© IIJP 2026 | Volume 2, Issue 3, pp. 575-597 | ISSN: 2636-4484

# PERCEIVED IMPACT OF GENERATIVE ARTIFICIAL INTELLIGENCE (AI) TOOLS ON PROGRAMMING EDUCATION AMONG UNDERGRADUATE COMPUTER SCIENCE STUDENTS IN NIGERIAN UNIVERSITIES

***Samuel Okechukwu Nnaji[1], Christabel Linda Uchenwa[2], Ndukanma Ngozika Agape[3] & Anyalebechi Felicia Nneamaka[4]**

*[1,2]Department of Computer Science, Cyber Security and Software Engineering, Federal University of Environment and Technology, Koroma/Saakpenwa-Ogoni, Rivers State.*

*[3,4]Department of Computer and Robotics Education, University of Nigeria, Nsukka.*

****Corresponding Email:** samuelokechukwunnaji@gmail.com*

**ORCID: https://orcid.org/0009-0007-5344-3058*

**ARTICLE INFO**

**Article No.:** 0459

**Accepted Date:** 10/07/2026

**Published Date:** 31/07/2026

**Type:** Research

**ABSTRACT**

This study examined the perceived impact of Generative Artificial Intelligence (AI) tools on programming education among undergraduate Computer Science students in Nigerian universities. Four research purposes, four research questions, and two null hypotheses guided the study. A descriptive survey research design was adopted. The target population comprised 100,000 undergraduate Computer Science students from selected Nigerian universities, while a sample of 341 students was selected using a multi-stage sampling technique. Data were collected using a structured instrument titled Generative AI and Programming Education Questionnaire (GAIPEQ), administered in both printed and Google Forms formats to facilitate wider participation. The instrument was validated by three experts from the University of Nigeria, Nsukka, and its reliability was established using Cronbach's Alpha, yielding coefficients of 0.81, 0.78, 0.83, and 0.86 for the four clusters, with an overall reliability coefficient of 0.82. Data were analyzed using Mean and Standard Deviation to answer the research questions, while the hypotheses were tested using the independent samples t-test and One-Way Analysis of Variance (ANOVA) at the 0.05 level of significance with SPSS version 25. The findings revealed that students used Generative AI tools to a moderate extent (Mean = 3.44), perceived them as supporting programming skills acquisition (Mean = 3.90), identified notable challenges associated with their use (Mean = 3.93), and endorsed strategies for promoting their responsible use (Mean = 3.84). The study concludes that Generative AI can enhance programming education in Nigerian universities when integrated responsibly. It recommends incorporating AI literacy into Computer Science curricula, establishing ethical guidelines, and promoting balanced use of AI alongside traditional programming instruction.

**Keywords:** Impact, Generative Artificial Intelligence, Programming Education, Academic Integrity, Nigerian Universities.

## Introduction

Programming education is a foundational component of computer science training. It is an essential pathway to employability in the digital economy and a critical skill for software development, artificial intelligence, cybersecurity, data science, and related computing disciplines. Undergraduate students are expected to develop skills in algorithmic thinking, code design, debugging, software development, and computational problem-solving (Dwivedi, et al., 2023). However, in many Nigerian universities, programming instruction faces major obstacles, including large class sizes, insufficient laboratory equipment, intermittent electricity supply, limited internet access, and overburdened lecturers (Zviel-Girshin, 2024). These challenges often reduce the quality of learning and restrict students' opportunities for meaningful practice.

The emergence of Generative Artificial Intelligence (GenAI) tools has created new opportunities for teaching and learning programming (Elenwo, 2025). Generative AI refers to artificial intelligence systems capable of creating content such as text, code, images, and solutions based on user prompts. Tools such as ChatGPT, GitHub Copilot, Amazon CodeWhisperer, Gemini, and Claude have gained widespread popularity among students due to their ability to generate programming code, explain concepts, detect errors, and provide instant feedback. For students who struggle with programming syntax or logic, this can be highly beneficial (Adewumi & Akinola, 2024). At the same time, there are growing concerns that excessive reliance on AI-generated solutions may weaken critical thinking, encourage plagiarism, and undermine the development of independent programming competence.

In the Nigerian context, the adoption of GenAI tools is especially significant because students often seek self-directed learning alternatives outside the classroom. Many undergraduate computer science students already use mobile devices and online platforms to supplement their studies. Generative AI tools now extend this support by offering conversational, personalized, and immediate assistance (Almulla, 2023). Yet the educational impact of these tools depends on how they are used, regulated, and integrated into academic practice. The use of AI tools in tertiary education is still at an early stage in developing nations like Nigeria. Although several institutions have begun to adopt AI-powered technologies such as virtual tutors, automated grading systems, and adaptive learning platforms, integration across universities remains limited. Although many universities have begun to integrate AI-assisted technologies and digital learning platforms into their educational systems, issues such as poor infrastructure, unreliable internet access, insufficient funding, and poor digital literacy still impede widespread adoption (Elenwo, 2025). Notwithstanding these restrictions, more and more Nigerian students are using AI-based technologies for academic purposes, such as research assistance, content production, online education, and exam preparation. This paper investigates the impact of generative AI tools on programming education among undergraduate computer science students in Nigerian universities and provides recommendations for effective integration into higher education.

## Statement of the Problem

The rapid development and broad adoption of Generative Artificial Intelligence (AI) tools like GitHub Copilot, Gemini, Claude, and CodeWhisperer, and ChatGPT have profoundly changed programming education through instant code generation, debugging, code explanation, and personalised learning support. Such tools can help students to improve their programming skills, learning efficiency and problem solving ability. But the growing use of AI tools has also sparked concerns among educators and researchers about the risks of over-reliance on AI solutions, reduced critical thinking, less practice with programming, academic dishonesty, and the potential for students to gain superficial rather than deep programming knowledge. Undergraduate Computer Science students in Nigerian universities are expected to develop strong programming competence to meet the national and global workforce demands.

However, there is little empirical evidence as to how these AI tools are perceived to influence programming education. Hence, it remains to be seen whether students perceive Generative AI as a facilitator of meaningful learning or as an obstacle in acquiring core programming skills. Generative AI tools are becoming widely available to the university students and they are also becoming popular, but there is little research that has looked into their perceived impact in Nigerian higher education. The present research on AI has been mainly focused on its adoption or acceptance or even educational applications in general, but not enough attention has been given to programming education and the unique challenges faced by undergraduate Computer Science students in Nigerian Universities. This knowledge gap hinders educators, curriculum developers, university administrators, and policy makers from formulating evidence-based strategies and guidelines for the responsible integration of Generative AI in programming instruction. Therefore, there is a need to investigate the perceived impact of Generative Artificial Intelligence tools on programming education among undergraduate Computer Science students in Nigerian universities to provide empirical evidence to inform teaching practices, curriculum development, and institutional policies that maximise the educational benefits of AI while mitigating its potential drawbacks.

