

Chapter 15

Pre-Service Teacher Preparation to Integrate Computational Thinking

Lesley Farmer

California State University, Long Beach, USA

Shuhua An

California State University, Long Beach, USA

ABSTRACT

United States education has experienced a big push for students to learn coding as part of computer science and more explicitly address computational thinking (CT). However, CT remains a challenging subject for many students, including pre-service teachers. CT, which overlaps mathematics and computer science, tends to be offered as an elective course, at best, in P-16 education. Pre-service teaching profession students usually do not have foundational knowledge to guide them in integrating computational thinking into the curriculum that they will eventually teach as instructors themselves. This chapter explains computational thinking in light of K-8 education, discusses issues and needs in integrating CT into K-8 curriculum, identifies relevant theories and models for teaching CT, describes current practice for integrating computational thinking into K-8 curriculum, and discusses pre-service teachers' preparation that can lead to their successful incorporation of CT into the curriculum.

INTRODUCTION

United States education has experienced a big push for students to learn coding as part of computer science and more explicitly address computational thinking. Some of this focus on STEM is motivated by the 2013 Next Generation Science Standards (NGSS), which asserts “In both science and engineering, mathematics and computation are fundamental tools for representing physical variables and their relationships. They are used for a range of tasks such as constructing simulations; statistically analyzing data; and recognizing, expressing, and applying quantitative relationships.” NGSS (2013) called for “us-

DOI: 10.4018/978-1-7998-1479-5.ch015

ing mathematics and computational thinking” as one of its eight practices of science and engineering, identifying that practice as essential for all students to learn. Additionally, the 2017 International Society for Technology in Education standards for students and for educators directly addressed computational thinking and its benefits.

Computational thinking (CT) is often associated with mathematics, which remains a challenging subject for many students, including pre-service teachers. CT also overlaps computer science, which tends to be offered as an elective course in P-16 education. Furthermore, even fewer pre-service teachers were taught STEM problem-solving such that their solutions were derived in ways that a computer could execute them. In short, pre-service teaching profession students usually do not have foundational knowledge to guide them in integrating computational thinking into the curriculum that they will eventually teach as instructors themselves.

This chapter explains computational thinking in light of K-8 education, discusses issues and needs in integrating CT into K-8 curriculum, identifies relevant theories and models for teaching CT, describes current practice for integrating computational thinking into K-8 curriculum, and discusses pre-service teachers’ preparation that can lead to their successful incorporation of CT into the curriculum.

COMPUTATIONAL THINKING

History

The underlying idea of computational thinking is centuries old. The term “computational” appeared as early as the 14th century, based on the ancient Latin word “computation,” meaning to calculate or reckon. Similarly, the first computers were humans: individuals who computed. As such, computational thinking was traditionally associated with mathematical reasoning. Along the way, mathematical reasoning was sometimes relegated to solving textbook math problems with little reference to the humanities.

With the advent of computing machines, computational thinking shifted to computer science, particularly in terms of computer programming. Computers manipulate data according to directions or algorithms that can be applied to different inputs and different kinds of problems. The advantage of such machines is their processing speed of such data. The Turing machine in World War II is an early example of machine computing power that could decipher a sophisticated coding algorithm more quickly than a room of humans. In the 1940s, John von Neumann posited that computational thinking was a scientific method, and computational thinking was labelled algorithmic thinking in the 1950s (Demir et al., 2018).

Seymour Papert was one of the early educators who saw the possibility of using a simple programming language to help children learn how to think and solve problems in a constructionist manner. Papert used the term “computational thinking” in his 1980 book *Mindstorms* as he explained how children, in particular, could represent problems mathematically and then explore those problems dynamically in computer-based microworlds. The concept of algorithmic thinking was central to this process.

In the 1980s, many schools offered elective courses in computer programming, such as Basic or Pascal. However, these courses were usually disconnected from other curricula and did not seem relevant to most students. 1990s programming courses increasingly came pre-packaged so students did less debugging and original designing. In the last ten years, computer programming education, often renamed as “coding,” has made a comeback, to some degree out of industry need for technically skilled employees (Kafai, 2016).

17 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/pre-service-teacher-preparation-to-integrate-computational-thinking/246601

Related Content

Taboos and Storytelling for Teaching and Learning in Zimbabwe: Curriculum Decolonization in Africa

Pindai Mangwanindichero Sithole and Beatrice Maupa Bondai (2020). *International Journal of Curriculum Development and Learning Measurement* (pp. 53-65).

www.irma-international.org/article/taboo-and-storytelling-for-teaching-and-learning-in-zimbabwe/260747

Nonverbal Learning Disorder: Past, Present, and Future

Joseph E. Casey (2016). *Special and Gifted Education: Concepts, Methodologies, Tools, and Applications* (pp. 130-168).

www.irma-international.org/chapter/nonverbal-learning-disorder/151203

Implementing iPad and Mobile Technologies for Students with Intellectual Disabilities

Cathi Draper Rodríguez, Iva Strnadová and Therese M. Cumming (2016). *Special and Gifted Education: Concepts, Methodologies, Tools, and Applications* (pp. 1408-1425).

www.irma-international.org/chapter/implementing-ipad-and-mobile-technologies-for-students-with-intellectual-disabilities/151261

Dance and Movement as Therapy for Children with Autism Spectrum Disorders (ASD): A Case for Kuching, Sarawak

Jane Teo and Ong Puay Hoon (2016). *Special and Gifted Education: Concepts, Methodologies, Tools, and Applications* (pp. 2294-2305).

www.irma-international.org/chapter/dance-and-movement-as-therapy-for-children-with-autism-spectrum-disorders-asd/151304

The Differences of Perceived Efficacy Between Pupils and Experts in Fostering 21st-Century Skills

Chi-Syan Lin and Cheng-Ying Lin (2022). *International Journal of Curriculum Development and Learning Measurement* (pp. 1-13).

www.irma-international.org/article/differences-perceived-efficacy-between-pupils/290386