Chapter 3
Frameworks for Integration of Future-Oriented Computational Thinking in K-12 Schools

Scott Garrigan
Lehigh University, USA

ABSTRACT

Computational thinking (CT) K-12 curricula and professional development should prepare students for their future, but historically, such curricula have limited success. This chapter offers historical analogies and ways that CT curricula may have a stronger and more lasting impact. Two frameworks are central to the chapter's arguments. The first recalls Seymour Papert's original description of CT as a pedagogy with computing playing a formative role in young children's thinking; the computer was a tool to think with (1980, 1996). This "thinking development" framework emphasized child-centered, creative problem solving to foster deep engagement and understanding. Current CT seems to include creativity only tangentially. The second framework encompasses emergent machine learning and data concepts that will become pervasive. This chapter, more prescriptive than empirical, suggests ways that CT and requisite professional development could be more future-focused and more successful. It could be titled "Seymour Papert meets Machine Learning."

INTRODUCTION

This chapter begins with a background review to identify the salient ideas that advanced to form the core of today's CT framework but also to consider important ideas that receded to the periphery. CT's core is often described within a problem-solving framework; CT comprises the thinking skills needed to formulate and solve a problem in a way that technology systems can carry out the process. The second section discusses this problem-solving framework and its underlying assumptions and omissions. The largest section looks at CT through the subject-focused lenses of mathematics, science, computer science, and cognitive science. Each subject uses different approaches, different thinking skills, applied to core knowledge of the discipline. Some of these disciplines, particularly science and computer science,

DOI: 10.4018/978-1-7998-1479-5.ch003

continue to rapidly evolve since CT was originally described. Therefore, the section suggests that CT should emphasize emerging content and appropriate thinking skills to best prepare students for their future. The final section briefly asks what the role of the teacher should be in fostering CT. The classroom teacher, rather than the curriculum, seems to be the most salient school factor in student learning. For this reason, the depth of teacher understanding of CT, and how to develop that depth, should be major elements in CT integration in K-12 schools.

BACKGROUND

Computational Thinking ideas arose from founders of the internet, artificial intelligence, and educational technology. These CT ideas gestated half-a-century ago in 1968 discussions between MIT professor Seymour Papert and the Bolt, Baranek, and Newman inventors of the eastern half of what would become the internet. CT ideas were disseminated in Papert's 1970 paper, *Teaching Children Thinking: Artificial intelligence memo number 247,* in which he stated the ideas were "deeply influenced by AI pioneer Marvin Minsky …" Papert christened (first named) "Computational Thinking" in a 1996 paper, and he researched and elaborated CT ideas through MIT's Artificial Intelligence lab (which he co-founded with Minsky). He created the LOGO computer language to develop children's mathematical thinking, and LOGO was an early educational technology used in schools for decades. LOGO was developed from LISP, the artificial intelligence language of the day, and modern children's languages like Scratch are LOGO's direct descendants. CT had an honorable beginning.

Three of Papert's central CT ideas form much of the framework discussed below: 1) CT is a way of thinking that needs to begin in elementary school, 2) essential core elements of CT are curiosity and creativity, and 3) CT helps us understand human thinking. These ideas evolved from Papert's 1960's work investigating how young children learn mathematics in collaboration with his mentor, child psychologist Jean Piaget. Through Papert's lens, CT in education could well be called Computational *Learning*. But for reasons to be discussed in the closing, Papert's work had minimal impact on mainstream classroom teaching and learning. It was a decade after Papert coined the term *Computational Thinking* that Jeanette Wing, head of Carnegie Mellon's computer science department, brought CT into the education mainstream.

Wing's 2006 ACM Viewpoint, *Computational Thinking,* explained six elements of CT she characterized as necessary for everyone to understand today's technical world. While Wing wrote for a university community, her meme resonated strongly in the K-12 community where, by then, computers had become common in classrooms. Prior to the general use of computers in schools, teachers and administrators were not prepared to understand the context and importance of CT. As computer science professor, later VP of Microsoft Research, and now Director of Columbia Data Sciences Institute, Wing developed a deep understanding of the network of connections of computer science concepts to a broad swath of university study and adult life. In CT she embraced comparisons of human and machine learning, randomness, heuristics, recursion, and, of course, abstraction. She connected CT to fields from biology and chemistry to physics and economics. But her only references to children learning CT outside of college were, "To reading, writing, and arithmetic, we should add computational thinking to every child's analytical ability" and "We should expose pre-college students to computational methods and models."