**Objective of the Study**

The study seeks to:

1. Examine the extent of Generative AI tool usage among undergraduate Computer Science students in Nigerian universities.
2. Investigate the influence of AI tools on programming skills acquisition among undergraduate Computer Science students in Nigerian universities.
3. Identify challenges associated with the use of Generative AI in programming education
4. Determine the strategies perceived by students as capable of enhancing the responsible use of Generative AI tools.

**Research Questions**

The following research questions guide the study:

1. To what extent do undergraduate Computer Science students use Generative AI tools?
2. How do Generative AI tools affect students' programming skills/competence acquisition?
3. What challenges arise from the use of AI-assisted programming education tools?
4. What strategies are perceived by undergraduate Computer Science students as capable of enhancing the responsible use of Generative AI tools in programming education in Nigerian universities?

**Null Hypotheses**

**$H_{01}$**: There is no significant difference in the mean ratings of undergraduate Computer Science students based on gender regarding the perceived impact of Generative Artificial Intelligence (AI) tools on programming skills/competence acquisition in Nigerian universities.

**$H_{02}$:** There is no significant difference in the mean ratings of undergraduate Computer Science students based on level of study regarding the strategies for enhancing the responsible use of Generative Artificial Intelligence (AI) tools in programming education in Nigerian universities.

**Literature Review**

**Generative Artificial Intelligence (AI) in Education**

Generative AI refers to machine learning systems capable of producing text, code, images, audio, and other forms of content in response to prompts. In education, GenAI can act as a tutor, assistant, brainstorming partner, or feedback tool. For programming education, these tools can generate sample code, explain errors, propose alternative solutions, and support code review. Artificial Intelligence (AI) has emerged globally in recent years, impacting multiple

sectors, including education. Its interdisciplinary nature has led to varied definitions. Shrivastava (2024) defines AI as a branch of computer science developing systems that simulate and extend human intelligence, while De Zúñiga et al. (2023) describe AI as the capacity of machines to perform tasks, solve problems, communicate, and make decisions with varying autonomy and performance. Despite no universally accepted definition of intelligence, scholars in psychology and computer science continue to explore its core attributes.

Across tertiary institutions, students commonly use tools such as ChatGPT, Grammarly, QuillBot, and Snapchat AI for assignments, studying, idea generation, and information retrieval (Ngonso et al., 2025; Orok et al., 2024). AI use has been linked to improved comprehension, writing and communication skills, and personalized learning experiences (Nwile et al., 2025; Omeh et al., 2025). Studies from Jigawa State and Lagos State University of Education highlight benefits such as improved accessibility, adaptability, collaboration, and learning outcomes, while also identifying challenges related to ethics, data security, and shortages of technically skilled personnel (Omar et al., 2024; Fasinro et al., 2024). Similar concerns are evident internationally, with universities in Ireland and Singapore revising assessment practices and questioning AI-assisted grading practices (McDermott, 2025). Overall, while AI is becoming deeply embedded in Nigerian university education, maximizing its long-term benefits will require improved training, equitable access, robust ethical frameworks, and sustained human oversight.

**Programming Education**

Programming education is the teaching of students how to convert problems into computational logic and to create solutions using programming languages such as Python, Java, C++, or JavaScript. In addition to syntax knowledge, logical reasoning, debugging skills, problem decomposition, and algorithmic thinking are all necessary for effective programming education. Python programming, Java programming, C programming, web development, data structures and algorithms, software engineering, and other programming courses are frequently offered. Logical reasoning, inventiveness, and problem-solving skills are necessary for success in programming education (Zastudil et al., 2024).

With recent research concentrating on comprehending instructional techniques, technology integration, and student results in higher education environments, programming education is still a crucial field of study. In Frontiers in Education, Vinueza-Morales et al. (2025) highlight important developments in teaching programming, highlighting the expanding influence of digital technologies and new teaching techniques intended to improve student abilities. This thorough analysis emphasizes how crucial it is to evaluate cutting-edge research publications, pinpoint knowledge gaps, and match curricula to industry demands. According to the analysis, current research on programming education is focusing more on how teachers may successfully incorporate cutting-edge technology into the classroom while upholding pedagogical rigour and meeting the requirements of a diverse student body (Vinueza-Morales et al., 2025).

The integration of artificial intelligence and large language models represents a significant emerging frontier in programming education. Elnaffar, Rashidi, and Abualkishik (2025), demonstrates that AI-powered tools including chatbots, generative AI systems, and intelligent tutoring systems offer substantial benefits such as personalized feedback and improved learning outcomes. However, these tools also present challenges including setup barriers and concerns about academic integrity. Concurrently, Zhu, Xu, and Ericson's (2025) systematic review of large language models in programming education (2023-2024) documents both the opportunities of LLMs for personalized learning and problem-solving efficiency, as well as persistent concerns regarding model limitations and biases. Both reviews advocate for instructional frameworks that emphasize critical skills like prompt engineering and human oversight, pointing toward a future where programming education integrates AI tools

thoughtfully while maintaining educational integrity and fostering genuine student learning (Elnaffar et al., 2025; Zhu et al., 2025).

**Influence of AI tools on Programming Skills Acquisition**

By giving students intelligent support for code generation, debugging, code explanation, syntax correction, and personalised feedback, artificial intelligence (AI) technologies have grown to be a significant part of programming education. Students may practise programming concepts more effectively, spend less time looking for solutions, and get instant explanations for programming problems thanks to tools like ChatGPT, GitHub Copilot, and other AI-powered coding assistants. These features encourage the development of useful programming skills including algorithm creation, code optimisation, debugging, and problem-solving while boosting students' enthusiasm, confidence, and engagement. When incorporated into well-designed instructional activities, recent empirical evidence shows that AI-assisted learning significantly improves programming performance and task completion efficiency, making AI an effective supplement to traditional programming instruction rather than a replacement for it (Güner & Er, 2025; Alanazi et al., 2025).

Despite these benefits, how students use AI tools will determine how much of an impact they have on learning programming abilities. The development of basic programming skills like logical reasoning, algorithmic thinking, debugging, and autonomous problem-solving may be hampered by an over-reliance on AI-generated code. Research has demonstrated that although AI tools increase productivity and coding speed, they do not always result in deeper conceptual knowledge unless they are used in conjunction with instructor instruction, active learning, and critical evaluation of AI-generated answers. In order to guarantee that students develop long-lasting programming skills in addition to AI literacy, programming educators are urged to implement pedagogical strategies that integrate AI responsibly by stressing coding fundamentals, reflective practice, and ethical AI use (Zviel-Girshin, 2024; Kohen-Vacs et al., 2025).

