Journal of Information and Organizational Sciences
Volume 50, Number 1 (2026)
Journal homepage: jios.foi.hr

JIOS

DOI: 10.31341/jios.50.1.5
Open Access

004.8:004.42:378(497.5)
Original Scientific Paper

AI Tools in Programming Education: Student Perspectives and Usage Trends

Matija Novak[1*] and Marija Maček[1]

[1]Faculty of Organization and Informatics, University of Zagreb, Varaždin, Croatia
[*]Correspondence: matija.novak@foi.unizg.hr

PAPER INFO

Paper history:
Received 24 July 2025
Accepted 08 February 2026

Citation:
Novak, M. & Maček, M. (2026). AI Tools in Programming Education: Student Perspectives and Usage Trends. In Journal of Information and Organizational Sciences, vol. 50, no. 1, pp. 83-104

ABSTRACT

This research investigates the expanding use of generative AI (Mahmoud et al., 2025) tools among computer programming students. Building on a 2023 study, a 2024 survey of 182 undergraduate and graduate students explored usage patterns, perceived effectiveness, and attitudes towards their academic use. Thematic analysis revealed a significant increase in AI adoption, with students frequently using tools for code generation, debugging, and explanation. While students value the speed and efficiency AI offers, they express significant concerns about output accuracy, over-reliance, and the impact on developing independent reasoning skills. Notably, attitudes shifted from favouring full allowance of AI tools to partial allowance for programming assignments, with distinct usage patterns observed across different study levels. These findings underscore the urgent need for educational institutions to establish clear guidelines. A balanced approach is required to integrate AI tools constructively while mitigating risks to academic integrity and deep learning.

Keywords: Artificial Intelligence, Student perspectives, Computer Programming

1. Introduction

Computer programming often presents a challenge for students (Bain & Barnes, 2014; Guzdial, 2010). Research indicates that many students rely on online resources to solve programming tasks, raising ethical concerns, especially in the context of academic integrity (Cosma & Joy, 2008; Novak, 2020). The ubiquitous availability of artificial intelligence (AI) tools, such as ChatGPT, has further amplified these concerns (Denecke et al., 2023). This trend reflects the expanding use of generative AI (Mahmoud et al., 2025), which has been met with a notably polarized public response characterized by a tension between technological excitement and concern over its impact on human skills. Unlike classic search engines, AI tools can generate complete solutions for programming tasks, without the need for code integration or major modification. This capability of AI tools calls into question the effectiveness of traditional plagiarism detection methods (Novak et al., 2019), as recent studies suggest that distinguishing AI-generated code remains challenging even for specialised tools (Ellis et al., 2024). While plagiarism detection is important, prevention is key. Understanding student perspectives on the use of AI tools is important for the prevention of academic dishonesty.

This study is situated within computing education (CE) research on academic integrity in programming assessment, where generative AI complicates both the detectability of unauthorised assistance and the interpretability of students' code as evidence of individual learning. Prior work has documented challenges with source-code plagiarism in programming coursework and the limits of detection-focused responses, motivating a shift towards more authentic assessment designs that foreground process evidence and demonstration of understanding rather than relying solely on end-product similarity. (Cosma & Joy, 2008; Novak et al., 2019; Ellis et al., 2024; Prather et al., 2023; Smolansky et al., 2023)

Our contribution is threefold. First, using a parallel mixed-methods survey design, we provide a 2024 primary account of how students report using AI tools in programming courses, distinguishing common functional uses (e.g., debugging, code generation, and explanation) and the perceived trade-offs between efficiency and learning. Second, we report patterns by study level within the 2024 sample, showing that adoption for programming assignments varies across programmes. Third, by repeating the survey across two consecutive years, we provide an exploratory year-on-year comparison that indicates a statistically detectable (though modest) shift from “fully allow” towards “partially allow” positions on AI use in programming assignments. We interpret this shift cautiously given non-equivalent study groups and possible repeated respondents, but we use it to motivate concrete, course-level implications for assessment and AI-use policy.

The main goal of this research is to explore students' perspectives on the use of AI tools in solving programming tasks. The paper examines how and to what extent students use AI tools, and analyses their attitudes towards the permissibility of such use. Following this introduction, the paper proceeds with a review of the relevant literature on the use of AI tools in education. The subsequent sections describe the related work, research methods, present the results, and discuss their implications, including suggestions for future research considering faculty perspectives (Mah & Groß, 2024). The paper concludes with a summary of the key findings and answers to research questions.

2. Related Work

Research on AI in programming education encompasses studies on detection, student perspectives, and reintegration strategies (Denecke et al., 2023; Rajendran & Ramasamy, 2024). (Ellis et al., 2024) investigated the detectability of AI-generated code, finding it challenging even with AI detection tools and recommending that instructors focus on student awareness and assignment design. Surveys (Saari et al., 2024; Farhi et al., 2023) have explored student usage of AI tools, revealing substantial use for learning, searching, coding, and debugging, alongside diverse reasons for abstaining. These studies also highlight student perceptions of AI's potential and concerns regarding academic integrity (Farhi et al., 2023). (Smolansky et al., 2023) examined perspectives from both students and teachers, highlighting the need for assessment reform that foregrounds process, higher-order thinking, and authentic applications rather than relying on detectability alone. (Yilmaz & Yilmaz, 2023) explored the use of ChatGPT for programming assignments, noting benefits such as rapid feedback and debugging assistance, as well as drawbacks including potential dependency and inaccuracies. They advocate for AI integration within programming courses, coupled with instruction on verification, critical thinking, and ethical considerations; related work in introductory programming similarly frames ChatGPT as a teaching assistant whose value depends on guided use (Yilmaz & Yilmaz, 2023; Ahmed et al., 2024). Similarly, in (Ali et al., 2024), it is highlighted how AI can contribute to personalized learning by enhancing student engagement and improving educational outcomes.

In computing education (CE), the integrity question is increasingly framed as one of valid evidence of learning rather than detection alone: a submitted programme may be syntactically correct and “original” in similarity terms yet still provide weak evidence of individual reasoning when produced with substantial external assistance. This perspective builds on established work on source-code plagiarism and similarity detection in programming coursework (Cosma & Joy, 2008; Novak, 2020; Novak et al., 2019) and is sharpened by the limited reliability of AI-code detectability as a control strategy (Ellis et al., 2024). Consequently, computing education scholarship and practice increasingly emphasise assessment designs that make students' process and understanding visible, such as traceable development checkpoints via version-controlled repositories and brief oral or walkthrough-style evaluations that require students to explain, test, and adapt their submitted code, aligning closely with students' own concerns about accuracy, over-reliance, and weakened independent reasoning (Prather et al., 2023; Smolansky et al., 2023).

Publications examining students' perspectives on digital tools in programming courses and, more broadly, AI-enabled tools in higher education, report predominantly positive attitudes towards their adoption and perceived usefulness (Asgari et al., 2024; Aydemir & Seferoğlu, 2024; Balogh, 2024). Students recognise the potential of AI to enhance learning experiences, foster increased motivation, and ultimately improve learning outcomes (Aydemir & Seferoğlu, 2024). They express appreciation for AI-driven tools such as conversational agents (chatbots), adaptive learning platforms, and automated assessment methodologies (Aydemir & Seferoğlu, 2024; Balogh, 2024). However, persistent concerns remain regarding ethical implications for learning and assessment, including authorship and academic integrity (Ou et al., 2024), as well as data security and privacy (Balogh, 2024; Idroes et al., 2023). A significant minority of students, representing between 10.17% and 35.42% of those surveyed, express anxieties concerning the potential for

AI to negatively impact the long-term sustainability of higher education institutions (Okulich-Kazarin et al., 2024). Despite these reservations, students generally exhibit confidence in their perceived ability to learn effectively with and utilise AI-based tools (Balogh, 2024).

