

# Detailed Table of Contents

Foreword .....	xviii
Preface .....	xxi
Acknowledgment .....	xxx

## Section 1

### Pedagogical Foundations and Instructional Design in Computer Science

#### Chapter 1

Pedagogical Innovations for the Modern Computer Science Classroom: 50	
Teaching Tips and Practical Strategies for Computing Education.....	1
<i>Manuel B. Garcia, FEU Institute of Technology, Philippines</i>	
<i>Ari Happonen, LUT University, Finland &amp; Taylor's University, Malaysia</i>	
<i>Friday Joseph Agbo, Willamette University, USA</i>	
<i>Michael Agyemang Adarkwah, Friedrich Schiller University, Germany</i>	
<i>Ioannis Kazanidis, Democritus University of Thrace, Greece</i>	
<i>Thomas K. F. Chiu, The Chinese University of Hong Kong, Hong Kong</i>	
<i>Robertas Damaševičius, Vytautas Magnus University, Lithuania</i>	
<i>Serhiy O. Semerikov, Kryvyi Rih State Pedagogical University, Ukraine</i>	

Computer science education is undergoing a global transformation as digital technologies continue to redefine how knowledge is created, shared, and applied. Despite this evolution, many teaching practices in computing remain rooted in instructor-centered models that prioritize syntax and correctness over creativity, collaboration, and reflection. This chapter addresses this gap by presenting pedagogical innovations for the modern computer science classroom. Drawing on contemporary research in computing education and learning theories, it identifies ten interrelated themes and fifty practical strategies for effective instruction. Each theme is grounded in theory, contextualized through classroom examples, and supported by current literature. Collectively, these strategies advocate a shift from transmission-based teaching toward inquiry-driven learning that cultivates computational thinkers, creative problem solvers, and socially responsible innovators. The chapter concludes by discussing the implications of these pedagogical approaches for the future of computer science education.

## **Chapter 2**

Interdisciplinary Approaches in Computer Science Teaching: Foundations, Integration, and Transformation of Education ..... 43

*Qurrotul Aini, UIN Syarif Hidayatullah Jakarta, Indonesia*

*Fitroh Fitroh, UIN Syarif Hidayatullah Jakarta, Indonesia*

This section highlights the importance of an interdisciplinary approach in teaching Computer Science (CS). This approach integrates various disciplines such as social sciences, mathematics, and technology to create holistic learning that is relevant to complex real-world challenges. An integrated curriculum model guides students in understanding the relationships between concepts from various fields, thereby increasing their engagement and deepening their understanding. This approach also encourages character and competency development, as well as supporting innovation in solving social and environmental problems. Although it offers many benefits, interdisciplinary implementation requires careful planning and sufficient resources, as well as addressing concerns related to technical depth. Overall, this approach is key to preparing graduates who are able to face global dynamics and advance the field of computer science in a sustainable manner.

## **Chapter 3**

Recentring Computer Science Education on Problem-Solving: Pedagogical Foundations and Curriculum Design ..... 73

*Mustafa Kayyali, Maaref University of Applied Sciences, Syria*

This chapter explores the centrality of problem-solving in computer science education through the dual lenses of project-based learning (PjBL) and problem-based learning (PBL). While both approaches emphasize inquiry, collaboration, and real-world engagement, they differ in scope, structure, and outcomes. PjBL often immerses learners in extended projects that mirror professional practice, whereas PBL foregrounds open-ended problems that cultivate critical reasoning and adaptability. Anchored in the belief that computer science is not merely about mastering syntax or tools but about nurturing the ability to think computationally and creatively, the chapter argues for a “problem-solving first” philosophy in curriculum design. By examining practical applications, challenges, and emerging trends, it highlights how integrating PjBL and PBL can empower students to approach complexity with confidence, cultivate resilience in the face of ambiguity, and prepare for a rapidly evolving technological landscape.

## **Chapter 4**

Formative Epistemics and Inclusive Design Approach Using Constructive Alignment and Metacognition: Sociotechnical Pedagogy for Computer Education .....	107
<i>Ashraf Alam, Alliance University, Bengaluru, India</i>	

This chapter proposes a pedagogy first architecture for computing assessment that couples formative routines with minimalist analytics and equity by design. It argues for alignment to disciplinary big ideas, visible success criteria, short revision cycles, and multimodal evidence including code histories, test rationales, diagrams, and oral defenses. Peer and self-judgment are cultivated as disciplined practices, supported by universal design and culturally sustaining tasks. Analytics remain small, interpretable, and privacy preserving, answering concrete instructional questions within hours. Capacity is built through coaching, lesson study, micro credentials, and calibrated observation, with change managed through protected schedules and toolkits. Comparative cases from K to 8, secondary, undergraduate, and reskilling contexts show feasibility and impact. An improvement science agenda links logic models to measurable gains in competence, fairness, and credible external signaling.