**Positive Impacts of Generative AI on Programming Education**

According Diana, Paul, David and Minh (2025), Generative AI tools are lowering the barrier to entry for learning to code, much as earlier innovations like high-level and block-based programming languages did before them by lowering the technical expertise necessary to create working programs and bringing computation to more people. A systematic review of 58 peer-reviewed studies from 2022–2025 found that the primary reported benefits of AI agents in programming education include enhanced programming support, motivational and emotional benefits for students, and increased efficiency for educators, with generative AI being the most frequently studied category of tool. Research with high school students has similarly found that Generative AI-assisted programming can significantly enhance computational thinking compared to traditional lecture-based instruction, suggesting benefits that extend beyond simple code production into higher-order reasoning skills (Diana, Paul, David and Minh, 2025).

Beyond skill-building, generative AI is reshaping how instructors design courses and assess learning. Studies have explored using large language models to generate personalized, contextually relevant programming exercises and to provide automated feedback on student code explanations, while other work has benchmarked LLMs' ability to assess multiple-choice question quality in introductory programming courses (Prather et al., 2024). Vocational and university-level studies have also linked GAI-supported programming instruction to improved learning motivation, self-efficacy, and broader competencies when the tools are integrated thoughtfully into course design (Li et al., 2025). Taken together, this body of research suggests generative AI can act as a scalable teaching assistant expanding access, personalizing instruction, and freeing instructor time for higher-value mentoring.

**Negative Impacts of Generative AI on Programming Education**

Alongside these benefits, a growing body of research documents real harms, particularly for novice programmers. A widely cited ICER 2024 study titled "The Widening Gap: The Benefits and Harms of Generative AI for Novice Programmers" found that AI coding tools can disproportionately help already-strong students while leaving weaker students further behind, widening rather than closing skill gaps (Prather et al., 2024). A 2026 experimental study similarly warns that unstructured GenAI adoption can introduce pedagogical misalignment, shallow cognitive engagement, and ethical risks that threaten the sustenance of students' programming skills, noting that without deliberate pedagogical scaffolding, GenAI use can drift toward answer extraction and passive acceptance of generated outputs, encouraging overreliance and weakening learners' independent reasoning.

This overreliance risk is compounded by concerns about superficial understanding: beginners who lean on GenAI tools may develop only a shallow grasp of underlying programming concepts rather than internalizing problem-solving logic (Elnaffar, S., et al., 2025). A dedicated systematic review, "Beyond the Benefits: A Systematic Review of the Harms and Consequences of Generative AI in Computing Education" (2025), catalogs additional concerns raised across the field, including academic integrity violations, erosion of debugging and problem-decomposition skills, and instructors' difficulty moderating AI-generated content in student-facing forums (Qiao et al., 2025). Instructors themselves have grappled with this shift since ChatGPT's release, moving from an initial instinct to ban the tools toward acceptance that resistance is largely futile, forcing a broader rethink of assessment design, academic integrity policy, and how programming fundamentals are taught in an AI-saturated environment (Lau & Guo, 2023).

Despite the growing adoption of Generative Artificial Intelligence (AI) tools in programming education worldwide, there is limited empirical evidence on how undergraduate Computer Science students in Nigerian universities perceive their use, educational impact, associated challenges, and responsible integration into programming instruction. Existing studies have largely focused on the technical capabilities of AI tools or have been conducted in developed countries, leaving a contextual gap regarding the Nigerian higher education environment. This study fills that gap by providing empirical evidence from 341 undergraduate Computer Science students across Nigerian universities, offering a comprehensive assessment of the extent of AI tool usage, their perceived impact on programming skills and competence acquisition, the challenges encountered, and the strategies required to promote their responsible use. The findings contribute context-specific knowledge that can guide educators, university administrators, curriculum developers, and policymakers in developing effective policies and instructional practices for the ethical and productive integration of Generative AI into programming education in Nigerian universities.

**Constructivist Learning Theory (Jean Piaget, 1936)**

This study is anchored on the Constructivist Learning Theory, originally proposed by Jean Piaget in 1936 and later expanded by Lev Vygotsky in the 1970s through his sociocultural perspective on learning. Constructivist Learning Theory posits that learners actively construct knowledge rather than passively receiving information from instructors. According to Piaget, learning occurs through processes of assimilation and accommodation, whereby individuals integrate new experiences into existing cognitive structures and modify those structures when necessary. Vygotsky further emphasized the importance of social interaction, collaboration, and guided support in the learning process through concepts such as the Zone of Proximal Development (ZPD) and scaffolding. The theory therefore views learning as an active, learner-centered process in which knowledge is built through exploration, problem-solving, interaction, and reflection.

The relevance of Constructivist Learning Theory to this study lies in its explanation of how undergraduate Computer Science students can utilize Generative AI tools to enhance programming education. Generative AI tools such as ChatGPT, GitHub Copilot, Gemini, and Claude provide interactive learning environments where students can ask questions, receive instant feedback, generate code examples, and obtain explanations of programming concepts. These features support the constructivist principle that learners build understanding through active engagement with learning resources. Furthermore, AI tools function as digital scaffolds that assist students in solving programming problems that may be beyond their current level of competence, thereby aligning with Vygotsky's concept of the Zone of Proximal Development. By facilitating personalized learning, immediate feedback, and collaborative human-computer interaction, Generative AI tools can improve programming competence, problem-solving skills, and independent learning among undergraduate Computer Science students in Nigerian universities. Consequently, Constructivist Learning Theory provides a suitable framework for understanding the educational impact of Generative AI tools on programming learning outcomes.

The findings indicate that Generative AI tools have a significant influence on undergraduate Computer Science students' programming skills and competence acquisition by supporting code generation, debugging, problem-solving, and self-directed learning. These findings are well explained by Constructivist Learning Theory, which posits that learners actively construct knowledge through interaction, exploration, and problem-solving. When used responsibly, Generative AI serves as a cognitive support tool that promotes active learning, conceptual understanding, and the development of programming competence rather than replacing students' own thinking and learning processes.

