Educator Insights: A Journal of Teaching Theory and Practice
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6964

THE ROLE AND IMPORTANCE OF ARTIFICIAL INTELLIGENCE IN TEACHING PROGRAMMING FUNDAMENTALS

Sh. Abdullayev
Associate Professor, Department of Information Technology Fergana State University
shaxbozfardu@2023@gmail.com

Abstract
The rapid evolution of Artificial Intelligence (AI) has significantly impacted various sectors, with education being one of the most transformed. In the realm of Computer Science, specifically the teaching of programming fundamentals, AI tools are shifting the pedagogical landscape from traditional lecture-based methods to personalized, adaptive learning experiences. This article explores the multifaceted role of AI in assisting both educators and students, focusing on its ability to provide real-time feedback, automate grading, and offer customized learning paths. While highlighting the benefits, the paper also addresses the ethical implications and the necessity of maintaining a balance between AI assistance and the development of core problem-solving skills.

Keywords: Artificial Intelligence, Programming Education, Intelligent Tutoring Systems, Adaptive Learning, Computer Science Pedagogy, Automated Assessment.

Introduction

The landscape of Computer Science Education (CSE) is undergoing a paradigm shift, driven by the rapid integration of Artificial Intelligence (AI) into pedagogical frameworks. Traditionally, learning the fundamentals of programming—encompassing syntax mastery, algorithmic logic, and computational thinking—has been characterized by a steep learning curve. For novice learners, the transition from natural language to the rigid, formal structures of programming languages like C++, Python, or Java often presents significant cognitive hurdles. Historically, these challenges have been addressed through

Educator Insights: A Journal of Teaching Theory and Practice
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6964

human-led instruction, yet the increasing global demand for software literacy has created a scalability gap that traditional classrooms struggle to fill.

The emergence of Generative AI and Large Language Models (LLMs) has introduced a transformative dimension to how programming is taught and learned. AI is no longer merely a theoretical concept but a functional partner in the educational process. In the context of teaching programming fundamentals, the role of AI extends beyond simple automation; it serves as a personalized cognitive scaffold that supports students during the critical early stages of their development. The importance of this technology lies in its ability to provide instantaneous, 24/7 feedback—a luxury seldom available in traditional academic settings where the student-to-teacher ratio is high.

However, the integration of AI into the curriculum is not without its complexities. While it offers unparalleled support in debugging, code generation, and personalized learning paths, it also necessitates a re-evaluation of academic integrity and the cognitive processes involved in problem-solving. If students rely on AI to generate solutions without engaging in the underlying logical construction, there is a risk of "superficial learning." Therefore, understanding the balanced role of AI is crucial. This article aims to explore how AI can be strategically utilized to enhance the comprehension of programming fundamentals, moving the pedagogical focus from rote syntax memorization to higher-order system design and algorithmic efficiency. By analyzing the current state of AI-driven educational tools, this study highlights the shift towards a more adaptive, inclusive, and effective methodology for training the next generation of software engineers.

2. AI as an Intelligent Tutoring System (ITS): A New Era of Pedagogy

The concept of Intelligent Tutoring Systems (ITS) represents one of the most significant applications of AI in computer science education. Traditional pedagogical models often follow a "one-size-fits-all" approach, which fails to account for the diverse cognitive paces and prior knowledge levels of students. An AI-driven ITS functions as a sophisticated, non-linear educational tool that simulates the behavior of a human tutor by providing personalized instruction and immediate feedback.

Educator Insights: A Journal of Teaching Theory and Practice
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6964

2.1. The Architecture of AI Tutoring in Programming

In the context of teaching programming fundamentals, a robust ITS is built upon four interconnected models:

The Domain Model: This contains the core knowledge of the programming language (e.g., C++ or Python syntax, control structures, and data types).

The Student Model: The AI tracks the learner's progress, identifying specific areas of struggle, such as recursion or pointer logic, and builds a unique profile for every user.

The Pedagogical Model: This acts as the "teacher," deciding when to provide a direct answer, when to offer a subtle hint, and when to introduce a more complex challenge.

The Communication Interface: A natural language processing (NLP) layer that allows students to ask questions in plain English and receive human-like explanations.

