



Computational Thinking in Early Childhood

Sunia Adavimath¹ & Dr. Anusha Soppimath²

¹Department of Computer Science,

Dr. D. Y. Patil Arts, Commerce Science College, Akurdi, Pune, India.

²Senior Resident, Department of Community Medicine

Sapthagiri Institute of Medical Sciences and Research Centre, Bangalore, Karnatak

Corresponding Author – Sunia Adavimath

DOI - 10.5281/zenodo.19331989

Abstract:

In the rapidly evolving digital age, traditional literacy skills alone are insufficient to prepare young children for the demands of the twenty-first century. This paper explores the importance of integrating computational thinking into early childhood learning, particularly within public library settings. Computational thinking—first introduced by Seymour Papert and later expanded by Jeannette Wing—is defined as a creative and systematic problem-solving process that includes skills such as logic, decomposition, pattern recognition, abstraction, evaluation, and algorithm design. Beyond skills, essential dispositions such as persistence, confidence, collaboration, and tolerance for ambiguity further strengthen this mindset.

The paper highlights the strong connections between computational thinking and key early childhood developmental domains, including approaches to learning, cognition, and language and literacy. It emphasizes that computational thinking can be introduced from infancy through developmentally appropriate, play-based experiences. Public libraries are identified as vital community institutions capable of promoting equitable access to these competencies through intentional programming, family engagement, and both low-tech and high-tech learning opportunities. Examples such as facilitated play sessions, storytime activities, LEGO clubs, and digital storytelling workshops demonstrate practical ways libraries can embed computational thinking into existing programs. By fostering computational thinking alongside creativity, critical thinking, collaboration, and communication, libraries play a crucial role in preparing young children and families to thrive in a media-rich, interconnected world.

Keywords: *Childhood, Education, Community etc.*

Introduction:

Libraries have traditionally supported early literacy development for young children and their families. However, in today's rapidly evolving digital world, basic literacy skills alone are not enough. Children and adults must now develop broader abilities that allow them to access, understand, evaluate, create, and share information across both digital and non-digital platforms.

These broader abilities are often described as twenty-first-century competencies, which include creativity, critical thinking, collaboration, and communication. Educational researcher Shuchi Grover argues that computational thinking should be recognized as an additional essential competency for children and youth.

Libraries are in a unique position to introduce computational thinking to children as young as birth through age nine, using developmentally appropriate and play-based

methods. By doing so, libraries help ensure equitable access to these essential skills for all families.

What is Computational Thinking:

The concept of computational thinking was first introduced by Seymour Papert in 1980 and later expanded by Jeannette Wing in 2008. Although definitions vary, computational thinking generally refers to a structured, creative approach to solving problems in ways that can be repeated or replicated.

It does not require computers. Instead, it involves:

- Breaking problems into manageable parts
- Recognizing patterns
- Removing unnecessary details
- Designing clear, step-by-step solutions

While it originated in computer science, computational thinking is now considered a broad problem-solving mindset applicable across disciplines.

Importantly, using digital tools does not automatically mean computational thinking is taking place. Coding can support it, but the mindset itself extends beyond technology.

Core Computational Thinking Skills:

Although no single universal list exists, six commonly recognized skills include:

- Logic – Predicting and analyzing outcomes
- Evaluation – Making informed judgments
- Decomposition – Breaking complex tasks into smaller parts
- Algorithm Design – Creating step-by-step instructions
- Pattern Recognition – Identifying similarities or repeated elements
- Abstraction – Removing unnecessary details to focus on essential elements

These skills are interconnected and can begin developing in early childhood through everyday experiences.

Computational Thinking Dispositions:

In addition to skills, certain attitudes or dispositions strengthen computational thinking. Organizations such as the International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) identify key dispositions, including:

- Confidence in handling complexity
- Persistence with challenging tasks
- Comfort with ambiguity
- Ability to solve open-ended problems
- Collaboration and communication skills

Researchers Karen Brennan and Mitchel Resnick add:

- Creative self-expression
- Questioning assumptions

These traits align closely with whole-child development principles and are essential for success in a connected world.

Why start in early childhood:

Jeannette Wing emphasized that computational thinking should begin in early childhood. Young children are naturally curious and actively construct knowledge about the world.

- Even infants demonstrate early forms of computational thinking when they:
- Recognize patterns in language
- Understand cause and effect
- Model social responses

Computational thinking connects closely with developmental domains identified by the National Education Goals Panel, including:

- Approaches to learning (curiosity, persistence, initiative)
- Cognition (math and scientific reasoning)
- Language and literacy

Because these domains overlap significantly, strengthening computational thinking also reinforces early literacy, math, and scientific reasoning skills.

The importance of equity:

Children from underserved communities may have limited access to digital tools, mentors, or enrichment opportunities. Libraries help close this gap by:

- Providing free programs
- Offering guided learning experiences
- Supplying access to digital devices and tools
- Supporting family engagement

By intentionally integrating computational thinking into programs, libraries promote educational equity.