While this research centres on student viewpoints, a comprehensive understanding requires integrating faculty and institutional perspectives. Recent studies confirm that educators are actively grappling with how to adapt their pedagogy and assessments in response to GenAI. Educators internationally share concerns about academic integrity, the risk of student over-reliance, and the need for clear institutional policies (Khlaif et al., 2024). Furthermore, phenomenological studies reveal the lived experiences of educators integrating these tools, including those in challenging contexts like Palestinian higher education (Hamamra et al., 2025a). These faculty perspectives provide crucial context for the students' shift towards partial allowance observed in our data, suggesting a growing consensus among all stakeholders on the need for a balanced, critically-aware approach to AI integration. In response to these findings, researchers advocate for a nuanced and balanced approach to AI integration within educational contexts, one that prioritises not only technological innovation and advancement but also careful consideration of ethical imperatives and the implementation of human-centred practices in AI development and deployment (Balogh, 2024; Idroes et al., 2023).

Recent educator-focused studies further contextualise students' ambivalence about full versus partial allowance of GenAI. Qualitative accounts describe tensions between perceived efficiency gains and ethical or institutional uncertainty in adoption, including evidence from Palestinian higher education showing how local context shapes both use and restrictions (Hamamra et al., 2025a; Hamamra et al., 2025b), while work on educator technostress underscores the need for clear guidance that supports both learning and wellbeing (Khlaif et al., 2025). Complementary quantitative modelling using the extended Unified Theory of Acceptance and Use of Technology (UTAUT) identifies acceptance drivers such as performance expectancy, effort expectancy, social influence, and facilitating conditions (Ayyoub et al., 2025); these same constructs offer a useful interpretive frame for students' reported motivations and hesitations in our survey, although we did not measure UTAUT variables directly.

Integrating this model would allow for a deeper interpretation of why, for instance, graduate students (GL1) in our study show higher adoption rates for programming tasks (97.62%) compared to first-year undergraduates (76.48%). This difference could be attributed to higher performance expectancy (i.e., belief that AI will improve job performance) or greater exposure to facilitating conditions within their advanced coursework.

Building on prior student surveys in programming contexts (Farhi et al., 2023; Saari et al., 2024; Smolansky et al., 2023), this study contributes a 2024 primary mixed-methods snapshot that distinguishes common usage functions (e.g., generation, debugging, explanation) and examines patterns by study level within one institutional context. An exploratory consecutive-year comparison is then used to contextualise how student preferences may be shifting towards partial allowance and what this implies for programming-course policy and assessment design.

3. Methods

In a previous study (Novak & Maček, 2024), a survey was administered to students in Croatia across five courses in 2023. A notable limitation of the 2023 study was the uneven distribution of respondents between graduate students at their first (8 responses) and second year (7 responses) of study (GL2023) and undergraduate students (74 responses) at their third year of study (UL2023). Additionally, the majority of participants were enrolled in a single course, which constrained the ability to draw meaningful conclusions regarding differences between student groups. Despite these limitations, the study yielded relevant and insightful findings.

To build on the previous study, the survey was repeated in 2024 across three courses: one at the first-year undergraduate level (UL1), one at the second-year undergraduate level (UL2), and one at the first-year graduate level (GL1). A total of 187 responses were collected, of which 5 were deemed invalid, resulting in 182 valid responses (42 from GL1, 106 from UL2, and 34 from UL1). Overview of the study workflow is presented in Figure 1.

In 2024, 94 students were enrolled in the first-year undergraduate course (UL1), 195 in the second-year course (UL2), and 77 in the first-year graduate course (GL1), yielding response rates of 36.17% (34/94) for UL1, 56.36% (106/195) for UL2, and 54.55% (42/77) for GL1. For the 2023 survey, 85 students were enrolled in the third-year undergraduate course, yielding a response rate of 87.06% (74/85). For the 2023 graduate level courses this information is not available but the approximate number is around 100, which would make a response rate of ~14.00% (14/100).

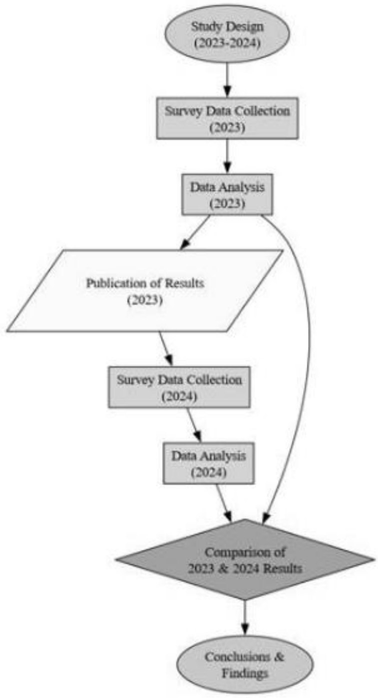

Figure 1. Study flowchart

The 2023 and 2024 surveys were administered to partially overlapping study programmes at the same institution. In 2023, the questionnaire was completed by students in third-year undergraduate and graduate level programming courses, whereas in 2024 it was administered in first-year and second-year undergraduate and first-year graduate programming courses. The survey was delivered online in the Croatian language using Google Forms, and links were shared with enrolled students through the institutional learning management system. Participation was voluntary and anonymous, and no identifiers were collected beyond self-reported level and programme of study, year of study, and current grade average. Because the 2023 and 2024 samples were drawn from different groups of students and some individuals may have participated in both surveys, the 2024 data are treated as the primary analytic sample. The 2023 results are used only for cautious, descriptive comparison rather than as a basis for formal longitudinal inference.

The questionnaire comprised background items (faculty, level and programme of study, year of study, and current grade average) and questions about students' use of AI tools for non-programming and programming assignments, the specific ways in which they used these tools, and their views on allowing or restricting AI in programming coursework. A final set of open-ended questions invited students to elaborate on their overall impressions of AI tools (advantages and disadvantages in the study context) and to add any further comments. An English translation of the full questionnaire, including response options, is provided in Appendix A. In both 2023 and 2024 the same questionnaire was used.

As already mentioned we did not collect detailed demographic information such as age or gender, nor did we include separate scales for prior programming or AI experience. This choice was made to keep the questionnaire concise and to preserve perceived anonymity, but it limits our ability to analyse differences by demographic subgroup or by formally measured prior experience.

To minimise the influence on students towards particular views, the questionnaire combined closed-ended items (e.g., yes/no questions, single-choice questions on AI allowance) with open-ended questions asking students to describe how they used AI tools and to explain their reasoning. Two binary items provide key anchors for subsequent analyses. "Used AI for assignments" refers to responses to a yes/no question asking whether students had used AI tools to solve non-programming coursework such as essays, seminar papers and other written tasks. "Used AI for programming assignments" refers to responses to a yes/no question asking whether they had used AI tools to solve programming assignments in the participating courses.

The questionnaire was purpose-designed for this study, drawing on the authors' experience of teaching programming and supervising student coursework. It was not subjected to formal psychometric validation, and we did not calculate reliability coefficients for multi-item scales. Consequently, the results should be interpreted as descriptive and context-specific, rather than as providing evidence for a validated measurement instrument.

