Students Speak Out: The Purposes, Impacts, and Concerns of Generative AI in Programming Activities

تأثير استخدام أداوت الذكاء الاصطناعي التوليدي في أنشطة البرمجة بين الأهداف والتحديات: دراسة استكشافية في تجارب الطلاب

أ. عهود دخيل الله الرحيلي	أ. نهى الرحيلي	أ. تغريد عبد الله الشنقيطي
محاضر كلية علوم وهندسة الحاسب جامعة طيبة	محاضر كلية إدارة الاعمال جامعة طيبة	محاضر كلية إدارة الأعمال جامعة طيبة
المملكة العربية السعودية	المملكة العربية السعودية	المملكة العربية السعودية
Tshenqiti@taibahu.edu.sa	Nrohily@taibahu.edu.sa	Orohili@taibahu.edu.sa

Abstract:

This study explores undergraduate students' perceptions regarding the use of AI tools in coding activities, with a focus on how often they use these tools, for what purposes, how accurate they perceive them to be, and how they believe these tools affect their learning and their ability to solve problems. This study also examines students' concerns about overreliance on AI tools and their use of AI-generated code without full comprehension of coding. A survey of 319 undergraduate students revealed that they regularly use AI tools for coding, primarily for debugging. While consistent use of AI tools improved the students' understanding of programming concepts, problem-solving abilities, and coding skills, they also led to more errors in the generated code. Interestingly, although most students did not use AI-generated code without full understanding of coding, those who perceived a negative effect of using these tools on their programming skills were more likely to do so. Additionally, some students expressed concerns that these tools could get in the way of their long-term learning and believed their use should be regulated. These insights confirm the important role of AI in programming education and the growing need for AI tools' conscientious integration and thoughtful guidance on their usage in academic contexts.

Keywords: Generative AI, Student perception, Programming, education

المستخلص:

تهدف هذه الدراسة إلى تحليل وجهات نظر طلاب المرحلة الجامعية حول استخدام أدوات الذكاء الاصطناعي التوليدي في البرمجة، من خلال استعراض أبرز النشاطات البرمجية التي يتم توظيف أدوات الذكاء فيها، والكشف عن أراء الطلاب حول مدى تأثير هذه الأدوات على تعلمهم وقدرتهم على حل المشكلات، وتسليط الضوء على المخاوف من الآثار المترتبة على الاعتماد على هذه الأدوات. تمثلت أداة البحث في استبانة وقد شملت عينة الدراسة ٣١٩ طالب في المرحلة الجامعية. أظهرت النتائج أن معظم الطلاب يستخدمون هذه الأدوات لتصحيح الأخطاء البرمجية في المقام الأول إضافة الى كتابة الأكواد والمساعدة في تفسير الأكواد التي يصعب فهمها. في حين أشار الطلاب أنهم استطاعوا تسخير أدوات الذكاء لمساعدتهم في تطوير مهاراتهم البرمجية وتحسين أدارتهم للوقت الذي يحتاجونه عادة لحل المشكلات، إلا أن نتائج الدراسة كشفت أيضاً أن

الطلاب الذين يعتقدون أن هذه الأدوات تؤثر بشكل سلبي على مهاراتهم البرمجية كانوا غالباً يستخدمون الأكواد المولدة دون فهمها. كما عبر الطلاب في هذه الدراسة عن قلقهم حول تأثير هذه الأدوات على تعلمهم على المدى الطويل، وأبدو دعمهم لفكرة تنظيم استخدامها في المنظومة التعليمية. قدمت الدراسة توصيات أبرزها التأكيد على الحاجة الملحة الى حوكمة استخدام أدوات الذكاء الاصطناعي في العملية التعليمية بشكل مسؤول وأخلاقي.

الكلمات المفتاحية: أدوات الذكاء الاصطناعي التوليدي، وجهة نظر طلابية، البرمجة، التعليم

Introduction:

Artificial intelligence (AI) is evolving at an accelerated pace, bringing significant transformations across many sectors. Education is no exception. In both general and higher education, AI is becoming more influential by shaping students' academic development through a blend of opportunities and challenges (Vieriu & Petrea, 2025). In the context of computer programming, the integration of AI tools (e.g., ChatGPT) in coding has become more prevalent, which has created new opportunities to enhance students' learning experiences (Zviel-Girshin, 2024).

