-step process to select studies. First, we developed a keyword search using 35 articles identified by domain experts as an initial validation set. The final query was: ("computational thinking" OR "Robotics" or "Unplugged" or "Computing Education" or "Computer Science Education") AND ("children" or "kindergarten" or "early childhood" or "early years"), published between 2006 and 2021. Second, we ran a keyword search across five databases (ERIC, ScienceDirect, SpringerLink, IEEE, and ACM) and yielded 8398 unique results. Third, we filtered these with a Bayesian classifier trained on article abstracts and hand-coded the results, producing a corpus of 125 studies. To analyze the selected studies, (1) we used content analysis to summarize characteristics of the selected studies' research questions, research design, theoretical framework, and participants' demographics; (2) we categorized how CT was defined, operationalized, or measured; and (3) we coded CT pedagogies including activity types, duration, teaching methods, technical tools, and effectiveness.

Our preliminary results reveal: (1) there is a lack of diversity in study populations, in research agendas (with a dominant focus on feasibility and effectiveness of CT activities), and in theoretical frameworks (constructionism or sociocultural theories dominate); (2) these studies narrowly define or operationalize CT as a set of cognitive concepts and skills; and (3) developmentally appropriate practice (DAP) in CT learning context is employed superficially. Programming is the dominant context for CT learning.

Situating the review results in the fields of early childhood education as well as K12 computing education (Kafai et al., 2019; Tissenbaum et al., 2021), we propose ways to diversify the research agenda and study populations, to broaden definitions of CT to better align with the idea of educating a whole child in EC while preparing children for future CT learning, and to expand the pedagogical repertoire with practices consistent with evidence from EC as well as K12 computing education.

Going beyond synthesizing research on CT in EC, this paper offers theoretical suggestions on how to better-align conceptualizations of CT for early childhood, as well as practical guidance for teachers and researchers on teaching developmentally appropriate CT. Finally, we suggest ways to diversify research agendas and study populations to address equity and inclusion.

Paper 2: Coding in Kindergarten with Screen-Free Tangible Robot Coding Toys

Coding provides a playful context for young children to engage in computational thinking (Bers, 2020). By nature, computational thinking (CT) is situative, and many frameworks developed to explain CT are situated in particular contexts. For example, Brennan and Resnick's (2012) framework is situated in Scratch and Weintrop et al.'s (2016) is situated in STEM. In this paper, we discuss how CT operationalized for early childhood is situated in both the context of the coding environment as well as the developmental level of the children. While there are many coding options available for kindergarten classrooms, this paper focuses on the context of screen-free tangible robot coding toys. Specifically, Botley, Cubetto, Robot Mouse, Bee Bot and Code-a-pilar (see table 1).

Algorithmic thinking, debugging, and decomposition are recognized as important CT skills in early childhood (see figure 1). In addition to these CT skills, tangible coding toys involve spatial skills as children program an agent to navigate a physical space with directional arrows as the primary codes (forward, back, rotate left, rotate right). Further, kindergarten is often a period of rapid developmental and cognitive growth, where numeracy and literacy skills are emergent and involve making meaning of symbols (i.e., numerals and letters). The coding context of the toys in combination with children's developing skills necessitate a broader view of CT that accounts for symbolic-type contextual proficiencies that kindergarten-aged children must have to operate the toys: knowledge of 1-to-1 code-to-movement correspondence; semantic knowledge of what the codes mean; knowledge that the orientation of the robot determines which code to choose; and knowledge of coordinating the path with the program. Similarly, we found that mathematical knowledge influences their performance on CT tasks, including: counting-on; rotation on a point; dynamic linear units of measure; sequencing; and spatial reasoning (Authors, under review). Context proficiencies and mathematical knowledge develop in parallel with their knowledge of the CT skills while also influencing their performance on CT tasks. These CT skills, contextual proficiencies, and mathematical knowledge are similar across these toys and could also apply to some boardgames designed to teach young children CT (see Authors, 2021).