The primary objective of the survey was to establish whether and how students used AI tools for programming assignments. A secondary objective was to understand students' perspectives on the advantages and disadvantages of such tools in the wider study context. These aims are reflected in the following research questions:

RQ1: What are students' perspectives on the use of AI tools in higher education for programming assignments?

RQ1.1: To what extent do students use AI tools for solving programming assignments?

RQ1.2: In what ways do students utilise AI tools for programming tasks, and how do they perceive their effectiveness?

RQ1.3: What are students' attitudes towards allowing or restricting AI tools in programming education?

RQ2: How do students' opinions on AI tools differ across different years of study within the same study population?

RQ2.1: Are there significant differences in AI tool adoption and perception between undergraduate and graduate students?

RQ2.2: How do students at different study levels (e.g., first-year vs. second-year undergraduate) differ in their use of AI for programming?

RQ3: In what ways have students' perspectives on AI-assisted programming changed between two consecutive years?

RQ3.1: Have students' perceptions of AI tools evolved over time, and if so, in what ways?

RQ3.2: Are there any trends in AI tool adoption between the two consecutive survey years?

1.1. Quantitative analysis

Quantitative analysis combined descriptive statistics with inferential tests. For both survey years, responses to closed-ended items were summarised using counts and percentages. The 2024 student sample was treated as the primary analytic sample because it is larger and better aligned with the current structure of the degree programmes. The 2023 responses were used only for exploratory comparison.

To examine associations between categorical variables (for example, use of AI tools versus study programme, and preferences for AI allowance versus study programme), we used Pearson's chi-square test where the expected cell counts satisfied standard assumptions. For 2×2 contingency tables with adequate expected counts we report the chi-square test with Yates's continuity correction. Where one or more expected cell counts were below 5, we instead used Fisher's exact test. For key 2×2 comparisons we report odds ratios (OR) with 95% confidence intervals (CI). For larger contingency tables we report Cramer's V with 95% confidence intervals as a measure of effect size. Statistical significance was assessed at $\alpha = 0.05$. When we conducted families of related tests on the same data (for example, pairwise comparisons between the three 2024 study programmes GL1, UL2 and UL1), we treated them as a single family and applied a Bonferroni correction to control the family-wise error rate, with adjusted threshold $\alpha\_adj = 0.05/3 \approx 0.0167$. Given the non-equivalent student groups described above, year-on-year comparisons are reported as exploratory and interpreted descriptively rather than as formal longitudinal inference.

For single-response items, percentages use all valid responses to that item as the denominator. For multiple-response items (for example, lists of AI tools used or categories of AI usage), individual student responses could contribute to several categories. These items were coded as a set of binary variables (category present/absent), and percentages in the corresponding tables represent the proportion of students who mentioned each category at least once. As a result, column percentages may sum to more than 100%.

1.2. Qualitative analysis

Qualitative analysis focused on the 2024 survey. Open-ended responses were obtained for four core questions, covering (i) how students used AI tools for non-programming assignments (e.g. essays, seminar papers), (ii) how they used AI tools for programming assignments, (iii) their reasons for allowing or restricting AI use in programming coursework, and (iv) their overall impressions of the advantages and disadvantages of AI tools in the study context. A final, "anything else" question invited additional comments. Among 182 valid respondents, 24 (13.19%) wrote something in this field; most entries were brief statements such as "nothing to add" or greetings. Only 8 students (4.40%) provided additional remarks that directly concerned AI tools or their use in studying, and these mainly reinforced themes already identified in the core open-ended questions.

We applied a descriptive thematic analysis of short open-ended responses at the semantic level. Our procedure drew on general guidance on conducting transparent and trustworthy thematic analysis (Nowell et al., 2017; Lochmiller, 2021), but was adapted to the brevity of survey answers. Two instructors who teach web programming on the participating degree programmes conducted the analysis. First, one researcher read all responses to a given question several times and generated initial inductive codes and categories that captured recurring ideas in the data. Coding was primarily data-driven in the sense that categories were derived from students' own wording, but it was also guided by the study's research questions about how AI tools are used, which uses are seen as educationally acceptable, and how students perceive impacts on learning, programming practice and assessment integrity. The same procedure was then applied to the remaining open-ended questions, with each question analysed separately so that categories remained specific to the question. Individual responses could be assigned to more than one category where students mentioned several distinct ideas.

The second researcher then reviewed the full set of coded responses and the proposed category labels for each question. Any disagreements or ambiguities were discussed until consensus was reached. Intercoder reliability statistics were not calculated, as coding was treated as a collaborative interpretive process rather than a psychometric rating task.

All coding was carried out in a shared spreadsheet, with one row per student response and one column for each category, indicating whether the category was present in that response. This provided a simple audit trail of coding decisions. We then used this spreadsheet to summarise the final categories as descriptive themes and to quantify their frequencies (allowing for multiple codes per response). These summaries are reported in the relevant tables for 2024. This quantitisation of codes allowed us to relate the qualitative themes to the closed-ended survey items on AI usage, perceived benefits and risks, and preferences for full versus partial allowance of AI tools in programming education. Given the limited length of most responses and the study's aim of informing course-level AI-use policies, the analysis was deliberately descriptive rather than deeply interpretive. Short translated examples are used to illustrate typical patterns of student reasoning.

Quantitative and qualitative strands were analysed in parallel in a parallel mixed-methods approach. Closed-ended items provided estimates of the prevalence and distribution of AI use and attitudes across study levels, while the open-ended responses were used to explain and contextualise these patterns. Quantitative results are therefore treated as the primary evidence for the strength and direction of associations (e.g., differences between study programmes or years), whereas qualitative themes are used to illuminate underlying reasons and to illustrate how students articulate perceived benefits and risks. Where an interpretation is supported solely by one strand, we indicate this in the text.

1.3. Limitations and ethics considerations

The study protocol, including the anonymous online questionnaire and data-handling procedures, was reviewed and approved by the institutional ethics committee of the authors' home institution. Participation was voluntary, and students indicated consent by proceeding to complete the survey after reading the information statement on the first page. No identifying information (such as names, e-mail addresses or student IDs) was collected, and results are reported only in aggregated form.

This design has several limitations. The study was conducted in web programming courses at a single Croatian higher education institution, whose AI uptake, institutional policies and digital readiness may differ from other contexts, which limits the generalisability of the findings. The 2023 and 2024 samples are non-equivalent student groups, and anonymity means that some students may have responded in both surveys. Year-on-year contrasts are therefore treated as descriptive comparisons rather than formal longitudinal inferences.

All data are self-reported and may not fully reflect students' actual behaviours, and the voluntary online format introduces potential self-selection bias, so reported adoption rates may overestimate AI use in the wider student population.

Finally, the questionnaire was developed specifically for this study and was not formally validated, which limits the extent to which our items can be treated as standardised measures or directly compared with instruments from other contexts. These constraints are taken into account in the interpretation of the Results and Discussion.

4. Results

This section presents the results of the survey in 2024 and the differences to the previously conducted survey in 2023. As noted in the Methods, the 2024 data are treated as the primary analytic sample, while the 2023 results are used for exploratory, descriptive comparison.

4.1. Which AI Tools are Students Using?

In the survey of 2024 a total of 187 responses were received. In response to the initial question about whether students used AI tools, only 5 indicated “no”. Since the primary focus of this research revolves around the usage of AI tools, the responses from these 5 students are excluded from the analysis. This leaves 182 responses which are used for analysis. In the 2023 survey, 84 students reported using at least one AI tool which are included in the comparative analyses.