Of the computer education fields, programming is one of the most challenging, as it demands proficiency in logic, syntax, and problem-solving skills that can vary from one student to another (Topalli & Cagiltay, 2018). Traditional programming teaching methods, due to their one-size-fits-all approach and limited customized feedback, can be less beneficial in meeting different learning needs (Cheah, 2020). In contrast, AI tools can adapt to the pace of each student and provide personalized exercises and feedback that cater to the individual strengths and weaknesses of each learner (Jaboob et al., 2025; Silva et al., 2024). These AI-powered features can assist students in understanding complex programming concepts, identifying and correcting errors in their code, and developing problem-solving skills, ultimately promoting an engaging and supportive learning environment customized to their personal pace (Philbin, 2023). However, as discussed in many studies, these tools also come with their challenges, such as overreliance on AI, weakened critical thinking abilities, shallow understanding of programming concepts, and academic misconduct (Silva et al., 2024; Vieriu & Petrea, 2025; Zviel-Girshin, 2024).

In light of this rationale, the potential benefits of using AI tools in programming activities, as well as the implications of their use, highlight the need for research into how undergraduate students currently employ AI tools in their programming courses and how they perceive the influence of these tools on their long-term learning and problem-solving abilities. Therefore, the present study aimed to investigate the frequency of AI tool usage among undergraduate students. Additionally, it sought to identify the specific purposes for which students use AI tools and their perceptions of the tools' accuracy. Moreover, it examined

students' perceived impact of AI tools on their coding skills and problem-solving abilities, as well as their concerns regarding the influence of AI tool usage on their long-term programming competence.

Based on the objectives outlined above, this study investigated the following research questions:

RQ1: To what extent do undergraduate students utilize AI tools in their coding activities?

RQ2: What are students' specific purposes and perceptions of using AI tools in coding activities?

RQ3: How do undergraduate students perceive the impact of AI tools on their coding skills and problem-solving abilities?

RQ4: What are the perceived concerns among students regarding the influence of AI tool usage on their long-term programming competence?

The results of this study provide insight into the current situation of students' use of AI tools in their programming activities and the areas where these tools are used. They also improve the understanding of students' concerns about the perceived influences of using AI tools on their programming skills. Moreover, these findings can help emphasize the need to develop new regulations and ethical guidelines for the use of AI tools in academic settings.

The next section reviews the related literature. The subsequent section describes the methodology, and Section 4 presents and discusses the results. The paper is concluded and recommendations for future research directions in light of the findings are offered in the final section.

1. Related Work:

This section reviews recent studies that investigate undergraduate students' use of AI tools in coding, with a focus on the following key aspects: widespread awareness and adoption; purpose of use, perceptions, and attitudes, impact on learning and performance, and challenges and concerns.

2.1 Widespread Awareness and Adoption:

According to several recent studies, the usage of AI tools in coding activities among students is increasing significantly. According to a study of 251 valid questionnaire responses from Chinese college students, 86.5% of respondents had used AI coding assistant tools, with only 13.5% having only heard of AI programming without ever using it (Pan et al., 2024). A similar study conducted in Europe showed that bachelor students used Generative AI (GenAI)in programming-related tasks and considered it a help-seeking strategy (Keuning et al., 2024). Another study carried out at a university in the United States revealed how students in fundamental programming courses are integrating tools like ChatGPT into their assignments,

المجلة الدولية لتكنولوجيا التعليم والمعلومات ISSN Print: 1658-9556 / ISSN Online: 2961-4023

concluding that students heavily rely on Generative AI tools when solving programming exercises (Ghimire & Edwards, 2024).

2.2 Purpose of Use, Perceptions, and Attitudes:

AI tools in coding can serve several purposes. They enhance the learning experience by offering real-time feedback, explanations, and suggestions to help learners understand complex concepts and improve their coding skills (Chan & Hu, 2023; Sun et al., 2024). In addition, AI tools increase students' productivity and efficiency by generating boilerplate code that they can use and providing them with solutions to complex coding problems (Keuning et al., 2024). Furthermore, Clarke and Konak (2025) asserted that Generative AI can assist learners by streamlining debugging processes by pinpointing errors, suggesting fixes, and explaining the reasoning behind these recommendations. According to many studies (Güner & Er, 2025; Molina et al., 2024; Prather et al., 2025), one of the most perceived benefits of AI tools among students, especially for non-native English speakers, is that they dissolve the language barrier in coding education. Notably, they provide explanations in multiple languages and simplify complex technical jargon, resulting in improved code comprehension.

2.3 Impact on Learning and Performance:

Numerous studies have concluded that AI tools influence the learning process and affect students' overall performance. One study showed an increase in the number of students who believe it ethical to auto-generate a complete solution for an assignment and submit it without thoroughly understanding it (Keuning et al., 2024). In contrast, another study indicated that using AI tools positively influences learning outcomes (Aghiomesi et al., 2024). Fan et al. (2025) explored this area by dividing students into three distinct groups: AI-assisted pair programming, human-human pair programming, and individual programming. The outcomes of this experiment showed that the students using AI-assisted pair programming demonstrated higher intrinsic motivation and lower anxiety levels compared to the other groups, which positively affected their performance. Güner and Er (2025) conducted a similar experiment and reached similar conclusions.