It is important to note that other coding contexts and toys have the potential to foster a wider range of CT skills such as repeat loops and math skills such as patterning, unitizing, and more advanced iterating. Similarly, using tangible coding toys with children in first grade who have a better understanding of unitizing would also require an adjustment of the contextual proficiencies and mathematical knowledge. In conclusion, coding with tangible coding toys in early childhood provide a context for children to develop CT, explore math concepts more deeply, and build connections to emergent literacy and mathematical skills. Some potential areas to explore are: Is there a hypothetical learning trajectory of CT? How would the model change across age spans (preK – second grade)? How do we expand our CT model to integrate equity and critical theories?

Paper 3: The Promise of Integrating CT into Literacy and Mathematics Instruction for Preschool Children

A number of studies have shown the value of CT for supporting students' academic performance, problem-solving skills, engagement in learning, as well as confidence and motivation (e.g., Rich et al., 2017, Yadav et al., 2017). Our team defines computational thinking (CT) as a creative way of thinking that empowers children to be systematic problem-solvers, enabling them to identify problems and then brainstorm and generate step-by-step solutions that can be communicated and followed by computers or humans. Across several NSF-funded projects, we have explored how to support CT skills in preschool children by integrating CT skills and practices into everyday learning opportunities, in particular during literacy learning and mathematics instruction. We posit that, with teacher support, children will begin to understand CT Core Ideas (figure 2) and how they can be leveraged to support problem solving or achieving goals in all areas of learning.

As part of our commentary, we will discuss our approach to developing learning resources, including the model that we have used to describe the CT integration process (Authors, 2020; figure 3). During mathematics instruction, children can leverage their emerging mathematical skills to explore CT concepts like patterns, problem solving, and begin to develop CT mindsets like curiosity, flexibility, inventiveness, and task persistence (National Association for the Education of Young Children & National Council of Teachers of Mathematics, 2002). We will also discuss our exploration of narrative storytelling as a framework for supporting computational thinking and computational literacy. As part of their narrative storytelling instruction, teachers commonly break down stories into parts, or what we call story elements (e.g., characters, actions, and consequences). This type of decomposition can help children understand the structure of stories, see the logical connections among elements, and create a coherent story by arranging (or combining) a series of characters, actions, events, and consequences in meaningful ways.

As part of our commentary, we will share these reflections from our research and development processes as they relate to how CT can be integrated into existing teaching and learning during preschool. Our contribution to this working group roundtable will reflect our lessons learned and evidence of promise from our work, as well as that from others, on how integration into the preschool curriculum presents real opportunities for CT in early childhood, including future research.

Paper 4: Diverse Perspectives on Including Data Science as a CT Skill for Young Children

A set of converging educational priorities has prompted the need to integrate computational thinking (CT) into STEM disciplines. As a distal goal, the need for highly skilled workers in STEM fields that include a 21st century data-capable workforce (Big Ideas for Future NSF Investment, NSF 2019), has led to the growing interest in computational thinking (Apone et al., 2005; Israel, Pearson, Tapia, Wherfel, & Reese, 2015), spurring recent efforts to prepare students from the early grades through high school by developing curricula to foster CT in preK–12 education (NSF, STEM+C solicitation).

Although definitions for CT vary, in essence it is a problem-solving process requiring the use of data. ISTE and CSTA (2011) describe CT as formulating problems in such a way that enables the use of technology to help solve them, collecting, organizing and analyzing data logically, and representing data through models and simulations. Specifically, we contend that one approach to supporting the development of CT in preK includes asking questions and investigating the answers through collecting, organizing, representing, and analyzing data with the goal of efficiently addressing real-world problems (ISTE & CSTA, 2011; Barr & Stephenson, 2011).

States and districts have begun to include standards for 4- and 5-year-olds that relate to data collection and analysis (DCA), CT, and general problem solving (e.g., NYS Prekindergarten), so there is an urgent need to ensure that teachers are able to facilitate learning in these domains in a developmentally appropriate and fun way for young children. Yet, very little research exists on explicitly integrating CT into data collection and analysis (DCA) in preK or in the early grades.

This paper will discuss what is known about data collection and analysis learning in the early years and describe how it overlaps and integrates with CT, mathematics, and science. Within this larger discussion, we will describe an exploratory project that develops a DCA intervention. The intervention provides opportunities for preschoolers to leverage their growing mathematical knowledge (i.e., counting, sorting, classifying, comparing and contrasting) to engage in investigations that are hands-on and play-based.