Concerning the question which AI tools are used by students, among the 182 students in 2024 the majority mentioned using ChatGPT (181/182, 99.5%) as illustrated in Figure 2. Microsoft Copilot was reported by 100 students (54.9%), Bing Chat by 68 students (37.4%), DALL-E by 42 students (23.1%), BlackBox AI by 22 students (12.1%) and Gemini by 40 students (22.0%). Among the 84 students in 2023, ChatGPT was also the most frequently reported tool (62/84, 73.8%), followed by Microsoft Copilot (12 students, 14.3%), Bing Chat (8 students, 9.5%), DALL-E (1 student, 1.2%), BlackBox AI (3 students, 3.6%), Gemini (4 students, 4.8%), and other tools (9 students, 10.7%). Some tools that were mentioned by three or fewer students in 2024 were: Midjourney (2 responses), Ollama (1 response), Claude (1 response), ThorAI (1 response), Perplexity (1 response), Phind (1 response), Julius (1 response). In 2023 some other mentioned tools were: you.com, Gencraft, RunwayML, LLaMA, Forefront.ai, Amazon CodeWhisperer, and Codeium AI.

As this was a multiple-response item, percentages represent the proportion of students who reported using each tool at least once and therefore do not sum to 100%.

From Figure 2. it is evident that there is an increase in the reported use of AI tools among students. To assess changes in AI tool usage between the two years, we conducted separate 2×2 comparisons for each individual tool (see Appendix B). These analyses, which treat each tool as a binary independent variable (user vs. non-user), show significant increases in the adoption of ChatGPT, Microsoft Copilot, Bing Chat, DALL-E, and Gemini (all $p < 0.007$ after Bonferroni correction). This approach provides a statistically valid assessment of adoption trends while maintaining the independence of observations required for inferential testing. Given the differences in the student populations between years and the possibility that some individuals responded in both surveys, this year-on-year comparison is treated as exploratory and does not constitute formal longitudinal evidence of change.

At the level of individual tools, we conducted exploratory 2×2 comparisons of use (yes/no) between 2023 and 2024 for each tool (Appendix B). These analyses showed that students in 2024 were substantially more likely to report using ChatGPT than in 2023 (OR = 64.23, 95% CI [8.48, 486.42]), and that reported use of Microsoft Copilot (OR = 7.32, 95% CI [3.72, 14.40]), Bing Chat (OR = 5.67, 95% CI [2.58, 12.46]), DALL-E (OR = 24.90, 95% CI [3.36, 184.29]) and Gemini (OR = 5.63, 95% CI [1.94, 16.32]) also increased. After applying a Bonferroni correction for seven comparisons, these differences remained statistically significant (adjusted $\alpha \approx 0.007$). In contrast, differences for BlackBox AI (OR = 3.71, 95% CI [1.08, 12.77]) and the aggregated “Other” category (OR = 0.38, 95% CI [0.14, 1.03]) were not statistically significant after correction. Again, these per-tool comparisons should be interpreted cautiously as exploratory indicators of change rather than as formal longitudinal effects.

4.2. Using AI Tools for Solving Assignments in General

When the students were asked if they used AI tools to solve non-programming assignments in general (for example, essays, seminar papers and other written coursework), the majority answered affirmatively. The data are presented in Table 1.

In 2024, 82.97% indicated that they used AI tools for such assignments, while 17.03% reported not using them. Usage was high across all three course groups (GL1, UL2, UL1), with between roughly four out of five and nine out of ten students reporting use. In the 2023 survey, 90.48% reported using AI tools for assignments and 9.52% did not, again indicating high adoption in both the graduate (GL2023) and undergraduate (UL2023) groups.

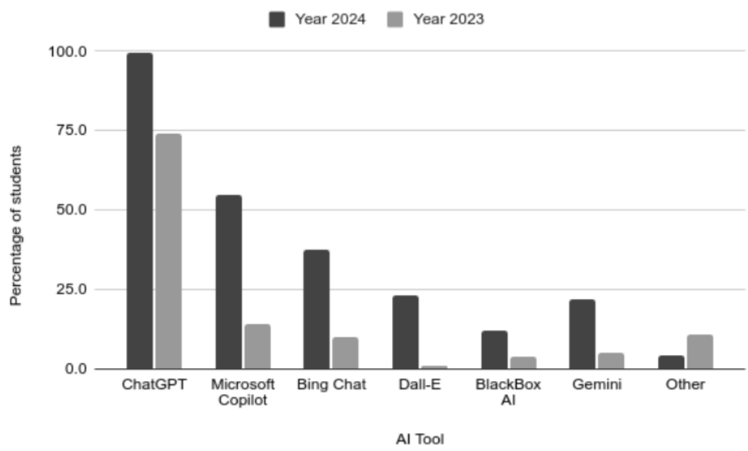

Figure 2. Percentage of students who used an AI tool

As an exploratory comparison, we examined whether overall use of AI tools for non-programming assignments differed between 2023 and 2024. A 2 × 2 chi-square test with Yates's continuity correction (year × use vs. non-use) showed no statistically significant association, $\chi^2(1) = 2.02$, $p = 0.15$. The corresponding effect size was small, Cramer's $V = 0.09$, 95% CI [0.00, 0.21]. Expressed as an odds ratio, students in 2024 had somewhat lower odds of reporting AI use than students in 2023 ($OR_{(2024 \text{ vs } 2023)} = 0.51$, 95% CI [0.22, 1.17]), but the confidence interval includes 1, indicating that this apparent decrease is statistically uncertain.

To examine whether AI tool usage for non-programming assignments differed between study programmes within the same year, we conducted a chi-square test on the 2024 data comparing GL1, UL2 and UL1. The test indicated no statistically significant difference in usage between these groups, $\chi^2(2) = 2.23$, $p = 0.33$. The associated effect size was small, Cramer's $V = 0.11$, 95% CI [0.03, 0.25], consistent with only modest variation in usage rates across the three course groups. Overall, the quantitative data suggest that the use of AI tools for non-programming assignments was widespread and relatively similar across study programmes and between the two survey years.

While the quantitative data indicate that AI tool usage is relatively stable across different student groups and years. Further qualitative aspects were explored of AI adoption to understand factors influencing students' decisions to use or avoid these tools. For the 2024 survey, open-ended responses about non-programming assignments were coded into five usage categories (Table 2). Because this was a multiple-response item, individual responses could be assigned to more than one category, and percentages represent the proportion of students in each course group whose response was coded with a given category.

		2024 survey				**2023 survey**		
Response	**Course (Participants)**	**GL1 (n = 42)**	**UL2 (n = 106)**	**UL1 (n = 34)**	**Total (n = 182)**	**GL2023 (n = 14)**	**UL2023 (n = 70)**	**Total (n = 84)**
Did not use AI tools	n	4	20	7	31	1	7	8
	%	9.52%	18.87%	20.59%	17.03%	7.14%	10.00%	9.52%
Used AI tools	n	38	86	27	151	13	63	76
	%	90.48%	81.13%	79.41%	82.97%	92.86%	90.00%	90.48%

Note: This was a single-response item. Percentages show, for each course group and year, the proportion of students who reported using or not using AI tools.
2024: UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.
2023: UL2023 = third-year undergraduate; GL2023 = first-year and second-year graduate

Table 1. Usage of AI tools for solving non-programming assignments in 2024 and 2023

Across all three course groups in 2024, the most frequently coded use of AI tools was support with text composition for seminar papers, essays, task completion and structure generation, followed by content correction, restructuring and translation, and information retrieval and verification. Idea generation and support with content analysis and comprehension were also common. Descriptively, graduate students (GL1) more often mentioned content correction and translation, whereas UL2 students more often mentioned text composition and information retrieval, which may point to subtle differences in how students at different stages of study integrate AI tools into their written coursework. The overall pattern of categories is, however, broadly similar across GL1, UL2 and UL1, and these patterns are exploratory and would need to be examined more rigorously in future work. Overall, the coded responses suggest that students at different study levels primarily draw on AI tools to help them draft, refine and make sense of written coursework, rather than to bypass it entirely.

