Chapter 3
Recentering Computer Science Education on Problem-Solving:
Pedagogical Foundations and Curriculum Design

Mustafa Kayyali
https://orcid.org/0000-0003-3300-262X
Maaref University of Applied Sciences, Syria

ABSTRACT

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.

DOI: 10.4018/979-8-3373-6546-6.ch003

INTRODUCTION

Computer science (CS) has always been more than the mastery of algorithms, programming languages, or hardware configurations. At its very core, it is the art and discipline of problem-solving: the persistent search for solutions in a world increasingly defined by complexity, uncertainty, and constant change. When we teach CS merely as the acquisition of technical tools, we risk reducing it to a mechanical exercise of syntax and commands, stripped of the deeper habits of mind that enable genuine creativity and resilience. Yet when we place problem-solving at the heart of the educational process, we begin to cultivate learners who can not only survive in a dynamic digital environment but also shape it with ingenuity and purpose. It is here that pedagogical approaches such as project-based learning (PjBL) and problem-based learning (PBL) come into sharp focus. Both methods reject the notion of education as passive reception of information, insisting instead that students must actively engage, inquire, and wrestle with authentic challenges. They transform the classroom into a laboratory of thinking, where knowledge is not handed down in neat packages but is discovered, contested, and applied in messy, real-world contexts. For CS, a field where the distance between theory and practice is often razor-thin, such approaches are not merely desirable but essential.

The turn toward problem-solving as the organizing principle of curriculum design reflects broader shifts in educational philosophy. Across disciplines, there is a growing recognition that the traditional lecture-dominated model, while efficient for transmitting large volumes of information, is ill-suited to fostering the critical thinking and adaptability demanded by contemporary societies (Gopal, 2025). Employers, too, increasingly seek graduates who can navigate ambiguity, collaborate across disciplines, and devise creative solutions rather than simply replicate existing methods. In CS, this demand is particularly acute. Software development, data science, cybersecurity, and emerging fields like artificial intelligence require not only technical competence but also the capacity to ask the right questions, frame problems thoughtfully, and imagine multiple pathways toward resolution (Kolosnjaji et al, 2024). PjBL and PBL are often spoken of in the same breath, and with good reason (Moore, 2025). Both share a commitment to student-centered pedagogy, collaborative inquiry, and the integration of knowledge across domains. Yet their nuances matter. PjBL tends to emphasize extended projects that culminate in tangible outputs that mirror professional practice. PBL, on the other hand, begins with a deliberately ill-structured problem, encouraging learners to analyze, hypothesize, and explore possible solutions before formal instruction provides additional scaffolding. One approach is expansive, often spanning weeks or months; the other is iterative, focused on cultivating the habits of inquiry and critical reasoning that underpin effective problem-solving.