**Methodology**

The study adopts a descriptive survey research design. According to Creswell & Creswell, (2018) and Saunders, Lewis, & Thornhill, (2019), a descriptive survey is a quantitative research design used to systematically collect, describe, and analyze information about the characteristics, opinions, attitudes, behaviors, or conditions of a defined population without manipulating any variables. This design is particularly appropriate when the researcher seeks to describe existing conditions rather than establish cause-and-effect relationships. Because it allows the researcher to gather information from a large number of undergraduate computer science students and describe their experiences, perceptions, and patterns of using Generative AI tools in programming education without changing any variables, a descriptive survey design is appropriate for this study. Additionally, it provides reliable data for describing the perceived effects of Generative AI tools in Nigerian universities. Nigerian Universities as an area of study, comprise federal owned, state-owned, and private owned universities established to provide teaching, research, innovation, and community service. These institutions play a vital role in producing skilled graduates, advancing scientific knowledge, and supporting national development. As of recent years, Nigeria has over 290 approved universities distributed across the six geopolitical zones, providing access to higher education for millions of students across diverse academic disciplines.

The population comprises 95,229 undergraduate Computer Science students enrolled in Nigerian universities, based on the programme enrolment statistics published by the National Universities Commission (NUC) in the 2019 Nigerian University System Statistical Digest. The enrolment includes 92,731 full-time and 2,498 part-time undergraduate students studying Computer Science across accredited Nigerian universities. There is no officially published national figure specifically for the current population of undergraduate Computer Science students across all Nigerian universities from 2020 till date by the NUC. Base on that, the researcher used 100,000 targeted undergraduate students of Computer Science. The sample size for this study consisted of three hundred and ninety eight (398) undergraduate computer

science students using multi-stage sampling. The sample size was obtained by using statistical formula derived by Taro Yamane shown as;

$$n = \frac{N}{1+N(e)2}$$

Where:

N = population
e = sampling error
n = sample size
N = 100,000, e = 5% (0.05)

$$n = \frac{100,000}{1+100,000(0.05)2}$$

$$n = \frac{100,000}{1+250}$$

$$n = \frac{100,000}{251}$$

$$n = 398.40637$$

$$n \sim 398$$

A multi-stage sampling technique was used for the study. In the first stage, the researcher used simple random sampling technique to draw 3 geopolitical zones out of the 6 geopolitical zones in Nigeria. In the second stage, the researcher adopted stratified random sampling technique. This technique was used to draw 3 states from each of the 3 selected geopolitical zones, making a total of 9 states. In the third stage, purposive sampling was used to select 3 universities from each of the 3 selected states, making it 9 universities actually available in each selected state. In the fourth stage, accidental sampling was used to draw the 398 respondents from those universities.

Data collection was carried out using a structured questionnaire titled: Generative AI and Programming Education Questionnaire (GAIPEQ). The questionnaire was designed in Google Form format to facilitate convenient distribution and response collection from the students who are spread across the country. It consisted of two sections: Section A, gathering demographic data (e.g., age, gender, and level), and Section B, capturing responses on the four research questions. In Section B, respondents indicated their level of agreement with statements relevant to the study's objectives on a 5-point Likert scale ranging from Very High Extent to Very Low Extent for research question one, Strongly Agree to Strongly Disagree for research question 2 to 4.

Face and Content Validity: To confirm that the items align with the research questions, the instrument was reviewed by three specialists; two Professor from Department of Computer Science and one Doctor from Measurement and Evaluation, all from the University of Nigeria, Nsukka. Reliability: A pilot test involving 30 students from an institution in a nearby state was carried out in order to assess internal consistency. The data were analyzed using Cronbach Alpha Reliability Coefficient with the aid of SPSS version 25, which produced a coefficient of 0.81, 0.78, 0.83 and 0.86 for clusters 1, 2, 3 and 4 respectively and 0.82 for all the clusters, demonstrating good reliability. Data were collected through both physical administration of the questionnaire and online Google Forms to reach a wider audience across the institutions. This hybrid approach ensured a high response rate and allowed for ease of data cleaning.

Data collected using the questionnaire were analyzed using Mean and Standard Deviation. Different values were assigned to the 5-point Likert scale adopted for this research. These are: Very High Extent (VHE) – 5, High Extent (HE) – 4, Moderate Extent (ME) – 3, Low Extent (LE) – 2, Very Low Extent (VLE) - 1 for research question one and Strongly Agree (SA) – 5, Agree (A) – 4, Undecided (UD) – 3, Disagree (D) – 2, Strongly Disagree – 1. The decision rule was 3.00, being the criterion mean for a 5-point scale that is (5+4+3+2+1) divided

by 5. Any mean score of 3.00 and above was regarded as High Extent of Use, High Effect, High Challenge and Highly Strategy with the statement, while mean scores below 3.00 were treated as Low Extent of Use, Low Effect, Low Challenge and Low Strategy. At the .05 level of significance, the null hypotheses were analyzed using t-test and One-Way Analysis of Variance (ANOVA). The null hypotheses were upheld (accepted) if the significant value (p-value) was equal to or greater than the α-value (.05); if not, they were discarded (rejected). The analysis was done using the Statistical Package for Social Sciences (SPSS) version 25.

**Results**

**Table 1: Questionnaire Distribution Analysis** ***N = 398***

| S/N | Questionnaire | Number | Percentage (%) |
|---|---|---|---|
| 1 | Questionnaire Administered | 398 | 100 |
| 2 | Questionnaire Returned | 355 | 89.2 |
| 3 | Questionnaire Unreturned | 43 | 10.8 |
| 4 | Questionnaire Rejected | 14 | 3.5 |
| 5 | Questionnaire Used for Analysis | 341 | 85.7 |

**Source:** *Field Survey, 2026*

Table 1 shows that the total copies of the questionnaire administered were 398, showing 100%; 355 copies of the questionnaire representing 89.2% were returned, 43 copies representing 10.8% were not returned; 14 copies representing 3.5% of the questionnaire were rejected. The 341 copies of the questionnaire representing 85.7% were correctly filled and returned were used for analysis.

**Demographic Characteristics of the Respondents**

**Table 2: Age of Respondents**

| Age | Frequency | Percentage (%) |
|---|---|---|
| 16 – 20 years | 138 | 40.5 |
| 21 – 25 years | 116 | 34.0 |
| 26 & Above | 87 | 25.5 |
| **Total** | **341** | **100** |

**Source:** *Field Survey, 2026*

Data analysis in Table 2 above showed that 138(40.5%) respondents were of 16-20years, 116(34.0%) respondents were of 21- 25years and 87(25.5%) respondents were of 26years and above.

**Table 3: Gender of Respondents**

| Gender | Frequency | Percentage (%) |
|---|---|---|
| Male | 143 | 41.9 |
| Female | 198 | 58.1 |
| **Total** | **341** | **100** |

**Source:** *Field Survey, 2026*

Data analysis in Table 3 above showed that 143(41.9%) respondents were males while 198(58.1%) respondents were females.