These categories are clearly visible in students' own descriptions. For idea generation, one student wrote that the tool "helped me to generate text, that is, to generate ideas that I could integrate into my assignments". Others emphasised information retrieval and verification, using AI "to check whether information is correct". Several students described AI as a way to polish or restructure their writing, for example by "generating example text or sentence beginnings that then needed to be corrected afterwards". Students also reported relying on AI for drafting seminar papers and essays "to help me with writing seminar papers and essays, or to give me ideas about what to write", and for making sense of complex topics, "to better understand the topic, to explain to me what it is about".

Categories	Course (Participant)	GL1 (n = 42)	UL2 (n = 106)	UL1 (n = 34)	Total 2024 (n = 182)
Idea generation	n	12	14	8	34
	%	28.57%	13.21%	23.53%	18.68%
Information retrieval and verification	n	12	23	6	41
	%	28.57%	21.70%	17.65%	22.53%
Content correction and restructuring / text translation	n	16	19	8	43
	%	38.10%	17.92%	23.53%	23.63%
Text composition (seminar papers, essays etc.) / task completion / structure generation	n	10	37	9	56
	%	23.81%	34.91%	26.47%	30.77%
Content analysis and comprehension	n	11	21	5	37
	%	26.19%	19.81%	14.71%	20.33%

Note: This was a multi-response item where each participant's open-ended response could be coded into more than one category. Percentages are the proportion of students in each course group whose open-ended response was coded with the given category.
UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.

Table 2. AI tools usage for non-programming assignments in 2024

4.3. Using AI Tools for Solving Programming Assignments

When students were asked whether they used AI tools specifically for solving programming assignments, the majority answered affirmatively, although there was more variation than for non-programming assignments.

From the data (Table 3.) it is visible that there is an increase of using AI tools for programming assignments between 2023 and 2024. Also there seems to be a difference between the different levels of study programmes.

		2024 survey				2023 survey		
Response	Course (Participants)	GL1 (n = 42)	UL2 (n = 106)	UL1 (n = 34)	Total (n = 182)	GL2023 (n = 14)	UL2023 (n = 70)	Total (n = 84)
Did not use AI tools	n	1	13	8	22	6	11	17
	%	2.38%	12.26%	23.53%	12.09%	42.86%	15.71%	20.24%
Used AI tools	n	41	93	26	160	8	59	67
	%	97.62%	87.74%	76.47%	87.91%	57.14%	84.29%	79.76%

Note: This was a single-response item. Percentages show, for each course group and year, the proportion of students who reported using or not using AI tools.
2024: UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.
2023: UL2023 = third-year undergraduate; GL2023 = first-year and second-year graduate

Table 3. Usage of AI tools for solving programming assignments in 2023 and 2024

As an exploratory comparison between years, we examined whether overall use of AI tools for programming assignments differed between 2023 and 2024. A 2 × 2 chi-square test with Yates's continuity correction (year × use vs non-use) showed no statistically significant association, $\chi^2(1) = 2.43$, $p = 0.12$. The corresponding Cramer's V was small, $V = 0.10$, 95% CI [0.00, 0.22]. Expressed as an odds ratio, students in 2024 had higher odds of reporting AI use than students in 2023 (OR = 1.85, 95% CI [0.92, 3.69]), but the confidence interval includes 1. Combined with the non-equivalent student groups across years and the possibility that some individuals responded in both surveys, these results suggest that any year-on-year difference in reported AI use for programming assignments is uncertain and should be interpreted descriptively.

We then examined whether, within 2024, the use of AI tools for programming assignments differed between study programmes. A chi-square test comparing GL1, UL2 and UL1 indicated a statistically significant association between programme and AI use, $\chi^2(2) = 7.92$, $p = 0.019$, with a small-to-moderate effect size, Cramer's V = 0.21, 95% CI [0.10, 0.36]. To explore which groups differed, we conducted three pairwise 2 × 2 comparisons using Fisher's exact test (GL1 vs UL2, GL1 vs UL1, UL2 vs UL1), treating them as a single family with a Bonferroni-corrected threshold $\alpha\_adj \approx 0.05/3 \approx 0.017$. The only pair that remained statistically significant after correction was GL1 versus UL1 ($p = 0.009$), where the odds of reporting AI use were much higher for GL1 than for UL1 (OR = 12.62, 95% CI [1.49, 106.81]), consistent with the near-universal usage reported by GL1 students. For GL1 versus UL2, Fisher's exact test yielded $p = 0.11$ with an odds ratio of 5.73 (95% CI [0.73, 45.28]), and for UL2 versus UL1, $p = 0.16$ with an odds ratio of 2.20 (95% CI [0.82, 5.88]). In both of these comparisons the confidence intervals include 1, indicating that any differences in usage between these groups are small or imprecisely estimated and do not meet the Bonferroni-adjusted criterion for statistical significance.

For the 2023 data, GL2023 ($n = 14$) combines students from two graduate programmes and is therefore both small and heterogeneous. Given this and the exploratory role of the 2023 survey, we did not conduct formal within-year tests for 2023. Instead, the GL2023 and UL2023 percentages in Table 3 are reported descriptively and interpreted with caution. Notably, the 2024 UL2 usage rate (87.7%) is similar to that of UL2023 (84.3%), whereas the lower usage observed in GL2023 (57.1%) contrasts with the very high usage in GL1 (97.6%). However, because the graduate groups differ in size and composition across years, and because we lack true longitudinal tracking, these differences cannot be taken as evidence of a real shift in graduate behaviour.

Overall, the results indicate that AI tool use for programming assignments is high across study programmes, with particularly strong uptake among first-year graduate students in 2024. Descriptively, this may reflect greater readiness or opportunity among GL1 students to integrate AI support into programming coursework, for example because of their prior experience or the way AI is discussed in their classes. At the same time, first-year undergraduates (UL1) reported the lowest usage rate in 2024 (76.5%), and similar patterns of more cautious or uneven adoption among early-year students have been noted in previous work on AI in programming education (Yilmaz & Yilmaz, 2023). The absence of statistically robust differences between UL2 and UL1 within 2024, and the uncertainty around year-on-year changes, nonetheless suggests that patterns of AI adoption for programming tasks are broadly similar across neighbouring study levels, with some local variation that warrants further investigation in future work.

Table 4. shows that AI tools are widely used in programming assignments, with code generation, debugging, and code explanation being the most prominent use cases across all three course groups. Documentation writing was mentioned only by a small subset of students, and only in the GL1 group, while

“learning” (for example, using AI to understand concepts or review material) appeared across all levels. As this was a multiple-response item, individual responses could be coded into several categories, and the percentages reflect the proportion of students in each course group whose open-ended description of AI use was coded with each category. Descriptively, GL1 students more often mentioned code generation, debugging and documentation than the undergraduate groups, whereas UL1 students somewhat more frequently emphasised code explanation. We did not conduct formal statistical tests on these categories, so these differences should be interpreted cautiously as indicative patterns rather than definitive group contrasts.