The library's role:

While schools are beginning to integrate computational thinking, many early education settings do not yet fully support its development. Parents and caregivers may also lack understanding of how to nurture it.

- Libraries serve as community learning hubs and are well positioned to:
- Embed computational thinking into existing programs
- Support whole-family learning
- Model strategies for caregivers
- Provide access to both low-tech and high-tech resources
- Family learning experiences—where adults and children actively learn together—are especially powerful.

Practical examples in Library Programs:**1. Ages 0–3: Facilitated Play Programs:**

Play-based programs such as “Little Builders” or Early Literacy Play Dates naturally introduce computational thinking.

Activities may include:

- Sorting objects by color or size (pattern recognition)
- Building structures (decomposition and algorithm design)

- Responding to open-ended prompts like “How would you travel to the moon?” (creativity and problem-solving)
- Library staff model questioning strategies that encourage children to describe steps, sequences, and reasoning.

2. Ages 4–6: Storytime:

Story time offers rich opportunities to integrate computational thinking.

For example, during a celebration of *The Very Hungry Caterpillar* by Eric Carle:

- Visual schedules model sequencing (algorithm design)
- Letter games encourage decomposition
- Comparing two caterpillar stories builds pattern recognition
- Felt boards demonstrate ordered events
- Making edible caterpillars reinforces sequencing
- Children practice logic, collaboration, and creative thinking within familiar literacy activities.

3. Ages 7–9: LEGO Club:

LEGO programs support both literacy and computational thinking.

- In a town-replica building project, children:
- Identified essential buildings (abstraction)
- Broke the project into tasks (decomposition)
- Designed step-by-step plans (algorithm design)
- Collaborated and communicated in teams

The open-ended format allowed children to experiment, revise, and build confidence.

4. Multi-Age Family Program: Digital Storytelling:

Digital storytelling workshops allow families to co-create stories using tools like Scratch Jr. Families:

- Brainstorm and storyboard (decomposition)
- Sequence events (algorithm design)
- Simplify details (abstraction)
- Connect code blocks to direct characters
- Collaborate and communicate

These programs support creativity, persistence, and self-expression while strengthening family bonds. To ensure accessibility, libraries provide devices and guided support for families without home technology access.

Conclusion:

Computational thinking can be integrated into many existing library programs without replacing traditional early literacy efforts. Instead, it enhances them.

By intentionally incorporating computational thinking skills and dispositions, libraries:

- Strengthen creativity, critical thinking, collaboration, and communication
- Empower caregivers as learning partners
- Promote equitable access to digital literacy skills
- Prepare children to thrive in a media-rich, connected world

There is no single correct way to foster computational thinking. By introducing one skill at a time through meaningful, playful experiences, libraries can confidently expand their role as leaders in early childhood learning and family engagement

References:

1. Aman Yadav et al., "Computational Thinking as an Emerging Competence Domain;" in *Competence-based Vocational and Professional Education*, ed. Martin Mulder (Cham, Switzerland: Springer, 2017), 1,051-067.
2. Eric M. Meyers, "Guest Editorial;" *Information and Learning Sciences* 120, no. 5/6 (2019): 254-65.
3. Grover and Pea, "Computational Thinking: A Competency Whose Time Has Come."
4. Jeannette Wing, "Computational Thinking Benefits Society;" *Social Issues in Computing* (blog), January 10, 2014, <http://social-issues.cs.toronto.edu/2014/01/computational-thinking/>.
5. Paula Langsam, Amy Steinbauer, and Patricia Ballentine, "Build Digital Futures: Family Learning in Libraries" presentation, National Center for Families Learning Conference, Fort Lauderdale (FL), September 2018.
6. Seymour Papert, *Mindstorms: Children, Computers, and Powerful Ideas* (New York: Basic Books, 1980); Jeannette M. Wing, "Com-putational Thinking and Thinking About Computing," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 366, no. 1881 (2008): 3,717-725.
7. Shuchi Grover and Roy Pea, "Computational Thinking: A Com-petency Whose Time Has Come;" in *Computer Science Educa-tion: Perspectives on Teaching and Learning in School*, eds. Sue Sentance, Erik Barendsen, and Carsten Schulte (London: Bloomsbury Academic, 2018), 19-37.
8. Shuchi Grover, "The 5th 'C' of 21st century skills? Try computa-tional thinking (not coding)," *EdSurge*, February 25, 2018, www.edsurge.com/news/2018-02-25-the-5th-c-of-21st-century-skills-try-computational-thinking-not-coding.
9. Wing, "Computational Thinking and Thinking About Computing."
10. Yu-Hui Ching, Yu-Chang Hsu, and Sally Baldwin, "Developing Computational Thinking with Educational Technologies for Young Learners;" *TechTrends* 62 (2018): 563-73.