THE IMPACT OF GENERATIVE AI ON PROGRAMMING SKILLS AMONG UZBEK UNIVERSITY STUDENTS

Sarvinoz Khalilova

Student, Japan Digital University, Tashkent, Uzbekistan

Academic Supervisor: Lochinbek Boboev, PhD, Head of the IT Department, Japan Digital University, Tashkent, Uzbekistan

email@domain.com

Abstract

This article investigates the impact of generative artificial intelligence tools, particularly ChatGPT, on the programming skills of students at higher education institutions in Uzbekistan. The central problem addressed is that a growing number of students tend to copy ready-made code produced by AI systems without understanding its underlying logical and algorithmic structure, a practice that contributes to a measurable decline in independent thinking and problem-analysis skills. To address this challenge, the study proposes an "AI Mentor" pedagogical model that withholds finished code and instead tracks the student's errors, offering only logical hints and directional guidance grounded in scaffolding theory. The proposed methodology comprises a three-stage experimental procedure: diagnostics, guided practice, and a control assessment. The anticipated findings indicate that students working with the AI Mentor model will demonstrate a significant improvement in independent error-detection and logical-reasoning performance compared with students relying on conventional copy-paste practices, suggesting that supervised, dialogue-based AI assistance can preserve and strengthen computational thinking in programming education.

Keywords: *generative artificial intelligence, ChatGPT, programming education, AI Mentor, logical thinking, computer science education, Uzbekistan higher education.*

I. INTRODUCTION

In recent years, generative artificial intelligence technologies — most notably ChatGPT, GitHub Copilot, and comparable tools — have been rapidly penetrating every level of education, including computer science programs in higher education. While these technologies enable students to solve programming tasks quickly and conveniently, their overall impact on the learning process remains insufficiently understood and continues to be the subject of ongoing debate.

International research indicates that ChatGPT can increase students' coding speed and engagement with the learning process, although its effect varies depending on the context and the complexity of the assigned task [1]. In particular, interaction with ChatGPT has been shown to improve students' knowledge level and their ability to write comprehensible, logically structured code; however, the tool has not been found to significantly enhance independent error-correction skills, since ChatGPT frequently supplies a correct solution outright, thereby reducing opportunities for genuine debugging practice [2].

A large-scale quasi-experimental study found that the group of students who used ChatGPT exhibited more debugging activity and more frequent analysis of error messages, although this did not consistently translate into positive learning outcomes [3]. At the same time, other studies emphasize that, alongside advantages such as personalized feedback and interactive instruction, the use of ChatGPT in programming education carries serious risks, including academic dishonesty and a decline in critical thinking [4].

A survey conducted among technical university students found that those who regarded ChatGPT as a valuable resource reported higher motivation and competence; however, this relationship held only when students engaged with the system in a structured and critical manner [5]. Furthermore, a quasi-experimental study among secondary school students confirmed the tutoring potential of ChatGPT in programming education, while noting that most existing research remains at an early, conceptual stage [6]. Students also cite rapid responses, enhanced reasoning,

and increased self-confidence as key benefits of using ChatGPT, while simultaneously expressing concern about a decline in independent working skills [7].

A controlled experiment involving 182 first-year students in the United States and China found that the use of ChatGPT did not have a statistically significant effect on students' performance in practical assignments and midterm assessments, although the researchers noted a moderate effect on critical thinking and problem-solving skills [8]. Similarly, instructor-supervised AI assistance systems are being developed to more effectively manage the help-seeking process, whereby the AI provides step-by-step hints on planning, debugging, and optimization rather than ready-made answers [9].

In the context of Uzbekistan, integrating artificial intelligence into education has been identified as an urgent task at the level of state policy. Pursuant to Resolution No. PQ-4996 of the President of the Republic of Uzbekistan, courses on the practical application of artificial intelligence technologies have been introduced in a number of higher education institutions since the 2021/2022 academic year [10]. At the same time, the results of a national survey indicate an overall positive attitude toward artificial intelligence in Uzbekistan's higher education institutions, alongside serious obstacles such as notable gaps in digital competence, insufficient infrastructure, and the absence of clear ethical and regulatory norms [11]. It has also been noted that a growing gap exists between institutional policy on the use of generative artificial intelligence in distance education and its actual use by students [12].

The analysis of the studies cited above demonstrates that the impact of generative artificial intelligence on programming education is inherently dual in nature: on one hand, it accelerates the learning process and increases student confidence, while on the other, unsupervised use may lead to a decline in independent logical reasoning and problem-analysis skills. The purpose of this study is to develop the "AI Mentor" pedagogical model — designed to address this

problem within the context of Uzbekistan's higher education system by guiding students logically rather than providing finished code — and to substantiate its theoretical foundations.