Students’ comments illustrate how these categories played out in practice. For code generation, one student described using AI “to generate very specific pieces of code that require more detailed knowledge of the language syntax”. Debugging support was also prominent: “When I had certain errors at run time or during compilation, I tried to resolve them using AI tools.” Others focused on explanation and learning, for example asking AI “to write out and explain parts of the code that I did not know how to write myself”, or using it “to explain certain programming concepts, and for examples that illustrate those concepts”. Only a small subset of students mentioned documentation explicitly, describing AI as useful “for documenting the programmes I had written and for solving problems in the programme”.

Categories	**Course (Participant)**	**GL1 (n = 42)**	**UL2 (n = 106)**	**UL1 (n = 34)**	**Total 2024 (n = 182)**
Code generation	n	22	42	11	76
	%	52.38%	39.62%	32.35%	41.76%
Debugging	n	19	41	7	68
	%	45.24%	38.68%	20.59%	37.36%
Code explanation	n	10	34	12	58
	%	23.81%	32.08%	35.29%	31.87%
Documentation writing	n	5	0	0	5
	%	11.90%	0%	0%	2.75%
Learning	n	10	20	7	38
	%	23.81%	18.87%	20.59%	20.88%

Note: This was a multi-response item where each participant’s open-ended response could be coded into more than one category. Percentages are the proportion of students in each course group whose open-ended response was coded with the given category.
UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.

Table 4. AI tools usage for programming assignments in 2024

This categorisation was developed inductively from student responses to reflect their self-described activities. While this student-centred approach differs from some existing taxonomies, our themes align with primary use cases reported in prior research. For example, our “Code generation” and “Debugging” categories correspond to the “Feature implementation” and “Software quality (Fixing)” tasks noted by (Keuning et al., 2024), as well as the common uses of “writing code” and “debugging” identified by (Hou et al., 2025) and (Prather et al., 2023). Our categorisation complements these works by offering a transparent scheme grounded directly in students’ own descriptions of their AI usage in an educational context.

4.4. Students Opinion about AI Allowance

Another interesting question was students' opinions on whether AI tools should be allowed or forbidden for solving programming assignments. A comparison was made between the data from 2023 and 2024, presented in Table 5. In 2024, most students favoured partially allowing AI tools 57.1%, with a smaller proportion supporting full allowance 40.1% and very few favouring a complete ban 2.7%. In 2023, the pattern was more evenly split between partial allowance 41.7% and full allowance 57.1%, with only one student favouring a complete ban 1.2%.

To explore whether the overall distribution of opinions differed between the two survey years, we conducted a chi-square test on the three response categories (partially allow, allow, forbid) by year. The test indicated a statistically significant association, $\chi^2(2) = 6.91$, $p = 0.031$. The corresponding effect size was small, Cramer's V = 0.16, 95% CI [0.06, 0.28], indicating a modest shift in the pattern of responses rather than a large change. Descriptively, this shift reflects an increased preference for partial allowance of AI tools in 2024 and a decrease in the proportion of students favouring full allowance. Given the non-equivalent student groups in 2023 and 2024 and the possibility that some individuals responded in both surveys, these year-on-year differences should be interpreted as exploratory and descriptive.

		2024 survey				2023 survey		
Response	Course (Participant)	GL1 (n = 42)	UL1 (n = 34)	UL2 (n = 106)	Total (n = 182)	GL2023 (n = 14)	UL2023 (n = 70)	Total (n = 84)
Partially Allow	n	26	21	57	104	7	28	35
	%	61.90%	61.76%	53.77%	57.14%	50.00%	40.00%	41.67%
Allow	n	16	10	47	73	7	41	48
	%	38.10%	29.41%	44.34%	40.11	50.00%	58.57%	57.14%
Forbid	n	0	3	2	5	0	1	1
	%	0.0%	8.82%	1.89%	2.75%	0.0%	1.43%	1.19%

Note: This is a single-response item. Percentages show, for each course group and year, the proportion of students who selected "allow", "forbid", or "partially allow".
2024: UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.
2023: UL2023 = third-year undergraduate; GL2023 = first-year and second-year graduate

Table 5. Students responses regarding AI allowance for solving programming assignments in 2023 and 2024

Further, we examined whether attitudes in 2024 differed between the three study programmes. Because the "forbid" option was chosen by very few students, we compared only the "partially allow" and "allow" responses across GL1, UL1 and UL2. A chi-square test showed no statistically significant association between programme and attitude, $\chi^2(2) = 1.87$, $p = 0.39$. The corresponding Cramer's V was small, V = 0.10, 95% CI [0.03, 0.25]. Although UL2 students numerically showed the highest support for fully allowing AI tools and UL1 students were somewhat more likely to support a complete ban, these differences are small and statistically uncertain. Overall, the 2024 data suggest that support for partial allowance versus full allowance of AI tools is broadly similar across the three course groups.

Overall, these results indicate that a clear majority of students in both survey years favoured a partial allowance model for AI tools in programming assignments, in which AI can be used but not without constraints. This pattern suggests that many students recognise both the benefits of AI for supporting their work and the need for responsible, controlled use that maintains academic integrity and encourages them to engage meaningfully with programming tasks, rather than relying on AI tools as a complete substitute.

Next, we analysed the qualitative data, the results of which are summarised in Table 6. This was a multiple-response item, so individual open-ended responses could be coded into several categories. Descriptively, there is substantial overlap between the "Partially allow" and "Allow" groups: in both, many students emphasised that AI tools help them to program faster, and relatively few raised ethical concerns. At the same time, the pattern of categories suggests important differences in emphasis between these two positions.

In the "Partially allow" group, students more often framed AI as something that should support learning rather than replace it. A higher proportion of these students referred to AI as "good for learning" and stressed that "code should be understood by students" and that AI tools are acceptable for partial code generation but not for writing entire solutions. They also more frequently mentioned debugging support and the idea that AI should be forbidden for simple problems, where students are expected to develop core skills.

This cautious stance in the "Partially allow" group is well illustrated by one student's reflection: "AI is a fine option for solving assignments only if it is not used as a complete substitute for learning the material." Another argued that AI should be permitted but coupled with verification of understanding: "They should be allowed, but then there should be defences of assignments where students must show that they understand the code that has been written." Students in this group often framed AI as acceptable for partial code generation: "AI tools should be allowed in the sense that they may be used to write only a small part of the assignment, and that this is explicitly stated", and for debugging support: "AI should not write the entire code; it would also be good if the student can write code without AI for practice and creativity. However, debugging is something for which AI would be excellent."

Categories	Response (Participant)	Partially allow (n = 104)	Allow (n = 73)	Forbid (n = 5)
Good for learning	n	29	10	
	%	27.88%	13.70%	
AI tools are just help to program faster	n	27	20	
	%	25.96%	27.40%	
Code should be understood by students	n	25	10	
	%	24.04%	13.70%	
AI tools for partial code generation	n	24	6	
	%	23.08%	8.22%	
Debugging	n	14	2	
	%	13.46%	2.74%	
Forbid in simple problems	n	13	2	1
	%	12.50%	2.74%	20.00%
Used in companies, this is the future	n	12	24	
	%	11.54%	32.88%	
Same as google	n	5	7	
	%	4.81%	9.59%	
Ethics problems	n	3	2	1
	%	2.88%	2.74%	20.00%
Forbid if the goal is learning	n			3
	%			60.00%

Note: This was a multi-response item where each participant's open-ended response could be coded into more than one category.
For each response option ("allow", "forbid", "partially allow"), percentages show the proportion of students who selected that option whose open-ended response was coded with the given category.