**Table 4: Marital Status of Respondents**

| Level (year) | Frequency | Percentage (%) |
|---|---|---|
| Year 1 | 83 | 24.3 |
| Year 2 | 96 | 28.3 |
| Year 3 | 72 | 21.1 |
| Year 4 | 90 | 26.4 |
| **Total** | **341** | **100** |

**Source:** *Field Survey, 2026*

Data analysis in Table 4 above showed that 83(24.3%) respondents were in year one, 96(28.3%) respondents were in year two; 72(21.1%) respondents were in year three while 90(26.4%) respondents were in year four.

**Research Question**

**Research Question 1:** To what extent do undergraduate Computer Science students use Generative AI tools?

**Table 5:** *Mean and standard deviation on the extent undergraduate Computer Science students use Generative AI tools* ***N = 341***

| S/N | Items | $\bar{x}$ | SD | Remark |
|---|---|---|---|---|
| 1 | I use Generative AI tools (e.g., ChatGPT, Gemini, Claude, GitHub Copilot) to learn programming concepts. | 4.12 | 0.64 | High Extent |
| 2 | I use Generative AI tools outside classroom hours for self-directed programming practice and learning. | 2.64 | 0.92 | Low Extent |
| 3 | I frequently integrate suggestions and code generated by AI tools into my academic programming activities. | 2.80 | 0.88 | Low Extent |
| 4 | I use Generative AI tools to improve the quality and efficiency of my programming work. | 4.35 | 0.79 | High Extent |
| 5 | I use Generative AI tools to obtain solutions to coding exercises and practical tasks. | 2.57 | 0.81 | Low Extent |
| 6 | I use Generative AI tools to generate source code for programming assignments and projects. | 3.20 | 0.62 | High Extent |
| 7 | I use Generative AI tools to debug and correct errors in my programs. | 2.91 | 0.77 | Low Extent |
| 8 | I rely on Generative AI tools to explain complex programming algorithms and techniques. | 4.10 | 0.73 | High Extent |
| 9 | I use Generative AI tools to learn new programming languages and frameworks. | 3.41 | 0.70 | High Extent |
| 10 | I use Generative AI tools when preparing for programming examinations and tests. | 4.27 | 0.72 | High Extent |
| | **Overall Mean** | **3.44** | **0.76** | |

**Source:** *Field Survey, 2026*

From the data presented in Table 5 above, it was revealed that the overall mean of the responses of the respondents on the 10 items in the table is 3.44 on 5-point rating scale. This indicated that undergraduate Computer Science students use Generative AI tools to a Moderate Extent, since 3.44 falls below the criterion mean of 3.00. The standard deviation values of the 10 items ranged from 0.62 to 0.92 which indicated that the responses of the respondents were fairly close to the mean.

**Research Question 2**: How do Generative AI tools affect students' programming skills/competence acquisition?

**Table 6:** *Mean and standard deviation on how Generative AI tools affect students' programming skills/competence acquisition* ***N = 341***

| S/N | Items | $\bar{x}$ | SD | Remark |
|---|---|---|---|---|
| 11 | The use of Generative AI tools has improved my understanding of programming concepts and principles. | 3.60 | 0.63 | Agree |
| 12 | The use of Generative AI tools has improved my ability to learn new programming languages independently. | 4.01 | 0.71 | Agree |
| 13 | Generative AI tools have enhanced my coding productivity and efficiency during practical programming exercises. | 3.59 | 0.66 | Agree |
| 14 | Generative AI tools encourage me to explore different approaches to solving programming problems. | 4.02 | 0.57 | Agree |
| 15 | Generative AI tools have increased my confidence in developing software applications. | 3.93 | 0.61 | Agree |
| 16 | The explanations provided by Generative AI tools have strengthened my understanding of algorithms and data structures. | 3.70 | 0.73 | Agree |
| 17 | Generative AI tools help me identify and correct programming errors more effectively. | 4.32 | 0.64 | Agree |
| 18 | Using Generative AI tools has improved my problem-solving skills in programming tasks. | 3.49 | 0.81 | Agree |
| 19 | Generative AI tools have enhanced my ability to write accurate and efficient program code. | 4.51 | 0.94 | Agree |
| 20 | Overall, Generative AI tools have positively contributed to my acquisition of programming skills and competencies. | 3.80 | 0.99 | Agree |
| | **Overall Mean** | **3.90** | **0.72** | |

**Source:** *Field Survey, 2026*

The data presented in Table 6 revealed that the overall mean of the responses of the respondents on the 10 items in the table is 3.90 on 5-point rating scale. This indicated that the respondents agreed that Generative AI tools have a positive effect on programming skills acquisition. The standard deviation values of the 10 items ranged from 0.57 to 0.99 which indicated that the responses of the respondents were fairly close to the mean.

**Research Question 3:** What challenges arise from the use of AI-assisted programming education tools?

**Table 7:** *Mean and standard deviation on the challenges arising from the use of AI-assisted programming education tools* ***N = 341***

| S/N | Questionnaire Items | $\bar{x}$ | SD | Remark |
|---|---|---|---|---|
| 21 | AI-generated solutions sometimes make it difficult for me to fully understand the logic behind a program. | 4.13 | 0.69 | Agree |
| 22 | Frequent use of AI tools reduces my motivation to learn programming concepts in depth. | 3.99 | 0.91 | Agree |
| 23 | I find it difficult to determine whether AI-generated programming solutions are completely accurate. | 4.64 | 0.63 | Agree |
| 24 | AI-assisted programming tools may encourage academic dishonesty, such as submitting AI-generated assignments as personal work. | 3.68 | 0.83 | Agree |
| 25 | I sometimes depend too much on AI-assisted programming tools instead of solving programming problems on my own. | 4.10 | 0.92 | Agree |
| 26 | Limited internet connectivity affects my ability to effectively use AI-assisted programming tools. | 3.19 | 0.68 | Agree |
| 27 | The subscription costs of some AI programming tools restrict my access to their advanced features. | 4.21 | 0.94 | Agree |
| 28 | AI-assisted programming tools occasionally provide incorrect or misleading code suggestions. | 3.22 | 0.85 | Agree |
| 29 | Concerns about data privacy and security discourage me from fully utilizing AI-assisted programming tools. | 4.40 | 0.87 | Agree |
| 30 | The use of AI-generated code reduces my ability to think critically about programming solutions. | 3.74 | 0.74 | Agree |
| | **Overall Mean** | **3.93** | **0.81** | |

**Source:** *Field Survey, 2026*

Also, data presented in Table 7 above showed that the overall mean of the responses of the respondents on the 10 items in the table is 3.93 on 5-point rating scale. This indicated that the respondents agreed that several significant challenges are associated with the use of AI-assisted programming education tools. The standard deviation values of the 10 items ranged from 0.63 to 0.94 which indicated that the responses of the respondents were fairly close to the mean.