2.4 Challenges and Concerns:

Despite the widespread use of AI tools, researchers continue to delve into the challenges these tools present. Some have reported that students who use AI tools encounter challenges regarding accuracy, engagement, privacy, and academic integrity (Aghiomesi et al., 2024; Chan & Hu, 2023). Fan et al. (2025) found that students' overreliance on AI tools could potentially hinder the development of their critical thinking and independent problem-solving skills. Another major issue when using AI tools is their negative effects on collaborative coding environments and

team dynamics (Pan et al., 2024). As noted in their findings, students who heavily rely on AI tools for coding may miss valuable opportunities to develop essential teamwork skills, which are often the key to thriving in a programming environment.

1.Methodology

This section presents survey structure, data collection methods, and data analysis.

3.1 Survey Structure:

To answer the four research questions, we designed a survey to collect data from undergraduate students at Taibah University, Saudi Arabia and thereby explore their usage and perceptions of AI tools for programming activities. We developed an initial set of 14 items for the students to rate according to a 5-point Likert scale ranging from *strongly disagree* to *strongly agree* or from *never* to *always*. We also created one open-ended question to gather deeper insights from the respondents. To ensure the clarity and relevancy of the survey, we consulted two experts on higher education research in educational technologies. We then carried out a pilot study prior to the formal data collection. Based on the feedback we received, we redefined some of the items for clarity.

The final survey included a set of eleven items and one open-ended question. It covered the following dimensions, with 2–3 items per dimension: utilizing AI tools for code activities, the nature of the coding tasks in which students use AI tools and their perceptions of these tools, the impacts of AI on coding skills and problem-solving abilities, and concerns among students about the influence of AI tool usage on their long-term programming competence.

3.2 Data Collection:

The data were gathered via an online survey distributed in a classroom setting to ensure the integrity of the participants. The survey targeted undergraduate students enrolled in the College of Computer Science and Engineering and the College of Business Administration at Taibah University from the fall 2024 to the spring 2025 semesters. Participation was voluntary, and the participants were informed that their responses were anonymous. the recorded responses were anonymous.

3.3 Data Analysis:

This section outlines our data analysis procedures, including filtering of results, calculating correlations, and processing the open-ended question.

3.3.1 Filtering the results:

We followed the best practices to ensure the quality of our data (i.e., Zijlstra et al., 2007). Notably, upon inspecting the responses to the item “How frequently do you use AI tools when coding?” we found that only 1.3% of participants (4 individuals) selected *never*. We filtered out these responses for lack of experience in utilizing AI

المجلة الدولية لتكنولوجيا التعليم والمعلومات ISSN Print: 1658-9556 / ISSN Online: 2961-4023

tools for programming activities. Therefore, the final sample size is 319 participants, including only respondents who have used AI when coding.

3.3.2 Calculating correlations:

To calculate correlations, we utilized the Spearman correlation coefficient, which is advised for use with Likert scales (Murray, 2013).

3.3.3 Processing the open-ended question:

To answer the second research question, we used an open-ended survey question to gather insight on the respondents' experience with coding tasks in which they used AI. We used thematic analysis to group these responses (Fereday & Muir-Cochrane, 2006).

1. Results and Discussion

This section presents the results of the data analysis and their relevance to existing literature.

4.1 Demographic Information:

A total of 319 undergraduate students completed the survey, consisting of 264 females (98.5%) and 4 males (1.5%). The majority of participants were enrolled in STEM fields (99%, 316 individuals) and primarily majored in management of information systems (45.1%, 144 students), artificial intelligence (37.9%, 121 students), computer science (5.6%, 18 students), information systems (8.5%, 27 students), and security (1.9%, 6 students). Contrastingly, only 0.9% of participants (6 students) were enrolled in fields outside of STEM. According to academic level, 14 students (4.4%) were freshmen, 61 (19.1%) were sophomores, 126 (39%) were juniors, and 118 (37%) were seniors.

Additionally, the participants were asked to report their programming experience, which revealed that 23.2% (74 students) had less than 1 year of experience. The majority at 63% (201 students) possessed 1–2 years of experience, while 12.9% (41 students) had 3–5 years of experience and 12.9% (3 students) had more than 5 years of experience. Table 1 presents the participants' full demographic information.

Table 1 Participant Demographic Information

		%	n
Gender	Female	98.5	264
	Male	1.5	4
Academic level	Year 1	4.4	14
	Year 2	19.1	61
	Year 3	39.5	126
	Year 4	37	118
Major	Computer science	5.6	18
	Information systems	8.5	27
	Management information systems	45.1	144

	AI	37.9	121
	Security	1.9	6
	Non-technical	0.9	3
Years of programming experience	Less than 1 year	23.2	74
	1–2 years	63	201
	3–5 years	12.9	41
	More than 5 years	12.9	3

4.2 Research Question 1

RQ1: To what extent do undergraduate students utilize AI tools in their coding activities?