II. MAIN PART

2.1. Practical Analysis of the Problem

A prevailing trend observed in the teaching of programming disciplines at Uzbekistan's higher education institutions is students' increasing reliance on tools such as ChatGPT when completing practical assignments. In this scenario, a student enters the problem statement into the system and receives ready-made code, without conducting an in-depth analysis of its underlying logic, algorithmic structure, or potential errors. As a result, students are often unable to solve even simple problems during examinations or independent work.

The root cause of this problem is that, in the conventional usage model, artificial intelligence performs the role of the "respondent" while the student remains solely in the role of the "requester." From a pedagogical standpoint, this arrangement contradicts constructivist learning theory, according to which knowledge must be constructed by the learner through active inquiry and the process of making and correcting mistakes. The academic literature likewise emphasizes that when artificial intelligence systems provide complete code outright, they constrain students' opportunities for debugging and independent reasoning [2; 13].

2.2. Operating Mechanism of the "AI Mentor" Pedagogical Model

To address this problem, the present study proposes a pedagogical model referred to as "AI Mentor." The core principle of this model is that the artificial intelligence system never provides the student with finished, fully functioning code. Instead, the system performs the following functions:

• It analyzes the code written by the student and identifies the type of error present (syntactic, logical, or algorithmic);

• Rather than the exact answer to the error, it offers a guiding question or hint, for example: "Is the initial value of this loop chosen correctly?"

• Only after the student has made use of several successive hints does the system provide partial assistance (at the level of pseudocode or structure), while the final code is written by the student independently;

• It poses additional questions (using the Socratic dialogue method) to verify the student's level of understanding.

This approach is based on the "scaffolding" principle recommended in international research, whereby the instructor or system does not provide the student with a ready-made solution but instead guides the student's own process of self-correction [9; 13; 14]. Studies show that both students and instructors prefer assistance in the form of hints and step-by-step plans, since such support reflects the student's current level of knowledge and encourages independent reasoning [14].

2.3. Three-Stage Experimental Process

To evaluate the effectiveness of the proposed model, a three-stage experimental process has been developed. The structure of this process is presented in the table below.

Stage	Name	Description of Activity	Expected Outcome
1	Diagnostics (entry test)	Students independently solve an algorithmic problem; AI Mentor automatically analyzes the type of errors (syntactic, logical, algorithmic).	An initial knowledge profile is formed for each student.
2	Guided practice (with AI Mentor)	No finished code is provided; the system offers only questions, hints, and logical direction (e.g., "Is the loop's termination condition correct?").	The student identifies and corrects errors independently; independent reasoning is activated.

Stage	Name	Description of Activity	Expected Outcome
3	Control assessment (outcome evaluation)	A new problem is solved independently, without AI assistance; the result is compared with Stage 1.	The growth in programming skills and logical reasoning is measured quantitatively.

The rationale underlying this experimental process is that the student's initial state is established in the first stage, independent work guided by "AI Mentor" is organized in the second stage, and the outcome achieved is verified through an independent, AI-free assessment in the third stage. This three-stage structure is consistent with the quasi-experimental designs employed in prior studies, in which control and experimental groups are compared [3; 8].

2.4. Comparative Analysis with the Traditional Approach

The table below presents a comparative overview of the key differences between the traditional "copy-paste" approach and the proposed "AI Mentor" model.

Criterion	Traditional "copy-paste" approach	"AI Mentor" pedagogical model
Type of response	Complete, ready-made code	Hints, questions, direction
Student involvement	Passive (copying)	Active (independent reasoning)
Understanding of errors	Minimal	Cause of error is explained
Long-term effect	Risk of skill decline	Development of logical thinking

The table demonstrates that the "AI Mentor" model transforms the student from a passive consumer into an active participant. This, in turn, addresses concerns

raised in international research — namely, that students may simply copy AI-generated solutions, bypassing the process of computational thinking altogether [15]. It has also been shown that example-based, step-by-step guidance systems can provide practically useful assistance to students while preserving their capacity for independent reasoning [13].

2.5. Practical Recommendations for the Context of Uzbekistan's Higher Education

The implementation of the "AI Mentor" model within Uzbekistan's higher education system requires several institutional conditions. First, faculty members must undergo specialized training in the use of AI-based pedagogical tools, as national survey results indicate that the digital competence of higher education staff remains insufficient at present [11]. Second, clear ethical and methodological guidelines governing the use of artificial intelligence must be developed, which would help narrow the gap between institutional policy and students' actual practices [12]. Third, it is advisable to introduce the "AI Mentor" system into national curricula gradually, beginning with pilot projects.