Table 6. Reasons regarding AI allowance for solving programming assignments in 2024

By contrast, students in the "Allow" group more often justified their position by appealing to external practice and future-oriented arguments. One student stated simply that AI tools "should be allowed because this is the future and we need to learn how to use these tools", while another compared AI directly to existing online resources: "Honestly? I think it should be fully allowed, because what is the point of allowing help from StackOverflow and Google but not from AI?" This 'same as Google' framing positions AI as a natural extension of current search and forum-based help.

The "Forbid" group is small, but their responses are more tightly focused on learning and ethics. One student wrote that AI tools "should be banned for any type of assignment whose aim is to demonstrate knowledge", while another warned that "they can sometimes help, but they can be very harmful for learning and are most often used for cheating". These comments underline a stricter interpretation of academic integrity, particularly in tasks designed to evidence individual competence.

Overall, these patterns suggest that while both the "Partially allow" and "Allow" groups recognise the practical benefits of AI in programming, the partial-allowance position is more strongly tied to concerns about preserving learning opportunities and code understanding, whereas full allowance is more often justified in terms of real-world practice and the perceived inevitability of AI.

4.5. General Opinion of AI Tools

Regarding the advantages presented in Table 7., the data suggests nuances in how students at different stages perceive the benefits of AI tools. While faster problem-solving and clarification is a key advantage overall (46.70%), it has a slightly higher prevalence among graduate (GL1: 50%) and second-year undergraduate (UL2: 48.11%) students compared to first-years (UL1: 38.24%). This might reflect the increasing complexity of tasks they face, making speed a more critical factor. Conversely, first-year undergraduates (UL1) stand out for their strong view of AI as more efficient than traditional search methods (52.94%), which is higher compared to UL2 (16.98%) and GL1 (33.33%). This could imply that students newer to higher education find the leap from conventional search tools to AI more impactful, perhaps having less established research habits. The minimal recognition of creativity fostering (1.10% overall) indicates students across levels primarily view these tools through a lens of productivity and problem-solving rather than generative exploration.

Categories	Course (Participant)	GL1 (n = 42)	UL2 (n = 106)	UL1 (n = 34)	Total 2024 (n = 182)
Faster problem-solving and solution clarification	n	21	51	13	85
	%	50.00%	48.11%	38.24%	46.70%
More efficient than traditional search engines or online forums	n	14	18	18	50
	%	33.33%	16.98%	52.94%	27.47%
Creativity fostering	n	1	1	0	2
	%	2.38%	0.94%	0%	1.10%

Note: This was a multi-response item where each participant's open-ended response could be coded into more than one category. Percentages are the proportion of students in each course group whose open-ended response was coded with the given category.
UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.

Table 7. Advantages of AI tools in the study context in 2024

Analysis of disadvantages in Table 8. reveal varying levels of concern among the student groups. The risk of incorrect results is a major concern across all study levels (36.26% overall), with graduate students (GL1: 42.86%) expressing this slightly more often, possibly due to the higher stakes associated with accuracy in advanced tasks. More pronounced differences emerge regarding the impact on learning habits. Graduate students (GL1) show higher concern about developing an over-reliance on AI (23.8%) and a potential decline in independent reasoning (19.0%) compared to both first-year (UL1: 14.71% for both concerns) and especially second-year undergraduates (UL2: 12.26% and 11.32% respectively) . This heightened concern among GL1 students might stem from a greater awareness of the necessity for deep, independent critical thinking at the graduate level and a recognition of how easily AI could potentially undermine its development. Second-year students (UL2) appear least concerned about these specific learning-related drawbacks. Ethical considerations were a minor theme overall (7.14%), with first-year students (UL1: 8.82%) mentioning them slightly more, perhaps reflecting an initial engagement with the academic integrity implications of these new tools.

The perceived advantages and disadvantages of AI tools also come through clearly in students' own words. For advantages, one student noted that AI is "useful if it is used for the right purposes… the advantages are faster problem-solving and more interesting solutions", while another described it as "a good replacement for Google and StackOverflow" that "makes work easier and faster" and "speeds up searching". A smaller group linked AI to creativity, commenting that it "increases students' creativity and efficiency, but there is also the possibility of plagiarism, which is negative". On the risk side, several students highlighted over-reliance and reduced independent thinking: "The advantage is saving time. The disadvantage is that relying too much on AI can sometimes be bad," and simply, "the disadvantage is a lack of one's own thinking." Concerns about incorrect outputs were equally direct: AI was seen as "a quick and simple way to get an answer to a given question, but sometimes those answers are incorrect." Together, these comments echo the survey categories by portraying AI as simultaneously time-saving and potentially misleading, and by tying ethical concerns directly to plagiarism and misuse.

Categories	Course (Participant)	GL1 (n = 42)	UL2 (n = 106)	UL1 (n = 34)	Total 2024 (n = 182)
Over-reliance on AI tools	n	10	13	5	28
	%	23.81%	12.26%	14.71%	15.38%
Lack of independent reasoning	n	8	12	5	25
	%	19.05%	11.32%	14.71%	13.74%

Incorrect results	n	18	35	13	66
	%	42.86%	33.02%	38.24%	36.26%
Ethical considerations (misuse, plagiarism, human replacement)	n	3	7	3	13
	%	7.14%	6.60%	8.82%	7.14%

Note: This was a multi-response item where each participant's open-ended response could be coded into more than one category. Percentages are the proportion of students in each course group whose open-ended response was coded with the given category.
UL1 = first-year undergraduate; UL2 = second-year undergraduate; GL1 = first-year graduate.

Table 8. Disadvantages of AI Tools in the study context in 2024

No formal significance tests were conducted for these multi-response categories, so the patterns reported here should be interpreted descriptively. Nonetheless, the combination of perceived advantages (speed, efficiency) and disadvantages (risk of inaccuracy, over-reliance, weaker independent reasoning, and ethical concerns) aligns with the broader shift towards partial allowance reported in Section 4.4. Students appear to value AI tools as supportive resources in programming and study more generally, while simultaneously recognising that unregulated or opaque use could compromise both learning outcomes and academic integrity.

5. Discussion

The use of AI-powered tools among students appears to have increased between 2023 and 2024, with a clear descriptive rise in the proportion of students reporting use of mainstream tools. In particular, reported use of ChatGPT rose from around three-quarters of respondents in 2023 to almost all respondents in 2024, with similar descriptive increases for Microsoft Copilot, Bing Chat and DALL-E. However, because the 2023 and 2024 samples were drawn from non-equivalent student groups and some individuals may have responded in both surveys, these year-on-year patterns should be interpreted as exploratory trends rather than as formal longitudinal evidence of change.

These findings highlight the growing role of AI in student education, with a clear preference for tools that integrate seamlessly into existing workflows and provide a high level of automation. Given the rapid development of AI models and their increasing adaptation to academic and professional needs, this trend is expected to continue. This consolidation around a small set of widely available tools also has practical implications for programming education, because course-level policies and assessment designs can realistically target a finite set of tools rather than an open-ended ecosystem.

Regarding usage of tools for solving assignments, the findings suggest that AI tools remain widely used for academic assignments. Although the proportion of students reporting AI use for non-programming assignments was somewhat lower in 2024 than in 2023, the year-on-year difference was not statistically significant and the associated effect size (Cramer's V) was small, with a confidence interval that includes zero. Similarly, within 2024 we found no statistically significant differences between programmes and only small effect sizes. Overall, these findings suggest that AI tool usage for non-programming assignments is broadly stable across study programmes and years, although the non-equivalent study groups limit the strength of year-on-year inferences.