**Research Question 4:** What strategies are perceived by undergraduate Computer Science students as capable of enhancing the responsible use of Generative AI tools in programming education in Nigerian universities?

**Table 8:** *Mean and standard deviation on the strategies perceived by undergraduate Computer Science students as capable of enhancing the responsible use of Generative AI tools in programming education* ***N = 341***

| S/N | Items | $\bar{x}$ | SD | Remark |
|---|---|---|---|---|
| 31 | Students should receive regular training on the responsible and ethical use of Generative AI tools for programming tasks. | 3.27 | 0.65 | Agree |
| 32 | Programming assignments should require students to explain and justify AI-generated code to demonstrate understanding. | 4.23 | 0.77 | Agree |
| 33 | Universities should develop clear institutional policies on the ethical use of Generative AI tools in programming education. | 3.61 | 0.59 | Agree |
| 34 | Programming lecturers should provide guidance on when and how Generative AI tools should be used in coding assignments. | 4.34 | 0.69 | Agree |
| 35 | Collaboration among the National Universities Commission (NUC), universities, lecturers, and students is essential for promoting the responsible use of Generative AI tools in programming education. | 3.71 | 0.79 | Agree |
| 36 | Programming assessments should include practical coding exercises conducted under supervised conditions to promote independent problem-solving skills. | 3.24 | 0.66 | Agree |
| 37 | Workshops, seminars, and awareness campaigns should be organized regularly to educate students on the benefits and risks of Generative AI tools. | 4.52 | 0.75 | Agree |
| 38 | Students should be encouraged to use Generative AI tools as learning assistants rather than as substitutes for independent | 4.00 | 0.83 | Agree |
| 39 | Universities should use plagiarism and AI-detection tools to discourage misuse of Generative AI in programming assignments. | 3.44 | 0.64 | Agree |
| 40 | Courses on AI ethics and academic integrity should be incorporated into the Computer Science curriculum. | 4.06 | 0.94 | Agree |
| | **Overall Mean** | **3.84** | **0.73** | |

**Source:** *Field Survey, 2026*

From the data presented in Table 8 above, it was revealed that the overall mean of the responses of the respondents on the 10 items in the table is 3.84 on 5-point rating scale, indicating that the respondents were only marginally above the Undecided point of the scale on whether these strategies would enhance the responsible use of Generative AI tools in programming education. The standard deviation values of the 10 items ranged from 0.59 to 0.94 which indicated that the responses of the respondents were fairly close to the mean.

**Hypothesis One**

**$Ho_1$**: There is no significant difference in the mean ratings of undergraduate Computer Science students based on gender regarding the perceived impact of Generative Artificial Intelligence (AI) tools on programming skills/competence acquisition in Nigerian universities.

***Table 9:*** **Independent Samples t-test on Gender (N = 341)**

| Gender | N | Mean | SD | df | p-value | Decision |
|---|---|---|---|---|---|---|
| Male | 143 | 3.68 | 3.68 | 339 | 0.202 | **Not Significant** |
| Female | 198 | 3.59 | 3.59 | | | |

**Source:** *Field Survey, 2026*

Since p = 0.202 > 0.05, the null hypothesis is not rejected. There is no statistically significant difference in the mean ratings of undergraduate Computer Science students based on gender regarding the perceived impact of Generative AI tools on programming skills/competence acquisition in Nigerian universities.

**Hypothesis Two**

**$Ho_2$**: There is no significant difference in the mean ratings of undergraduate Computer Science students based on level of study regarding the strategies for enhancing the responsible use of Generative Artificial Intelligence (AI) tools in programming education in Nigerian universities.

***Table 10:*** **One-Way Analysis of Variance (ANOVA)**

| Source | SS | df | MS | F | p-value | Decision |
|---|---|---|---|---|---|---|
| Between Group | 2.01 | 4 | 0.503 | | | |
| Within Groups | 91.83 | 336 | 0.273 | 1.84 | 0.121 | Not Significant |
| Total | 93.84 | 340 | | | | |

**Source:** *Field Survey, 2026*

Since p = 0.121 > 0.05, the null hypothesis is not rejected. There is no statistically significant difference in the mean ratings of undergraduate Computer Science students based on their level of study regarding the strategies for enhancing the responsible use of Generative AI tools in programming education in Nigerian universities.

**Discussion of Findings**

The findings of this study in respect to research question one showed that undergraduate Computer Science students use Generative AI tools to a Moderate Extent. This finding agrees with Kasneci et al. (2023) and Dwivedi et al. (2023) that undergraduate Computer Science students increasingly use Generative AI tools such as ChatGPT, GitHub Copilot, Gemini, and Claude to support their programming education. These tools assist students in generating source code, explaining programming concepts, debugging errors, writing documentation, and providing step-by-step solutions to coding problems. By offering instant feedback and personalized learning support, Generative AI tools help students improve their understanding of programming languages, algorithms, data structures, and software development practices. As a result, many students use these technologies to complement classroom instruction, complete assignments, prepare for examinations, and develop practical programming skills. The use of Generative AI tools among undergraduate Computer Science students has transformed the learning process by making programming education more interactive and accessible. Students can experiment with different coding approaches, receive real-time explanations, and learn independently at their own pace. However, the effectiveness of these tools depends on how responsibly they are used. While appropriate use can enhance problem-solving abilities, creativity, and coding proficiency, excessive dependence on AI-generated solutions may reduce critical thinking, programming practice, and independent problem-solving skills. Therefore, students are encouraged to use Generative AI as a learning

aid rather than a substitute for developing their own programming competence (Kasneci et al., 2023; Tlili et al., 2023).

Also, research question two found that the respondents agreed that Generative AI tools have a positive effect on programming skills acquisition. This finding aligns with Kasneci et al. (2023) and OpenAI (2024) that generative AI tools are artificial intelligence systems capable of producing programming-related content such as source code, debugging suggestions, code explanations, documentation, and software design recommendations from natural language prompts. Popular tools such as GitHub Copilot, ChatGPT, and Amazon Code Whisperer have become increasingly integrated into programming education by assisting students in writing, testing, and understanding code. These tools can positively influence programming skills by providing instant feedback, reducing the time spent on syntax errors, demonstrating alternative coding approaches, and supporting personalized learning. As a result, students can strengthen their problem-solving abilities, improve code comprehension, and gain confidence in applying programming concepts to practical tasks. Despite these benefits, excessive dependence on Generative AI tools may hinder genuine competence acquisition if students rely on AI-generated solutions without understanding the underlying programming principles. When integrated responsibly as a learning support tool rather than a substitute for practice and reasoning, it can enhance programming competence while preserving students' ability to develop independent coding skills (Dwivedi et al., 2023; Kasneci et al., 2023).