### **Section 2**

#### **Inclusivity, Ethics, and Social Responsibility in CS Education**

## **Chapter 5**

Transforming Computer Science Education: Inclusivity, Ethics, and Emerging Technologies.....	143
<i>Hanh Le, University of the Cumberlands, USA</i>	

Computer science education is undergoing a profound transformation at the intersection of inclusivity, ethics, and rapidly evolving technologies. This chapter explores how innovative pedagogical approaches can effectively address the diverse needs of contemporary learners while simultaneously addressing the ethical, social, and societal challenges posed by technological advancements. Interdisciplinary models such as STEAM (Science, Technology, Engineering, Arts, and Mathematics), along with open-source tools and collaborative platforms, enhance experiential learning, creativity, and critical thinking skills. Central to this vision is the integration of ethical reasoning and responsible computing practices, equipping students to make principled, informed decisions in an increasingly automated and AI-driven world. The chapter concludes with actionable recommendations for educators, administrators, and policymakers, emphasizing the importance of professional development for computer science education.

## Chapter 6

Gender-Inclusive Pedagogies in Computer Science Education: Insights, Implications, and Future Directions..... 175

*Narcisan S. Galamiton, Cebu Technological University, Philippines*

*Mary Jane C. Sim, Cebu Technological University, Philippines*

*Dharel P. Acut, Cebu Technological University, Philippines*

*Damola Olugbade, First Technical University, Nigeria*

*Asegul Hulus, University of Nicosia, Cyprus*

*Ramesh Chander Sharma, Dr. B.R. Ambedkar University, Delhi, India*

*Ari Happonen, LUT University, Finland*

*Manuel B. Garcia, FEU Institute of Technology, Philippines*

Despite growing demand for computing professionals, women remain significantly underrepresented in computer science education and technology careers. This chapter presents a narrative review of literature on gender-inclusive pedagogies, examining classroom practices, curriculum design, assessment strategies, mentorship, and institutional initiatives that support female learners' engagement, persistence, and sense of belonging. Drawing on international and regional studies, the review identifies effective approaches such as collaborative learning, identity-affirming and real-world curricula, formative assessment, and structured support systems, while highlighting persistent gaps including Western-centric perspectives, limited intersectional analyses, and misalignment between policy rhetoric and classroom practice. Grounded in technofeminist theory, the chapter discusses implications for educators, institutions, policymakers, and future research aimed at advancing equity in computing education.

## Chapter 7

Inclusive Pedagogy in Computer Science Education: A Critical Pedagogical Framework for Computational Access..... 207

*Akshara Singh, Christ University, India*

*Lawrence Kujur, Christ University, India*

Computer science (CS) inclusive teaching provides an opportunity for everyone to learn in a way that is fair and accessible. Through the traditional approaches, neurodiverse and disadvantaged learners are often cut off due to the strictness of the instruction and the surrounding limitations on support. UDL (Universal Design for Learning) and CRP (Culturally Responsive Pedagogy) are the two major strategies for inclusive teaching that facilitate flexible, engaging, and relevant learning. UDL allows you to choose different ways to learn and to express your understanding, and CRP connects the content with the students' cultural backgrounds. Gallaudet University and AccessCSforAll are some of the initiatives that have been proving that learners are more engaged when using assistive tools and through the application of adaptive methods. Supported by educating CS for all, U.N. SDG 4, inclusivity

in computer sciences education not only narrows the digital gap but also opens up the door for every student to thrive in a diverse and tech-savvy world.

## **Chapter 8**

Ethics and Responsible Computing in the Classroom: Embedding Ethical Reasoning in Computer Science Education ..... 237

*Bright Keswani, Poornima University, Jaipur, India*

Rapid growth in computing technologies has been witnessed in recent decades. Along with progress, serious ethical concerns have been raised. Issues of privacy, security, algorithmic bias, digital well-being, and the responsible use of artificial intelligence have become common. Students of computer science are expected to become future designers and innovators. They must be trained not only in technical knowledge but also in ethical responsibility. However, traditional curricula have placed more focus on programming, systems, and theory. Ethical aspects have often been left out or given little attention. To build responsible digital citizens, ethics must be included in classroom learning. This chapter focuses on the need to integrate ethics into computing education, and pedagogical practices, challenges, and strategies for responsible teaching have been discussed.