Our first research question focused on understanding the current situation of students' use of AI tools when coding. The results are summarized in Figure 1. Most of the respondents (45.5%) revealed that they use AI tools at least sometimes, with 14.4% responding *always* and 25.1% *often*. Conversely, 25% of respondents fell into the lower frequency category, with 13.8% responding *rarely*. These results demonstrate the spectrum of AI usage among the surveyed undergraduate students. They also align with previous findings that specify that the majority of students utilize AI tools in their coding tasks (Ghimire & Edwards, 2024; Keuning et al., 2024; Pan et al., 2024).

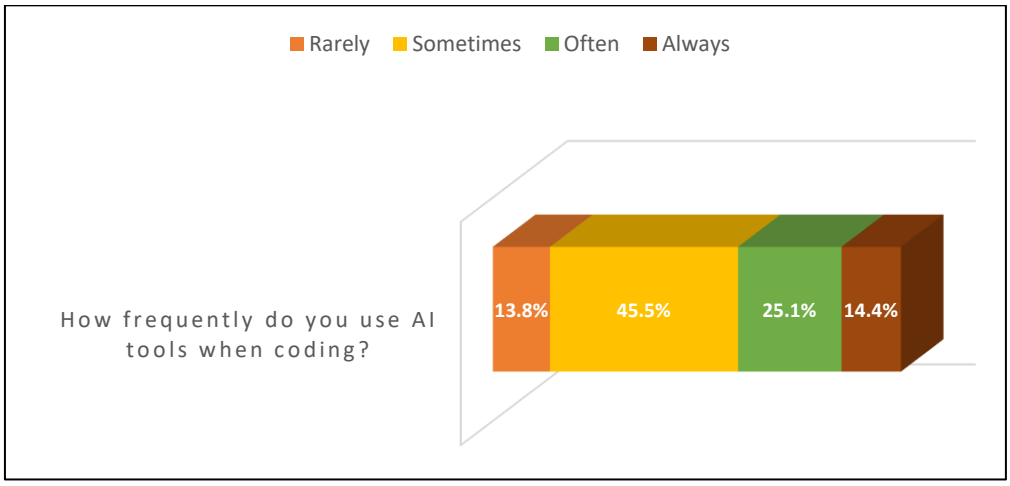


Figure 1 General opinions of respondents about frequency of employing AI tools in coding

4.3 Research Question 2

RQ2: What are students' specific purposes and perceptions of using AI tools in coding activities?

Our second research question discussed the programming activities in which the students use AI tools and the students' perceptions of the accuracy of these tools. First, we examined the students' purposes for using AI tools in the form of an open-ended question ("What are the types of programming tasks in which you use AI

tools?"), taking into account that some students might use the tools for multiple programming activities. The thematic analysis resulted in five distinct themes, grouped according to the answers to the question, indicating varying levels of employment of AI tools across different programming tasks. The responses in Table 2 showcase the number of students whose answers fit each theme.

Table 2 Resulting Themes in Response to the Question, "What Are the Types of Programming Tasks in Which You Use AI Tools?"

Theme	Responses
Generating code snippets	78
Understanding code	258
Optimizing existing code	178
Learning new programming concepts	158
Debugging	285

Notably, 90% of students identified debugging as a primary activity wherein their usage of AI tools was prevalent, followed by asking for AI assistance to explain concepts at 81%. Optimizing existing code was highlighted by 56% of participants, while 50% prioritized learning new programming skills. Surprisingly, 25% mentioned generating code snippets. This finding is consistent with previous studies that mention debugging as the most common purpose for using AI tools in programming courses (Clarke & Konak, 2025; Ghimire & Edwards, 2024; Groothuijsen et al., 2024).

Second, the students were asked to rate the accuracy of AI-generated code. As shown in Figure ٢, the majority of participants (46.7%) sometimes encounter errors in AI-generated code that require significant corrections, while only 7.5% always and 13.5% often do so. Meanwhile, 32.3% reported lower frequencies, with 22.9% selecting *rarely* and 9.4% claiming *never*. Chan and Hu (2023) reported similar results that indicated that more than half of the students they surveyed expressed lack of trust in the correctness and credibility of AI-generated code.