2.2. Scaffolding and the Zone of Proximal Development

The primary importance of an AI tutor in programming is its ability to provide "scaffolding." According to Vygotsky's theory of the Zone of Proximal Development (ZPD), effective learning occurs when a student is challenged just beyond their current ability but receives enough support to succeed. AI tutors achieve this by:

Real-time Error Analysis: Instead of a generic "Syntax Error" message, the ITS analyzes the student's code intent. For instance, if a student misses a terminating condition in a loop, the AI might prompt: "Your loop is running, but what happens when the counter reaches the end of the array?"

Cognitive Load Management: By handling the routine aspects of debugging (like finding missing semicolons), the AI allows the student to focus their cognitive energy on higher-level algorithmic design.

2.3. Dynamic Path Adaptation

Unlike static Massive Open Online Courses (MOOCs), an AI-based ITS does not follow a fixed sequence. If a student demonstrates mastery in "Variable Declaration" but struggles with "Conditional Logic," the system dynamically rearranges the curriculum. This adaptive nature ensures that fast learners are not bored by repetition, while struggling students are not left behind. This level of

Educator Insights: A Journal of Teaching Theory and Practice
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6964

granular personalization was historically impossible to achieve in a classroom with 30 or more students, making AI an indispensable tool for modern programming educators.

2.4. Data-Driven Insight for Human Instructors

The ITS also serves as a bridge to the human instructor. By aggregating data from the "Student Models," the AI can generate heatmaps of class performance. This allows professors to identify systemic misunderstandings early in the semester. For example, if the AI detects that 60% of the class is struggling with "Class Inheritance," the instructor can pivot their next lecture to address this specific conceptual gap, thereby creating a hybrid environment of AI-supported individual practice and human-led conceptual synthesis.

References

1. Alam, A. (2021). Possibilities and Apprehensions in the Landscape of Artificial Intelligence in Education. 2021 International Conference on Computational Intelligence and Computing Applications (ICCICA), 1–8.
2. Becker, B. A., Denny, P., Pettit, R., & Prather, J. (2023). The Impact of Large Language Models on Computer Science Education. ACM Inroads, 14(1), 25–31. https://doi.org/10.1145/3583561
3. Chen, L., Chen, P., & Lin, Z. (2020). Artificial Intelligence in Education: A Review. IEEE Access, 8, 75264–75278.
4. Denny, P., Kumar, V., & Giacaman, N. (2023). Conversational AI, Chatbots, and the Future of Learning to Code. Communications of the ACM, 66(6), 34–37.
5. Goksel, N., & Bozkurt, A. (2020). Artificial Intelligence in Education: Current Insights and Future Perspectives. In S. Misra (Ed.), Handbook of Research on Digital Learning. IGI Global.
6. Humbert, L., & Schwill, A. (2022). Intelligent Tutoring Systems for Programming: A Systematic Mapping Study. Education and Information Technologies, 27(4), 4567–4592.
7. Ismail, N., & Abdullayev, S. (2024). Adaptive Learning Frameworks in IT Education: Integrating AI in Developing Countries. Central Asian Journal of Mathematical Theory and Computer Sciences.

Educator Insights: A Journal of Teaching Theory and Practice
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6964

8. Kasneci, E., Sessler, K., & Küchemann, S. (2023). ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education. Learning and Individual Differences, 103, 102274.
9. Kulkarni, C., & Bernstein, M. S. (2021). Scaling Human-Centered Education through Artificial Intelligence. Journal of Educational Computing Research, 59(4), 812–835.
10. Lau, S. T., & Guo, P. J. (2020). Data-Driven Autocompletion for Code: How Novice Programmers Use AI-Powered Tools. Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.
11. Ouhbi, S., & Pombo, N. (2022). Software Engineering Education: The Role of Artificial Intelligence. IEEE Transactions on Education, 65(3), 320–334.
12. Prather, J., Denny, P., & Leinonen, J. (2023). The Robots Are Here: Navigating the Generative AI Era in Computer Science Education. Proceedings of the 54th ACM Technical Symposium on Computer Science Education.
13. Roll, I., & Wylie, R. (2016). Evolution and Revolution in Artificial Intelligence in Education. International Journal of Artificial Intelligence in Education, 26(2), 582–599.
14. Zawacki-Richter, O., Marín, V. I., & Bond, M. (2019). Systematic Review of Research on Artificial Intelligence Applications in Higher Education. International Journal of Educational Technology in Higher Education, 16(1), 1–27.
15. Zhai, X., Chu, H. E., & Chai, C. S. (2021). A Review of Artificial Intelligence in STEM Education: Progress and Prospects. Educational Technology & Society, 24(3), 101–115.