### **Section 3**

## **Teaching, Learning, and Assessment in the Age of Artificial Intelligence**

## **Chapter 9**

Assessment of Coding Skills and Programming Knowledge in the Age of Generative AI: Best Practices and Effective Strategies for Computer Science Education ..... 275

*Manuel B. Garcia, FEU Institute of Technology, Philippines*

*Ahmed Mohamed Fahmy Yousef, Fayoum University, Egypt*

*Ramazan Yilmaz, Bartın University, Turkey*

*Ramesh Chander Sharma, Dr. B.R. Ambedkar University, Delhi, India*

*Thomas K. F. Chiu, Chinese University of Hong Kong, Hong Kong*

*Robertas Damaševičius, Vytautas Magnus University, Lithuania*

The rapid emergence of artificial intelligence (AI) and generative AI tools poses a significant threat to the validity of traditional programming assessments. As the boundary between authored and AI-generated code becomes increasingly indiscernible, long-standing assessment models centered on output correctness and code submission are at risk of obsolescence. Despite its urgency, prior work has largely concentrated on detection, with limited emphasis on reimagining assessment design. This chapter addresses that gap by proposing strategies for assessing programming proficiencies in an AI-mediated context. Its objective is to help educators move

beyond surveillance-based models and adopt approaches that emphasize uniquely human cognitive capacities. These pedagogical strategies advance the field by shifting the discourse from reactive prevention to proactive, pedagogically aligned assessment design. In doing so, the chapter affirms that the future of programming assessment is not about resisting AI but about designing systems that assess human thinking over code output.

## **Chapter 10**

Pedagogical Promise and Peril of AI: A Text Mining Analysis of ChatGPT  
Research Discussions in Programming Education..... 305

*Juvy C. Grume, Pampanga State University, Philippines*

*John Paul P. Miranda, Pampanga State University, Philippines*

*Aileen P. De Leon, Pampanga State University, Philippines*

*Jordan L. Salenga, Pampanga State University, Philippines*

*Hilene E. Hernandez, Pampanga State University, Philippines*

*Mark Anthony A. Castro, Pampanga State University, Philippines*

*Vernon Grace M. Maniago, Pampanga State University, Philippines*

*Joel D. Canlas, Pampanga State University, Philippines*

*Joel B. Quiambao, Pampanga State University, Philippines*

GenAI systems such as ChatGPT are increasingly discussed in programming education, but the ways in which the research literature conceptualizes and frames their role remain unclear. This chapter applies text mining to publications indexed in a leading academic database to map scholarly discourse on ChatGPT in programming education. Term frequency analysis, phrase pattern extraction, and topic modeling reveal four dominant themes: pedagogical implementation, student-centered learning and engagement, AI infrastructure and human–AI collaboration, and assessment, prompting, and model evaluation. The literature prioritizes classroom practice and learner interaction, with comparatively limited attention to assessment design and institutional governance. Across studies, ChatGPT is positioned both as a learning aid that supports explanation, feedback, and efficiency and as a pedagogical risk linked to overreliance, unreliable outputs, and academic integrity concerns. These findings support responsible integration and highlight the need for stronger assessment and governance mechanisms.

## **Chapter 11**

Pedagogical Framework for Developing Students' Python Programming Competencies Using Intelligent Systems ..... 331

*Kasim Khusanov, Turin Polytechnic University, Uzbekistan*

*Ravshanjon Kakharov, Namangan State University, Uzbekistan*

*Xoshimjon Erkinjonovich Xolmirzayev, Namangan State University, Uzbekistan*

*Akhrorbek Abdullayev, Namangan State University, Uzbekistan*

This chapter explores the use of intelligent learning systems to develop students' Python programming competencies in higher education. The research highlights how these systems provide automated assessment, personalized feedback, and gamified learning experiences that foster programming literacy, learner independence, and motivation. By demonstrating the effectiveness of intelligent systems in enhancing Python programming competencies, this chapter underscores the transformative role of artificial intelligence and machine learning in education. The results of this study clearly demonstrate that integrating intelligent learning systems into Python programming education significantly improves both the quality and efficiency of the learning process. The use of intelligent systems transforms traditional instructional models into adaptive, interactive, and learner-centered environments that align with the pedagogical demands of the digital age.