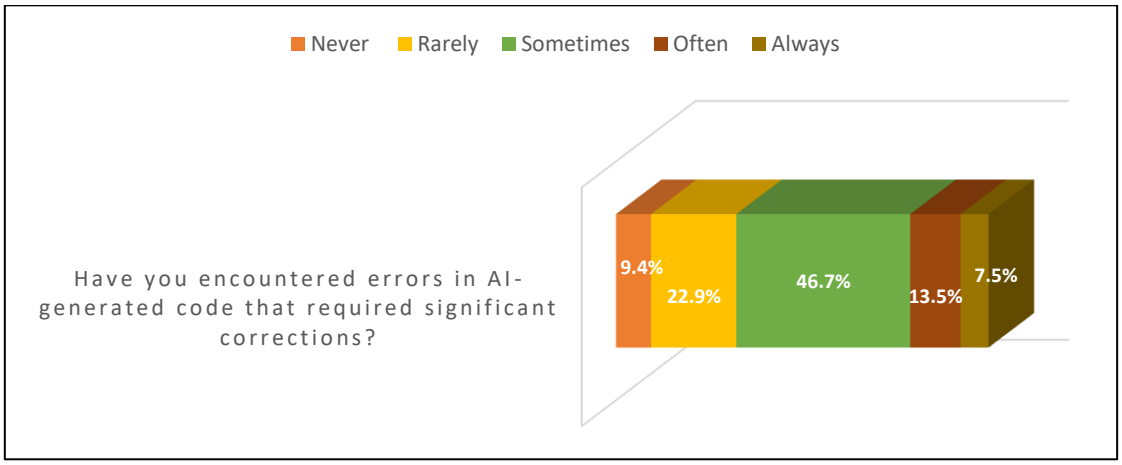


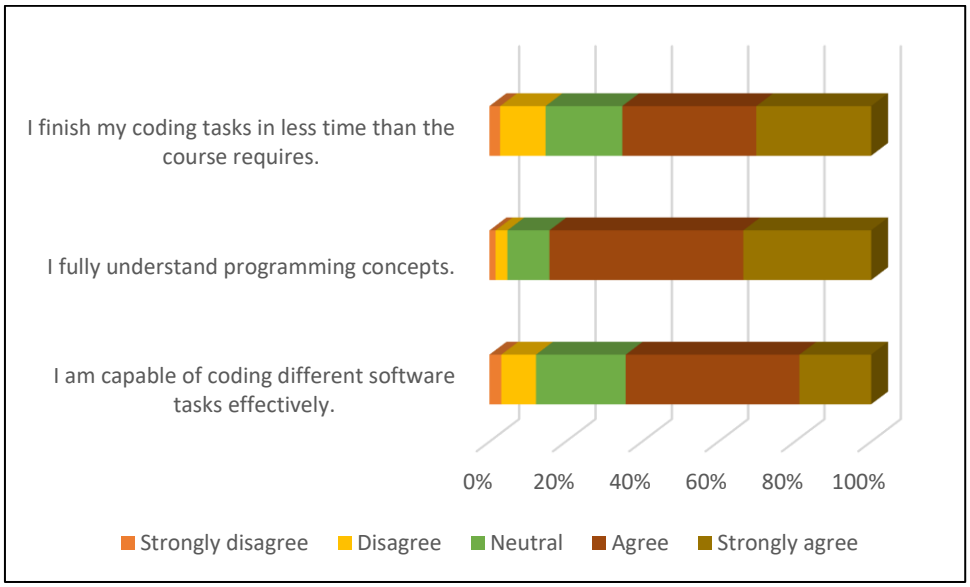
Figure 2 Respondents' perceptions of the accuracy of AI-generated code

A Spearman correlation helped us examine the correlation between the students' frequency of use and perceived accuracy of AI-generated code. The results indicated a very weak positive correlation between the frequency of encountering errors in AI-generated code and the frequency of using AI tools when coding ($R = 0.123$, $p < 0.05$). This suggests that there is a slight tendency for those who use AI tools more often to also encounter errors in AI-generated code. This finding is supported by Tosi (2024), who reported that the more users rely on AI-generated code, the higher the likelihood of encountering errors in that code.

4.4 Research Question 3

RQ3: How do undergraduate students perceive the impact of AI tools on their coding skills and problem-solving abilities?

Mastery of programming concepts allows students to apply them effectively, which can reduce the time needed to solve problems. Moreover, when students efficiently break down problems to quickly address the features associated with different software tasks, this suggests that the students possess the strengths and characteristics of high-quality problem-solving abilities. Therefore, we asked the participants to express their perceptions in the form of Likert scales on (1) their understanding of programming concepts, (2) their ability to code different software tasks effectively, and (3) their ability to finish coding tasks in less than the course-required time. The results are illustrated in Figure 3.



General respondent opinions about the impact of AI tools on their coding skills Figure 3

At 84.3%, the majority of participants felt confident about their comprehension of programming concepts, with 33.5% responding *strongly agree* and about half (50.8%) responding *agree*. In contrast, a mere 15.7% reported feelings of disagreement; only 1.5% strongly disagreed, indicating a high level of confidence overall. This confidence expanded to the students' ability to effectively code different software tasks, with 45.5% responding *agree* and 18.8% *strongly agree*. In contrast, only 12.2% revealed their disagreement, with 9.1% responding *disagree* and 3.1% *strongly disagree*, which suggests overall self-assurance in the surveyed students' comprehension of programming concepts.