A key component is a tablet-based, teacher-facing digital app that supports the collaboration of preschool teachers and children in collecting data, creating simple graphs, and using the graphs to answer authentic questions. The app scaffolds this process by supporting teachers as they moved through specific DCA steps, such as identifying research questions, collecting data, creating simple representations (i.e. graphs and tally charts), and discussing and interpreting the graphs to answer questions. Investigations include curricular investigations, scaffolded teacher-generated investigations, and an emergent theme-based investigation that results in the creation of a short, narrative story about the students' research question and DCA process. Data from pilot studies will describe what we have learned about the intervention's developmental appropriateness, feasibility, and identify what supports teachers need to implement DCA activities, and how this might apply more broadly.

Overall, we will situate the inclusion of data sciences within the umbrella of CT and describe the diverse perspectives related to how and whether data science is included as a CT skill for young children.

Paper 5: Contrasting Practices in Computational Thinking Special Education

Over seven million students in the United States public school system are served by the special education system, including over 700,000 children between the ages of three and five (National Center for Education Statistics, 2021). These students are legally entitled to the educational opportunities and programs offered to their non-disabled peers, but as the computational thinking field develops best practices for early childhood CT education, best practices for early childhood CT special education are not yet developed. This paper will discuss contrasting current practices for including young children with disabilities in CT classrooms and suggest future directions for research.

As multiple approaches to special education exist in research and practice, there is equal variation in approaches to computational thinking education for students with disabilities. One approach refers to explicit instruction, a method of instruction derived from behaviorism(Knight, Wright, & DeFreese, 2019). These practices are very precise, teaching specific skills as behaviors and sequences of behaviors, and are commonly used for students with autism and intellectual disabilities. These practices have been used to teach coding, robotics, and computational thinking to students with moderate and severe disabilities in substantially separated special education classroom (Knight, Wright, & DeFreese, 2019; Knight, Wright, Wilson, et al., 2019; Taylor, 2018). Another common approach is Universal Design for Learning, a process for planning and developing curricula in order to accommodate students with a range of disabilities in inclusive settings (Capp, 2017). Where EBP-based curricula are precise and prescribed, UDL promotes the use of multiple approaches to present knowledge, engage students, and assess learning. Research by Maya Israel has examined use of UDL by teachers and mechanisms by which teachers can incorporate UDL practices into their CT classrooms (Israel et al., 2020).

As a field, there is a lack of knowledge as to how students with disabilities learn CT but there is reason to believe these processes are different for students with different disabilities. Young children in the special education system have a range of disabilities, including language delays, autism, learning disabilities, and intellectual disabilities (National Center for Education Statistics, 2021). Some disabilities impact math learning, language processing, or working memory, which may impact computational thinking learning, but we do not know enough about learning processes to make this assessment. As students have a variety of impairments and access needs, one approach to CT learning may not equally serve students with all disabilities. For this reason, researchers should aim to understand how students with different disabilities learn CT to develop best practices for the individual needs of students. As future steps, the authors of this paper will be differentially modifying a computational thinking curriculum to serve students with mild and moderate to severe disabilities in both inclusive and substantially separate classrooms, as well as examining outcomes for these students. Future work should examine how CT assessments can be modified and adapted to accurately assess the CT knowledge of students with different disabilities.

Figure 1: Early Childhood CT Competency Model. Components and sub-components, context proficiencies, and math knowledge are interrelated and influence children's development of CT competency and performance on CT assessments.