The result obtained for research question three showed that the respondents agreed that several significant challenges are associated with the use of AI-assisted programming education tools. In line with this finding, Kasneci et al., (2023) suggested that the challenges arising from the use of AI-assisted programming education tools refer to the various academic, technical, ethical, and pedagogical difficulties encountered when students and educators integrate artificial intelligence technologies into programming instruction. Although AI-powered tools such as code generators, automated debuggers, and intelligent tutoring systems can improve learning efficiency, they may also encourage overdependence, reducing students' ability to develop critical thinking, problem-solving, and independent coding skills. Students who rely excessively on AI-generated solutions may struggle to understand underlying programming concepts, algorithm design, and debugging techniques, which are essential competencies for computer science education. AI-generated code is not always correct, secure, or optimized, and students with limited programming experience may find it difficult to identify errors or misleading outputs (Cotton, Cotton, & Shipway, 2024). The unequal availability of premium AI tools also creates disparities in learning opportunities among students. Moreover, institutions face difficulties in designing fair assessment methods that distinguish between students' original work and AI-assisted solutions while maintaining academic standards. These challenges highlight the need for responsible AI usage policies, digital literacy training, and instructional strategies that encourage students to use AI as a learning aid rather than a replacement for genuine programming practice (UNESCO, 2023).

Finally, the finding in Table 4 shows that the respondents were undecided, rather than in strong agreement, that these strategies would enhance the responsible use of Generative AI tools in programming education. This finding is in agreement with UNESCO (2023) that responsible use of Generative AI tools in programming education refers to the adoption of appropriate policies, ethical practices, and instructional approaches that encourage students to use AI as a learning aid rather than a substitute for independent thinking and problem-solving. Effective strategies include establishing clear institutional guidelines on acceptable AI use, promoting academic integrity, integrating AI literacy into the curriculum, and training students to critically evaluate AI-generated code. Universities should encourage collaborative learning, emphasize transparency by requiring students to disclose AI assistance in assignments, and foster digital responsibility through regular workshops and awareness campaigns. By

combining ethical guidelines, learner-centered teaching practices, and appropriate assessment strategies, institutions can maximize the educational benefits of Generative AI while minimizing risks such as overdependence, inaccurate code generation, and academic misconduct (OECD, 2023).

**Conclusion**

The findings of this study indicate that undergraduate Computer Science students in Nigerian universities make moderate to high use of Generative Artificial Intelligence (AI) tools in programming education, primarily for code generation, debugging, code explanation, learning new programming concepts, and completing programming tasks. The study further revealed that these tools have a predominantly positive impact on students' programming skills and competence acquisition by improving problem-solving ability, programming efficiency, conceptual understanding, coding confidence, and independent learning. Nevertheless, the study identified several challenges associated with AI-assisted programming education, including overdependence on AI-generated solutions, reduced critical thinking and debugging skills, concerns about academic integrity and plagiarism, inaccurate or misleading code suggestions, and unequal access to reliable AI technologies. To maximize the educational benefits of Generative AI, the respondents perceived that responsible use could be enhanced through AI literacy training, the development of institutional policies and ethical guidelines, curriculum integration of AI literacy, lecturer supervision, assessment methods that emphasize practical coding competence, and improved digital infrastructure. Overall, the 341 undergraduate Computer Science students who participated in the study reported that Generative AI tools are valuable learning resources whose benefits can be sustained when accompanied by responsible usage practices, effective institutional support, and balanced pedagogical approaches that promote genuine programming competence rather than excessive dependence on AI-generated solutions.

**Recommendations**

Based on the foregoing discussion, the following recommendations are proposed:

1. Undergraduate Computer Science students use Generative AI tools to a moderate extent in programming education for code generation, debugging, concept explanation, and learning support. Nigerian universities should integrate approved Generative AI tools into the Computer Science curriculum through structured classroom activities, practical programming exercises, and institutional guidelines that encourage their educational use while discouraging overdependence. University Management and Computer Science Departments.
2. Generative AI tools have a positive effect on undergraduate students' programming skills and competence acquisition by improving understanding of programming concepts, problem-solving ability, coding efficiency, and debugging skills. Computer Science lecturers should incorporate AI-assisted programming activities into teaching and assessment, ensuring that students use AI as a learning aid while also completing independent programming tasks that strengthen critical thinking and coding competence. Computer Science Lecturers and Academic Departments.
3. The use of AI-assisted programming education tools presents challenges such as excessive dependence on AI-generated code, reduced critical thinking, academic integrity concerns, inaccurate code suggestions, and limited opportunities for independent problem-solving. Universities should establish comprehensive policies and training programmes on the ethical and responsible use of Generative AI, including academic integrity regulations, AI literacy workshops, and assessment methods that evaluate students' original programming abilities alongside AI-supported learning. National Universities Commission (NUC) and University Senates.

4. Undergraduate Computer Science students perceive that responsible AI use can be enhanced through AI ethics education, clear institutional guidelines, practical programming exercises, lecturer supervision, and continuous training on AI literacy. The National Universities Commission should develop national standards and curriculum guidelines for the responsible integration of Generative AI into Computer Science education, while universities implement regular capacity-building programmes for lecturers and students on ethical AI use, prompt engineering, verification of AI-generated code, and responsible programming practices. National Universities Commission (NUC).

## References

Adewumi, S. O., & Akinola, A. O. (2024). Artificial intelligence and higher education in Africa: Opportunities and challenges. African Journal of Educational Technology, 12(1), 45–60

Alanazi, M., Soh, B., Samra, H., & Li, A. (2025). The influence of artificial intelligence tools on learning outcomes in computer programming: A systematic review and meta-analysis. Computers, 14(5), 185. https://doi.org/10.3390/computers14050185

Almulla, M. A. (2023). The effectiveness of artificial intelligence applications in education. Education and Information Technologies, 28(5), 5601–5620

Cotton, D. R. E., Cotton, P. A., & Shipway, J. R. (2024). Chatting and cheating: Ensuring academic integrity in the era of ChatGPT. Innovations in Education and Teaching International, 61(2), 228–239

Creswell, J. W., & Creswell, J. D. (2018). Research design: Qualitative, quantitative, and mixed methods approaches (5th ed.). SAGE Publications.