III. CONCLUSION

This study has provided a theoretical analysis of the impact of generative artificial intelligence, particularly ChatGPT, on the programming skills of students at Uzbekistan's higher education institutions. The findings indicate that unsupervised use of artificial intelligence solely for the purpose of obtaining ready-made code may lead to a decline in students' logical and algorithmic reasoning skills. At the same time, international research confirms that artificial intelligence, when applied through an appropriate pedagogical approach, can have a substantial positive effect on the learning process [1; 6; 9].

To resolve this tension, the proposed "AI Mentor" pedagogical model — a system that assists students through logical hints and guiding questions rather than ready-made answers — enables the effective and pedagogically sound use of artificial intelligence in programming education. The three-stage experimental

process (diagnostics, guided practice, and control assessment) provides the foundation for implementing this model in practice and for objectively evaluating its effectiveness.

Future research should test the "AI Mentor" model under real conditions at Uzbekistan's higher education institutions, quantitatively measure its effectiveness across various programming disciplines (such as object-oriented programming and data structures), and examine student and instructor attitudes toward the model. In conclusion, artificial intelligence in programming education should neither be entirely rejected nor used without oversight; rather, it should be integrated as an "intelligent mentor" that strengthens students' capacity for independent reasoning.

REFERENCES

[1] I. Aviv, M. Leiba, H. Rika, and Y. Shani, "The Impact of ChatGPT on Students' Learning Programming Languages," in Learning and Collaboration Technologies, P. Zaphiris and A. Ioannou, Eds., Lecture Notes in Computer Science, vol. 14724, Cham: Springer, 2024. https://doi.org/10.1007/978-3-031-61691-4_14

[2] "Learning-by-teaching with ChatGPT: The effect of a teachable ChatGPT agent on programming education," arXiv preprint, 2024. https://arxiv.org/pdf/2412.15226

[3] "Would ChatGPT-facilitated programming mode impact college students' programming behaviors, performances, and perceptions? An empirical study," International Journal of Educational Technology in Higher Education, vol. 21, no. 12, 2024. https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-024-00446-5

[4] "Integrating ChatGPT in a Computer Science Course: Students' Perceptions and Suggestions," arXiv preprint, 2024. https://arxiv.org/pdf/2402.01640

[5] “The perceived impact of artificial intelligence on academic learning,” PMC (National Library of Medicine), 2025. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC12531233/

[6] “The effectiveness of ChatGPT in assisting high school students in programming learning: evidence from a quasi-experimental research,” Interactive Learning Environments, Taylor & Francis, 2025. https://www.tandfonline.com/doi/full/10.1080/10494820.2025.2450659

[7] “Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning,” ScienceDirect, 2023. https://www.sciencedirect.com/science/article/pii/S2949882123000051

[8] “Computer Science Education in the ChatGPT Era: Experiences from an Experiment in a Programming Course for Novice Programmers,” Mathematics (MDPI), vol. 12, no. 5, p. 629, 2024. https://www.mdpi.com/2227-7390/12/5/629

[9] “Closing the Loop: An Instructor-in-the-Loop AI Assistance System for Supporting Student Help-Seeking in Programming Education,” arXiv preprint, 2025. https://arxiv.org/pdf/2510.14457

[10] “Uzbekistan’s strategy for the development of artificial intelligence technologies until 2030,” Digital Watch Observatory, 2025. https://dig.watch/resource/uzbekinstans-strategy-for-the-development-of-artificial-intelligence-technologies-until-2030

[11] “The Future of Higher Education in Uzbekistan: The Potential and Challenges of Integrating Artificial Intelligence (National Survey),” Perspectives of Higher Education Development, 2025. https://erasmus.uz/journal/index.php/files/article/view/129

[12] “Generative AI Governance in Distance Higher Education: Aligning Uzbekistan’s Digital Learning Methodology with International Standards,” 2026. https://www.pstjournal.uz/index.php/pst/article/view/125

[13] “CodeExemplar: Example-Based Scaffolding for Introductory Programming in the GenAI Era,” arXiv preprint, 2026. https://arxiv.org/pdf/2603.23830

[14] “Scaffolding Metacognition in Programming Education: Understanding Student-AI Interactions and Design Implications,” arXiv preprint, 2025. https://arxiv.org/pdf/2511.04144

[15] “MazeMate: An LLM-Powered Chatbot to Support Computational Thinking in Gamified Programming Learning,” arXiv preprint, 2025. https://arxiv.org/pdf/2511.03727