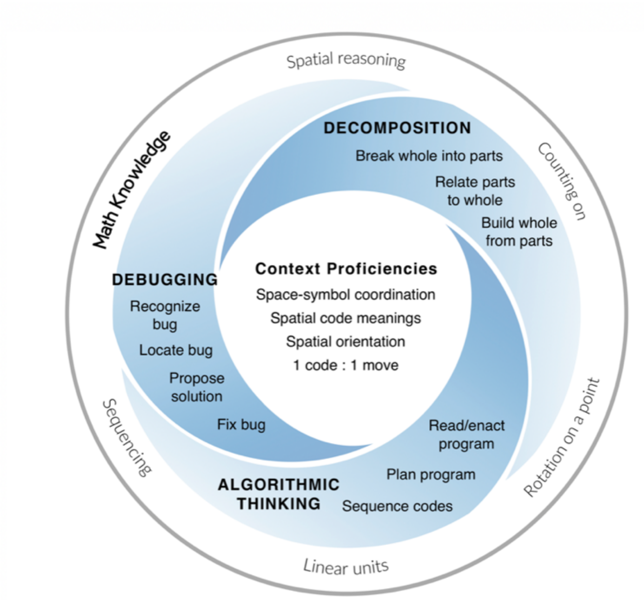


Figure 2. CT Core Ideas for Preschool Children

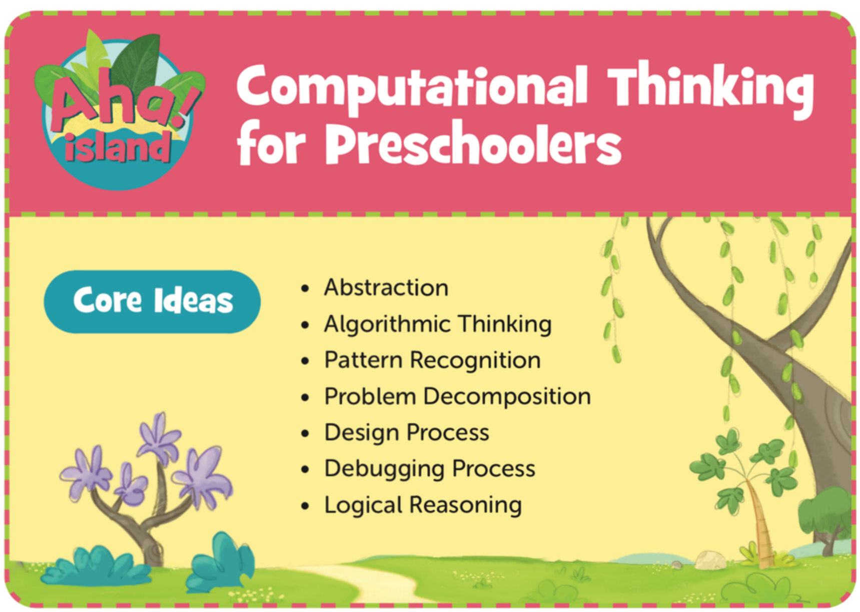


Figure 3. Theoretical model for the integration of CT into preschool learning within mathematics

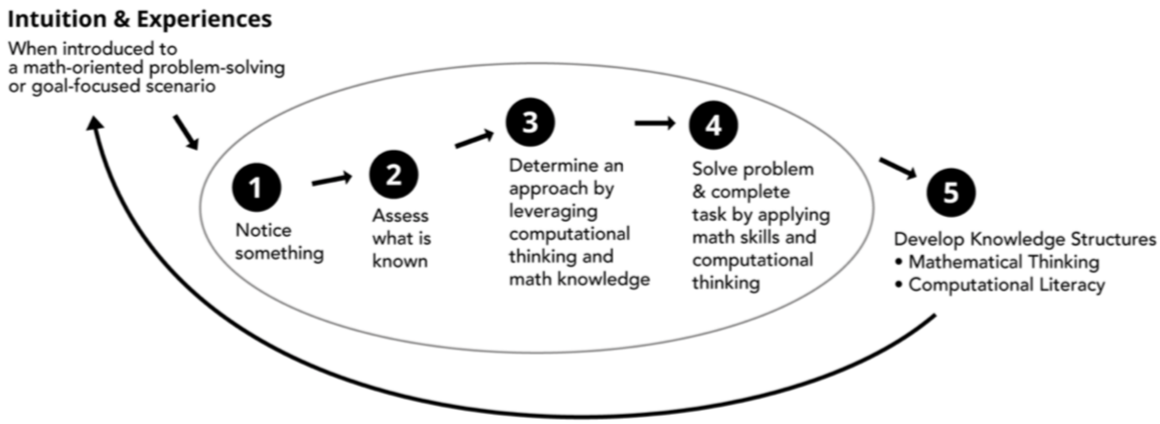


Table 1. Robot Coding Toys

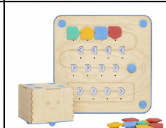
	BeeBot	Robot Mouse	Code-a-Pillar	Botley	Cubetto
Robot					
Manufacturer	Terrapin	Learning Resources	Fisher Price	Learning Resources	Primo Toys
Movements	forward rotate right 90° rotate left 90° back	forward rotate right 90° rotate left 90° back	forward turn right turn left	forward rotate right 90° rotate left 90° back	forward rotate right 90° rotate left 90° back
Images of movements (syntax)					