### **Section 4**

#### **Emerging Learning Environments and Educational Media**

## **Chapter 12**

Game Development as a Pedagogical Innovation for Project-Based Learning in Computer Graphics and Gaming ..... 367

*Rachana Yogesh Patil, Pimpri Chinchwad College of Engineering, India*

*Yogesh Patil, D.Y. Patil College of Engineering, India*

The advent of project-based learning (PBL) as a pedagogical method has made the Computer Graphics and Gaming course a practical, real-life learning experience to undergraduate computer engineers. Creating and producing their own video games, students combined the knowledge of graphics programming, storytelling, user interface design, animation, and game mechanics. It was a step-by-step process, starting with basic ideas and going into practice, and ending with demonstrations judged on technical implementation, creativity, interactivity and teamwork. The approach incorporated use of iterative design, peer feedback, and reflective practices and was inspired by the experiential learning cycle and multiple intelligences of Kolb and Gardner respectively. The chapter provides planning, implementation and evaluation approaches, discusses possible obstacles like resource shortage and

diversity of learners, and provides flexible information on transferring this PBL model to other STEM fields involving creativity, technical skill and teamwork.

### **Chapter 13**

Reimagining Computer Science Education in Remote, Hybrid, and Flipped Classrooms for Future-Ready Learners ..... 399

*Kamlesh Kumar Gautam, SDGI Global University, India*

*Tarun Kumar Vashishth, Department of Computer Applications, Vidya University, Meerut, India*

*Shashiraj Teotia, Swami Vivekanand Subharti University, India*

*Vikas Sharma, Department of Computer Applications, SRM Institute of Science and Technology, Ghaziabad, India*

*Jyotsana Khera, IIMT University, India*

*Anand Kumar, SRM Institute of Science and Technology, India*

*Shahanawaj Ahamad, University of Hail, Saudi Arabia*

Advances in Learning Technology are promoting education in Computer Science from models of education utilizing predominantly lectures as forms of education, to innovations in pedagogy and into emerging models of student centered pedagogies. Distance and hybrid education in knowledge delivery and the concepts of the flipped learning environment are changing the learning output in Computer Science, providing more flexible, accessible, and participatory models of education which are synchronous with the evolving demands of the society for skills formation. Challenges to the pedagogical basis, strengths, and weaknesses of these forms of education and how they cope with problem solving and interdisciplinary learning are discussed in the light of the canons of equitable access to education. It examines how distance delivery is increasing access to greater diversity of learner, how hybrid models can help provide a degree of collaborative learning while also affording flexibility, and how flipped learning can enhance levels of active learning and levels of cognitive comprehension.

### **Chapter 14**

A Bibliometric Review of Pedagogical Innovations and Future Directions for Generative Artificial Intelligence in Computer Science Education ..... 429

*Damola Olugbade, First Technical University, Nigeria*

*Samson O. Ajao, Bowen University, Nigeria*

*Hassan Abuhassna, Sunway University, Malaysia*

*Ifedayo Emmanuel Adu, Bamidele Olumilua University of Education, Science, and Technology, Nigeria*

*Owolabi Paul Adelana, The Open University, Milton Keynes, UK*

This chapter provides a bibliometric review of research on Generative Artificial Intelligence (GAI) in Computer Science (CS) education over the past decade. A

total of 582 documents been catalogued in Scopus for the period of 2014-2024 was used. The publications were analyzed using Biblioshiny for Bibliometrix and the results were used to identify gaps in the literature and guide future research directions on GAI's impact in education. The findings revealed the surge in publications and citations post 2020 can be attributed to large language models and AI coding assistants. Thematic and keyword analyses show that GAI has been applied primarily in programming education, intelligent tutoring, personalized learning, and assessment. These applications yield outcomes such as enhanced efficiency, student engagement, and scalability, yet also raise concerns about academic integrity, over-reliance, and algorithmic bias. GAI research is spearheaded by developed nations which poses fairness concerns, and simultaneously fosters advancement and moral quandaries in computer science pedagogy.

<b>Afterword</b> .....	457
<b>Compilation of References</b> .....	459
<b>About the Contributors</b> .....	519
<b>Index</b> .....	533