Another attribute of students with strong problem-solving abilities that we investigated is ability to finish coding tasks quicker than expected. The findings showed that about two-thirds of participants expressed positivity by selecting either *agree* (30.1%) or *strongly agree* (35%), whereas only 14.7% expressed negative opinions by choosing *disagree* or *strongly disagree*. These results reflect the students' faith in their ability to adapt in coding situations that require problem-solving skills.

Our third research question also focused on the students' understanding of the effect that AI tool usage has on their coding skills and problem-solving abilities. We performed a Spearman correlation to examine the relationship between the frequency of using AI tools when coding and the students' understanding of programming concepts. As presented in Table 3, the results indicated a weak positive relationship between these aspects, suggesting that as usage of AI tools in coding increases, so too does understanding of programming ($R = 0.234$, $p < 0.05$). This is supported by Yilmaz and Yilmaz's (2023) experimental study in which they explored the impact of AI tool usage frequency on students' programming skills, including conceptual understanding. Although the authors noted no direct causal effect between frequency and deeper learning, they highlighted how the use of AI tools can boost self-efficacy and overall comprehension.

Table 3 Spearman Correlations Between Aspects of Coding Skills versus Frequency of Using AI Tools.

Aspects of coding skills	Frequency of AI tool usage
Full understanding of programming concepts	0.234
Coding different software tasks effectively	0.343
Finishing coding tasks in less time than the course requires	0.423

The presented values are statistically significant at $p < 0.05$.

Furthermore, our findings demonstrated a statistically significant weak-to-moderate positive correlation between the frequency of using AI tools in coding and the students' self-perceived coding abilities ($R = 0.343$, $p < 0.05$). This result

also echoes Yilmaz and Yilmaz (2023) by indicating that AI tool usage positively affects students' self-perceived coding ability. Specifically, the more students engage with AI tools, the more confident they feel about their programming skills. In addition, we explored the correlation between the frequency of AI tool usage in coding and the ability to finish coding tasks in less time. These aspects had a statistically significant moderately positive correlation ($R = 0.423$, $p < 0.05$), as presented in Table 3. This implies that more frequent use of AI tools is associated with a tendency for individuals to finish coding tasks in less time, although this relationship is not very strong and other factors are likely involved. Similarly, Ghimire and Edwards (2024) suggested that task completion time decreases considerably for students who frequently used AI tools such as ChatGPT.

4.5 Research Question 4

RQ4: What are the perceived concerns among students regarding the influence of AI tool usage on their long-term programming competence?

Finally, it was critical to understand the concerns students may pose regarding the use of AI tools for their coding activities. To do so, we studied three aspects that may affect students' programming progress: (1) likeliness to employ AI-generated code without completely understanding it, (2) overreliance on AI tools for coding, and (3) the need to regulate the utilization of these tools in academic courses. The results are depicted in Figure ٤.

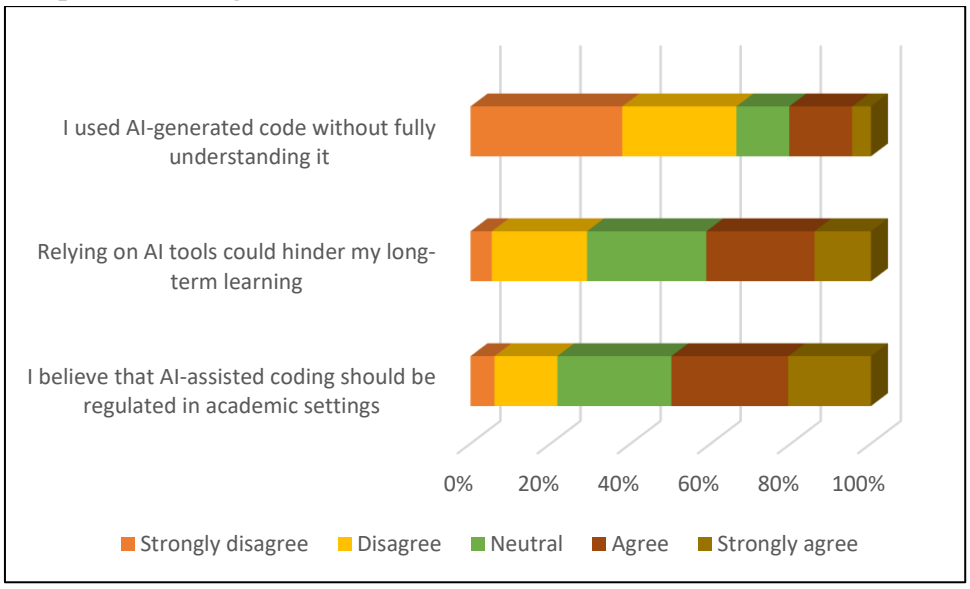


Figure 4 Students' perceived concerns regarding coding while using AI tools

The survey results demonstrated that 37.9% of respondents strongly disagreed with the likelihood to employ AI-generated code without fully comprehending it, and an additional 28.5% disagreed, indicating predominantly comprehensive use of AI-generated code. Markedly, under 5% strongly agreed. Related, we examined

