Teaching Computational Thinking

An Integrative Approach for Middle and High School Learning

By: Maureen D. Neumann, Lisa Dion

Citation:

Teaching Computational Thinking: An Integrative Approach for Middle and High School Learning
By: Maureen D. Neumann, Lisa Dion
DOI: 10.7551/mitpress/11209.001.0001
ISBN (electronic): 9780262366144
Publisher: The MIT Press
Published: 2021

OA Funding Provided By:

The open access edition of this book was made possible by generous funding from Arcadia—a charitable fund of Lisbet Rausing and Peter Baldwin.

Preface

> This book provides teachers with a dynamic learning experience using computational thinking, computer programming, and unplugged activities that are grounded in different learning theories and pedagogical strategies to help students gain twenty-first-century skills.

It is generally accepted that there is an urgent need to introduce computer science concepts and skills into the K–12 curriculum of the United States (Barr and Stephenson 2011; Cuny 2011; Grover and Pea 2013; Kafai and Burke 2014; Nager and Atkinson 2016; Walser 2008; Wilson et al. 2010; Wing 2006, 2008, 2016). As digital technology becomes more ubiquitous around the world, the need for professionals with computing skills is expected to increase. Even those employed in other fields will need to learn about the capabilities and limitations of computers, data networks, and mobile devices. One might also argue that all citizens need to be able to respond to the political and ethical issues that advances in digital technology bring to the forefront (e.g., trade-offs between personal privacy and convenient data access, copyright protection vs. fair use, and the security of the nation's data infrastructure, including the integrity of electronic voting). We believe that the solutions to these problems encompass policy, cultural, and educational change. Our book focuses on the latter. As more schools add computer science to their curricula, more teachers will need to learn about computational thinking to provide a solid foundation for incorporating computer science principles and applying programming skills. Our book offers a way to present these concepts to all middle and high school teachers and students.

In brief, we integrate a variety of pedagogical practices, across content and curricula, to demonstrate how computational thinking is a fundamental skill that we all (not just computer scientists) use. We believe that computational thinking enhances every student's learning of the core content subjects as well as every child's repertoire of reading, writing, and arithmetic. We concur with Jeanette Wing's (2006) views that computational thinking teaches students critical thinking skills: how to think algorithmically (i.e., how to solve problems using a progression of logical steps) and how to create useful and artistic artifacts with digital technology. Just as the printing press facilitated the spread of the "three Rs" (reading, writing, and arithmetic) during the Renaissance, computing and computers facilitate the spread of computational thinking and complex problem-solving abilities.

Since computational thinking is derived from human thought processes, it is vital to demonstrate its relevance to every student and every teacher. Children naturally learn algorithms (e.g., the sequence of steps required to tie their shoes) as well as logical systems (e.g., the rules of a game). Likewise, computational thinking can be integrated into many different contexts. Our book demonstrates ways for middle and high school teachers to incorporate computational thinking through the incorporation of computer science principles and programming skills into a broad range of subjects. For example, we show how art teachers can develop lessons on algorithmic art (also called generative art). By writing programs in Scratch or Codesters, students learn how to apply algorithms and geometric properties to create interesting and artistic computer images. Likewise, our book shows how English language arts students can use topological graphs to analyze the social networks that explicitly connect fictional characters in a given literary work. Social studies teachers see how to include algorithms in their lessons to help students generate Cretan, Roman, and medieval labyrinths using a simple algorithm.

With our combined 60 years of teaching mathematics, coding, and educational topics across the K–20 spectrum, we have found that a variety of open-ended problem-solving tasks, project-based inquiries, and paper-and-pencil activities enable students to develop a concrete understanding of the abstract principles needed for creating code and other digital artifacts. We employ a low-threshold, high-ceiling pedagogical practice for the activities and rich tasks we include for computational thinking. The activities and rich tasks we present in this book grew out of our experience teaching students in grades 5–12 during after-school programs (e.g., Girls Who Code) and summer camps (e.g., Governor's Institutes and Code Camp). We present our activities and rich tasks in broad strokes because it is our intent to give teachers flexibility and choice in how to present the material to their students rather than being too prescriptive with timing, organization, and procedural steps. We believe all teachers are smart, industrious, and continuous learners, but some may not currently have an understanding of computational thinking and are seeking ways to transform their teaching practice. We understand that not everyone will be able to use these activities as they are provided, but we are confident in teachers' professional ability to choose and adapt activities that work for them in their context.

In creating this book, we are motivated by our belief that computational thinking, like written and oral communication and basic numeracy, is too important to relegate to a single subject. By inserting computational thinking into many different contexts, students will learn well before high school that computer science and programming has broad relevance. Distributing these "seeds" early and widely should generate greater interest in the need for studying computer science in high school and college. Broad coverage of computer science across K–12 education enables students from underrepresented groups to access information and experiences they may not have had at home or after school. Moreover, by recruiting teachers to add computational thinking strategies (and possibly coding) to their curricula, we may be able to gradually increase the number of students interested in computer science as a career or their ability to use these strategies and skills in other careers of interest.

We also wish to emphasize that this book is designed to help all teachers see how to include computational thinking and computer programming in a wide range of subjects, including literature, history, physical sciences, social sciences, and the visual arts. Though our primary focus is on middle and high school grades, some ideas can be used in the elementary grades. As awareness of the importance of computer science in education increases, we hope that this curriculum-integration model will be replicated across a variety of learning spaces.

Maureen D. Neumann
Lisa Dion
Robert Snapp

The MIT Press would like to thank the anonymous peer reviewers who provided comments on drafts of this book. The generous work of academic experts is essential for establishing the authority and quality of our publications. We acknowledge with gratitude the contributions of these otherwise uncredited readers.

This book was set in Stone Serif and Stone Sans by Westchester Publishing Services.

Library of Congress Cataloging-in-Publication Data

Names: Neumann, Maureen D., author. | Dion, Lisa (Computer scientist), author.
Title: Teaching computational thinking : an integrative approach for middle and high school learning / Maureen D. Neumann and Lisa Dion with Robert Snapp.
Description: Cambridge, Massachusetts : The MIT Press, 2021. | Includes bibliographical references.
Identifiers: LCCN 2021000766 | ISBN 9780262045056 (paperback)
Subjects: LCSH: Computer science—Study and teaching (Secondary) | Critical thinking—Study and teaching (Secondary)
Classification: LCC QA76.27 .N474 2021 | DDC 004.071—dc23
LC record available at https://lccn.loc.gov/2021000766