De Zúñiga, H., Goyanes, M., & Durotoye, T. (2023). A Scholarly Definition of Artificial Intelligence (AI): Advancing AI as a Conceptual Framework in Communication Research. Political Communication, 41, 317 – 334

Dwivedi, Y. K., Hughes, L., Baabdullah, A. M., Ribeiro-Navarrete, S., Giannakis, M., Al-Debei, M. M., et al. (2023). So what if ChatGPT wrote it? Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. International Journal of Information Management, 71, 102642

Elenwo, P.M. (2025). Constraints of Artificial Intelligence Integration in the 21st Century Human Resource Management in Higher Institutions in Nigeria. International Journal of Educational Management, Rivers State University. 1(1), 278-287

Elnaffar, S., Rashidi, F., & Abualkishik, A. Z. (2025). Teaching with AI: A systematic review of chatbots, generative tools, and tutoring systems in programming education. arXiv, 2510.03884. https://doi.org/10.48550/arXiv.2510.03884

Fasinro, K. S., Udofia, G. I., Akinkuotu, F. A., & Aremu, V. I. (2024). Impact of Artificial Intelligence on the Learning Assessment of Students in Tertiary Institutions in South-West, Nigeria. West African Journal of Open and Flexible Learning, 13(1), 113-134

Frontiers in Computer Science (2026). An experimental study of structured generative AI integration to mitigate pedagogical, cognitive, and ethical barriers in programming education. https://doi.org/10.3389/fcomp.2026.1789829

Güner, H., & Er, E. (2025). AI in the classroom: Exploring students' interaction with ChatGPT in programming learning. Education and Information Technologies, 30, 12681–12707. https://doi.org/10.1007/s10639-025-13337-7

Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., et al. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103, 102274

Kohen-Vacs, D., Usher, M., & Jansen, M. (2025). Integrating generative AI into programming education: Student perceptions and the challenge of correcting AI errors. International

Journal of Artificial Intelligence in Education, 35(5), 3166–3184. https://doi.org/10.1007/s40593-025-00496-4

Lau, S., & Guo, P. J. (2023). From "Ban It Till We Understand It" to "Resistance is Futile." ICER 2023. https://doi.org/10.1145/3568813.3600138

Li, H.-J., Huang, Q.-R., Wen, L.-P., Chen, W., & Xu, Z.-Z. (2025). Generative Artificial Intelligence Supported Programming Learning: Learning Effectiveness and Core Competence. SAGE Journals.

McDermott, R. (2025, September 9). 'It's a monster': How generative AI is forcing university professors to rethink learning. The Irish Times. https://www.irishtimes.com/ireland/education/2025/09/09/its-amonster-how-generative-ai-isforcing-university-professors-torethink-learning/

Ngonso, B., Egielewa, P., Egenti, G., Uduehi, I., Sunny-Duke, F., Ukhurebor, K., & Osemengbe, U. (2025). Influence of Artificial Intelligence on educational performance of Nigerian students in tertiary institutions in Nigeria. Journal of Infrastructure, Policy and Development, 9(1), 9949

Nwile, C. B., Amaewhule, E. C., & Ezeugo, C. R. (2025). Utilization of Artificial Intelligence Tools for Improving Educational Outcomes of Students in Public Universities in Rivers State, Nigeria. African Journal of Humanities and Contemporary Education Research, 19(1), 195-209

OECD. (2023). Generative AI in the classroom: From hype to reality? Paris: Organisation for Economic Co-operation and Development.

Omar, B. A., Babale, A., & Bukar, Z. S. (2024). Impact of Artificial Intelligence (AI) On Tertiary Education In Jigawa State, Nigeria. Journal of Association of Educational Management and Policy Practitioners, 6(2), 203-209

Omeh, C. B., Olelewe, C. J., & Ohanu, I. B. (2025). Impact of Artificial Intelligence Technology on Students' Computational and Reflective Thinking in a Computer Programming Course. Computer Applications in Engineering Education, 33(3), e70052

OpenAI. (2024). GPT models and educational applications. OpenAI Research Reports.

Orok, E., Okaramee, C., Egboro, B., Egbochukwu, E., Bello, K., Etukudo, S., & Akawa, O. (2024). Pharmacy students' perception and knowledge of chatbased artificial intelligence tools at a Nigerian University. BMC Medical Education, 24(1), 1237

Piaget, J. (1936). The origins of intelligence in children (M. Cook, Trans.). International Universities Press.

Prather, J., Reeves, B. N., Leinonen, J., MacNeil, S., Randrianasolo, A. S., Becker, B. A., Kimmel, B., Wright, J., & Briggs, B. (2024). The Widening Gap: The Benefits and Harms of Generative AI for Novice Programmers. ICER 2024. https://doi.org/10.1145/3632620.3671116

Qiao, S., Denny, P., & Giacaman, N. (2025). Oversight in Action: Experiences with Instructor-Moderated LLM Responses in an Online Discussion Forum. ACE 2025. https://doi.org/10.1145/3716640.3716651

Saunders, M., Lewis, P., & Thornhill, A. (2019). Research methods for business students (8th ed.). Pearson.

Shrivastava, A. (2024). Artificial Intelligence (AI): Evolution, Methodologies, and Applications. International Journal for Research in Applied Science and Engineering Technology.

Tlili, A., Shehata, B., Adarkwah, M. A., Bozkurt, A., Hickey, D. T., Huang, R., & Agyemang, B. (2023). What if the devil is my guardian angel: ChatGPT as a case study of generative artificial intelligence in education? Smart Learning Environments, 10(1), 15.

UNESCO. (2023). Guidance for Generative AI in Education and Research. Paris: UNESCO.

Vinueza-Morales, M., Rodas-Silva, J., Vidal-Silva, C., Córdova-Morán, J., & Cevallos-Ayón, E. (2025). Teaching programming in higher education: A bibliometric analysis of trends, technologies, and pedagogical approaches. Frontiers in Education, 10, 1525917. https://doi.org/10.3389/feduc.2025.1525917

Vygotsky, L. S. (1978). Mind in society: The development of higher psychological processes. Harvard University Press.

Zastudil, C., Rogalska, M., Kapp, C., Vaughn, J., & MacNeil, S. (2024). Teachers' and students' perceptions of AI-generated concept explanations: Implications for integrating generative AI in computer science education. Computers and Education: Artificial Intelligence, 7, 100283

Zhu, M., Xu, L., & Ericson, B. (2025). A systematic review of research on large language models for computer programming education. arXiv, 2506.21818. https://doi.org/10.48550/arXiv.2506.21818

Zviel-Girshin, R. (2024). The good and bad of AI tools in novice programming education. Education Sciences, 14(10), 1089. https://doi.org/10.3390/educsci14101089