Table 1: Robot Coding Toys

References

Alexander, P. A. (2020). Methodological guidance paper: The art and science of quality systematic reviews. *Review of Educational Research, 90*(1), 6-23. https://doi.org/10.3102/0034654319854352

Apone, K., Bers, M., Brennan, K., Franklin, D., Israel, M., & Yongpradit, P. (2005). Bringing grades K-5 to the mainstream of computer science education. In *Proceedings 15th Proceedings of the 46th ACM Technical Symposium on Computer Science Education,* p. 671-672.

Authors blinded for review. (2021).

Authors blinded for review. (under review).

Barr, V. & Stephenson, C. (March, 2011). Bring computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads, 2* (1), 48-54.

Bers, M. U. (2020). *Coding as a playground: Programming and computational thinking in the early childhood classroom*. Second Edition. Routledge.

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada* (Vol. 1, p. 25).

Capp, M. J. (2017). The effectiveness of universal design for learning: A meta-analysis of literature between 2013 and 2016. *International Journal of Inclusive Education*, *21*(8), 791–807. https://doi.org/10.1080/13603116.2017.1325074

International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA; 2011). Operational definition of computational thinking for K-12 education.

Israel, M., Jeong, G., Ray, M., & Lash, T. (2020). Teaching Elementary Computer Science through Universal Design for Learning. *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 1220–1226. https://doi.org/10.1145/3328778.3366823

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

ISTE (2018). Computational Thinking Competencies. Accessed on 7/14/21 from https://www.iste.org/standards/computational-thinking.

Kafai, Y. B., Proctor, C., & Lui, D. (2019). From theory bias to theory dialogue: Embracing cognitive, situated and critical framings of computational thinking for K-12 CS education. *ICER '19: Proceedings of the 2019 ACM Conference on International Computing Education Research*, 101-109. https://doi.org/10.1145/3291279.3339400

Knight, V. F., Wright, J., & DeFreese, A. (2019). Teaching Robotics Coding to a Student with ASD and Severe Problem Behavior. *Journal of Autism and Developmental Disorders*, *49*(6), 2632–2636. https://doi.org/10.1007/s10803-019-03888-3

Knight, V. F., Wright, J., Wilson, K., & Hooper, A. (2019). Teaching Digital, Block-Based Coding of Robots to High School Students with Autism Spectrum Disorder and Challenging Behavior. *Journal of Autism and Developmental Disorders*, *49*(8), 3113–3126. https://doi.org/10.1007/s10803-019-04033-w

Manches, A., & Plowman, L. (2017). Computing education in children's early years: A call for debate. *British Journal of Educational Technology, 48*(1), 191-201. https://doi.org/10.1111/bjet.12355

National Association for the Education of Young Children & National Council of Teachers of Mathematics. (2002). *Early childhood mathematics: Promoting good beginnings.* Retrieved from http://www.naeyc.org/positionstatements/mathematics.

National Center for Education Statistics. (2021). *Digest of Education Statistics: 2019*. U.S. Department of Education. https://nces.ed.gov/programs/digest/d19/

Rich, P. J., Jones, B., Belikov, O., Yoshikawa, E., & Perkins, M. (2017). Computing and engineering in elementary school: The effect of year-long training on elementary teacher self-efficacy and beliefs about teaching computing and engineering. *International Journal of Computer Science Education in Schools,* 1 (1), 1-20.

Taylor, M. S. (2018). Computer Programming With Pre-K Through First-Grade Students With Intellectual Disabilities. *The Journal of Special Education*, *52*(2), 78–88. https://doi.org/10.1177/0022466918761120

Tissenbaum, M., Weintrop, D., Holbert, N., & Clegg, T. (2021). The case for alternative endpoints in computing education. *British Journal of Educational Technology, 52*(3), 1164-1177. https://doi.org/10.1111/bjet.13072

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127-147.

Yadav, A., Stephenson, C., & Hong, H. (2017). Computational thinking for teacher education. *Communications of the ACM*, 60(4), 55-62.