المجلة الدولية لتكنولوجيا التعليم والمعلومات ISSN Print: 1658-9556 / ISSN Online: 2961-4023

whether students were concerned that relying on AI tools may hinder their long-term learning. Among the survey participants, 27% reported *agree* and 14% *strongly agree*, while a significant 29.8% reported *neutral*. This suggests strong overall concern about AI tool usage, with only 5% expressing strong disagreement with the item. Lastly, we assessed the students' opinions regarding the need to regulate the use of AI-assisted code in academic settings. In the survey, 20% of respondents indicated that they *strongly agree* with the statement, while 29% chose *agree*. In contrast, 15.7% disagreed and only 6% strongly disagreed, leaving 28.5% of respondents' neutral.

We then delved into whether the students' concerns about AI tools hindering their long-term programming competence correlates to using AI-generated code, even if they did not completely understand the code. The results showed a moderately positive correlation between the two aspects ($R = 0.456$, $p < 0.05$). This suggests that individuals who are more concerned about the potential negative impact of AI on their programming competence are also more likely to use AI-generated code without fully understanding it. Keuning et al. (2024) raised similar concerns about overreliance on generative AI tools at the expense of foundational learning in programming courses, showing that excessive use of generative AI could hinder the acquisition of core programming skills.

1. Conclusion and Future Research:

This study delved into undergraduate students' minds to explore their perceptions of generative AI tool usage in programming activities. The findings indicated that a significant number of students utilize these tools for various programming activities, with debugging being the most common. The findings also suggested that individuals who employ AI tools in their code more frequently tend to view themselves as having greater programming skills. Furthermore, the results demonstrated students' concerns regarding overreliance on AI tools and the resulting potential negative impact on their long-term learning skills.

Overall, the results revealed valuable insights that set the groundwork for future endeavors to utilize AI tools in the field of programming for different parties, including students, academics, universities, AI tool developers, and researchers.

For students: The raised issues regarding the accuracy of AI tools to generate code should make it clear that AI tools are supporting tools, not a substitute for developers. While they help in performing programming tasks in less time, this does not suggest that they are a substitute for students' basic programming abilities. Additionally, students must be aware that although AI tools can boost their confidence in their ability to code, this does not necessarily imply that these tools possess the underlying skills required to write, understand, or debug code.

Developing programming skills depends on active and reflective use of these tools rather than passive reliance on them.

For academics and universities: This study has proven that the vast majority of students use AI tools for various purposes, regardless of their current academic level. This calls on academics and educators to educate students on the proper mechanism for employing these tools while also highlighting their shortcomings and limitations. Similarly, universities would be wise to issue a regulatory guide for the use of AI tools to maintain academic integrity while giving students the opportunity to leverage them to develop their programming and learning skills.

For AI tools developers: Current generative AI tools have proven to be inaccurate when generating code. This poses a future direction for the builders and developers of these tools to improve AI models to generate more accurate and reliable code.

For researchers: Future research could use evidence (e.g. test scores or coding tasks) to accurately measure students' programming progress. This can be quite helpful in determining whether students' self-perceived of the impact of AI tools on their coding skills and problem-solving abilities match what is really happening.

Referencies:

Aghiomesi, I. E., Olusegun, A. J., Silas, D. O., Adebayo, A. O., Abiola, W. S., Ojo, O. J., Ndidi, M. P., Ayokunle, J. C., Zacchaeus, J., Osiano, I. B. (2024). Evaluating the impact of generative AI tools on learning outcomes of computer science students in tertiary institutions in New Bussa Metropolis, Niger State, Nigeria. *Dutse Journal of Pure and Applied Sciences*, *10*(4b), 1–13.

Chan, C. K. Y., & Hu, W. (2023). Students' voices on generative AI: Perceptions, benefits, and challenges in higher education. *International Journal of Educational Technology in Higher Education*, *20*(1), 43. https://doi.org/10.1186/s41239-023-00411-8

Cheah, C. S. (2020). Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. *Contemporary Educational Technology*, *12*(2), Article ep272.

Clarke, C. J. S. F., & Konak, A. (2025). The impact of AI use in programming courses on critical thinking skills. *Journal of Cybersecurity Education, Research and Practice*, *2025*(1), 5.

Fan, G., Liu, D., Zhang, R., & Pan, L. (2025). The impact of AI-assisted pair programming on student motivation, programming anxiety, collaborative learning, and programming performance: A comparative study with traditional pair programming and individual approaches. *International Journal of STEM Education*, *12*(1), 16. https://doi.org/10.1186/s40594-025-00537-3

Fereday, J., & Muir-Cochrane, E. (2006). Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development. *International Journal of Qualitative Methods*, *5*(1), 80–92.

Ghimire, A., & Edwards, J. (2024). Coding with AI: How are tools like ChatGPT being used by students in foundational programming courses. Olney, A.M., Chounta, IA., Liu, Z., Santos, O.C., Bittencourt, I.I. (eds). In *International Conference on Artificial Intelligence in Education* (pp. 259–267). Springer Nature Switzerland.

Groothuijsen, S., van den Beemt, A., Remmers, J. C., & van Meeuwen, L. W. (2024). AI chatbots in programming education: Students' use in a scientific computing course and consequences for learning. *Computers and Education: Artificial Intelligence*, *7*, Article 100290.

Güner, H., & Er, E. (2025). AI in the classroom: Exploring students' interaction with ChatGPT in programming learning. *Education and Information Technologies*, 1–27. https://doi.org/10.1007/s10639-025-13337-7

Jaboob, M., Hazaimeh, M., & Al-Ansi, A. M. (2025). Integration of generative AI techniques and applications in student behavior and cognitive achievement in Arab higher education. *International Journal of Human–Computer Interaction*, *41*(1), 353–366.

Keuning, H., Alpizar-Chacon, I., Lykourentzou, I., Beehler, L., Köppe, C., de Jong, I., & Sosnovsky, S. (2024). Students' perceptions and use of generative AI tools for programming across different computing courses. In *Proceedings of the 24th Koli Calling International Conference on Computing Education Research* (pp. 1–12). arXiv, Article arXiv:2410.06865.

Molina, I. V., Montalvo, A., Ochoa, B., Denny, P., & Porter, L. (2024). Leveraging LLM tutoring systems for non-native English speakers in introductory CS courses. *arXiv*, Article arXiv:2411.02725.

Murray, J. (2013). Likert data: What to use, parametric or non-parametric? *International Journal of Business and Social Science*, *4*(11).

Pan, Z., Xie, Z., Liu, T., & Xia, T. (2024). Exploring the key factors influencing college students' willingness to use AI coding assistant tools: An expanded technology acceptance model. *Systems*, *12*(5), 176. https://doi.org/10.3390/systems12050176

Philbin, C. A. (2023). Exploring the potential of artificial intelligence program generators in computer programming education for students. *ACM Inroads*, *14*(3), 30–38.

Prather, J., Reeves, B. N., Denny, P., Leinonen, J., MacNeil, S., Luxton-Reilly, A., Orvalho, J., Alipour, A., Alfageeh, A., Amarouche, T., Kimmel, B., Wright, J., Blake, M., Barbre, G. (2025). Breaking the programming language barrier: Multilingual prompting to empower non-native English learners. In *Proceedings of the 27th*

Australasian Computing Education Conference (pp. 74–84). arXiv, Article arXiv:2412.12800.

Silva, C. A. G. D., Ramos, F. N., De Moraes, R. V., & Santos, E. L. D. (2024). ChatGPT: Challenges and benefits in software programming for higher education. *Sustainability*, *16*(3), 1245.

Sun, D., Boudouaia, A., Zhu, C., & Li, Y. (2024). Would ChatGPT-facilitated programming mode impact college students' programming behaviors, performances, and perceptions? An empirical study. *International Journal of Educational Technology in Higher Education*, *21*(1), 14. https://doi.org/10.1186/s41239-024-00446-5

Topalli, D., & Cagiltay, N. E. (2018). Improving programming skills in engineering education through problem-based game projects with Scratch. *Computers & Education*, *120*, 64–74.

Tosi, D. (2024). Studying the quality of source code generated by different AI generative engines: An empirical evaluation. *Future Internet*, *16*(6), 188.

Vieriu, A. M., & Petrea, G. (2025). The impact of artificial intelligence (AI) on students' academic development. *Education Sciences*, *15*(3), 343.

Yilmaz, R., & Yilmaz, F. G. K. (2023). The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. *Computers and Education: Artificial Intelligence*, *4*, Article 100147.

Zijlstra, W. P., Van Der Ark, L. A., & Sijtsma, K. (2007). Outlier detection in test and questionnaire data. *Multivariate Behavioral Research*, *42*(3), 531–555.

Zviel-Girshin, R. (2024). The good and bad of AI tools in novice programming education. *Education Sciences*, *14*(10), 1089.