For programming assignments, the 2024 data highlight meaningful variation in usage between study levels. First-year graduate students (GL1) in 2024 demonstrated the highest adoption rate, while first-year undergraduate students (UL1) had the lowest. These patterns are consistent with the idea that adoption of AI tools for programming varies by study level, with the most pronounced contrast between first-year graduate students (GL1) and first-year undergraduates (UL1).

The higher adoption rate among graduate students may reflect their greater familiarity with advanced tools, a stronger focus on efficiency in completing assignments, or better awareness of benefits of AI tools. On the other hand, the lower adoption rate among undergraduate students could be attributed to a lack of exposure, insufficient training, or a perception that such tools are not necessary for their level of study, similar to (Yilmaz & Yilmaz, 2023). The odds ratio for reporting AI use was much higher for GL1 than for UL1, although the confidence interval was wide due to the very small number of GL1 students who reported non-use, so this difference should be interpreted cautiously.

These results underscore the importance of targeted interventions that support transparent and critical use of AI tools, particularly among undergraduates. For example, workshops, tutorials or explicit integration of AI into the curriculum can help students understand both the benefits and limitations of these tools. Additionally, further research is needed to explore the specific barriers faced by undergraduate students and to evaluate the effectiveness of potential interventions. Rather than simply encouraging more use among undergraduates, targeted interventions should help students at all levels to use AI tools transparently and critically. In programming education this points to assessment designs that foreground code understanding and provenance, for example, combining supervised labs with version-controlled repositories and brief code walkthroughs that require students to explain, adapt and test AI-assisted solutions.

Similarly, based on the question about allowing AI tools for programming assignments, there has been a shift in students' critical perspectives on AI tools. Consistent with the statistically detectable but modest shift from full towards partial allowance between 2023 and 2024, the qualitative analysis shows that the ‘Allow’ and ‘Partially allow’ groups articulate AI’s benefits and risks in systematically different ways. Specifically, the percentage of students supporting the complete allowance of AI tools has decreased, while those advocating for partial allowance have increased between 2023 and 2024. While the majority of students across all courses in 2024 preferred a balanced approach, the lack of statistically significant differences between programmes suggests that viewpoints regarding AI allowance are broadly similar across neighbouring academic stages and study programmes.

However, the observed shift toward “partial allowance” should be interpreted with caution. While it may suggest a maturation of student perspectives over time, it could also be influenced by the different composition of the 2024 student population. Specifically, the inclusion of first-year undergraduate students (UL1), who reported more cautious adoption rates (76.5%), may introduce a “freshman mindset” characterized by stricter adherence to perceived academic rules compared to the more experienced third-year students surveyed in 2023. These student concerns regarding efficiency versus over-reliance mirror the broader societal polarization identified by Mahmoud et al. (2025), where public sentiment oscillates between excitement over AI's potential and fear of its long-term impact on human skills.

Nonetheless, from the qualitative data one can see that "Allow" and "Partially Allow" groups reflect distinct attitudes towards the use of AI tools in programming, but also show some overlap in terms of where they find value in these tools. The "Allow" group, with 73 students, represents a more permissive stance, where AI tools are fully embraced, especially for their ability to speed up programming processes (27.39%). This group seems to believe that AI tools can be fully integrated into the programming workflow, particularly in practical, real-world scenarios, such as in companies, where they are seen as a future technology (32.87%). However, the "Partially Allow" group, while similarly open to AI's utility, is more cautious and reserved, with 104 students supporting a limited usage model. This group tends to appreciate AI tools as helpful for certain tasks like partial code generation (23.07%) and debugging (13.46%), but is less inclined to fully trust or rely on them in all contexts, especially when the goal is learning or understanding code.

Where these two groups converge is in their shared recognition of AI tools' potential to improve programming speed and assist in tasks like debugging and partial code generation. Both groups see value in AI as a tool to help students become more efficient, yet they differ significantly in how much they are willing to allow AI to take over the programming process. While the "Allow" group supports AI as a future-forward, all-encompassing tool, the "Partially Allow" group emphasizes the importance of maintaining a balance between using AI as an assistant and ensuring students learn the fundamentals of programming. Our qualitative data clarifies that “partial allowance” is specifically envisioned as using AI for debugging support (13.46%) and partial code generation (23.08%), while strictly forbidding its use for solving entire assignments or simple foundational problems where core skill acquisition is the primary goal. This divergence highlights a central tension for programming educators: how to capitalise on AI’s productivity benefits while preserving opportunities for students to practise algorithmic reasoning, debugging and code comprehension in ways that support academic integrity.

For instructional practice, these findings imply a necessary shift from outcome-based to process-based evaluation. Educators should explicitly integrate AI literacy into the curriculum, moving beyond mere permission or prohibition. Specifically, we suggest “collaborative debugging” sessions where instructors and students critique AI-generated code together. This approach not only addresses the accuracy concerns raised by 36.26% of our participants but also reinforces the “student-as-editor” role, which is essential for maintaining independent reasoning while utilising generative tools.

Operationalising a partial-allowance approach in programming assessment. Our results suggest that many students prefer partial allowance because they value GenAI’s efficiency while recognising risks related to inaccurate outputs, over-reliance, and weakened independent reasoning. Rather than treating “balance” as a general principle, we translate it into a simple assessment framework that links (i) accuracy risks to explicit verification requirements (e.g., tests and justification of outputs), (ii) over-reliance to traceable process evidence and disclosure (e.g., staged development checkpoints and a brief statement of when AI was used and what was accepted or rejected), and (iii) weakened reasoning to short walkthrough-style checks in which students must explain, test, and adapt their submitted code. This framing aligns with computing-education calls to refocus assessment on process evidence and demonstrated understanding, given that AI-generated programming work may be difficult to identify reliably and can weaken the evidential value of take-home artefacts (Prather et al., 2023; Smolansky et al., 2023; Ellis et al., 2024).

While this study provides valuable insights from a Croatian context, its focus on web programming students necessarily limits the direct generalisability of the findings. However, the core themes, such as the student-led shift towards partial AI allowance and concerns about over-reliance, resonate with broader trends observed in international studies. Large-scale surveys have documented prevailing attitudes and usage patterns across 20 countries (Prather et al., 2023), while specific studies in North America have tracked a dramatic increase in GenAI adoption, with the percentage of students who reported never using ChatGPT plummeting from 34.1% in 2023 to just 6.3% in 2024. (Hou et al., 2025). Similar trends of evolving usage have also been documented at large European universities (Keuning et al., 2024).

Future research should continue to explore these dynamics in diverse settings, including the Global South, where cultural and institutional factors may uniquely shape AI adoption, as demonstrated in studies from Palestinian universities that explore both student usage patterns and educator experiences (Hamamra et al., 2025a; Hamamra et al., 2025b). In addition, it would be beneficial to add more courses and different study programmes, and investigate things such as the role of course content, prior exposure to AI tools, or cultural factors within the programmes, to better understand how to promote responsible and effective use of AI tools in education. Pursuing these research directions can lead to a more comprehensive understanding of AI tools' evolving role in computer programming education, shaping more effective integration strategies.

5.1. Summary of findings

Regarding students' overall perspectives on using AI tools in higher education for programming assignments (RQ1), the findings revealed several key aspects.

RQ1.1: To what extent do students use AI tools for solving programming assignments? The study found high usage, with 87.92% of students reporting using AI for programming tasks in 2024, representing a slight (though not statistically significant and the effect size was small) increase from 79.76% in 2023. Almost all students surveyed had experimented with AI tools.

RQ1.2: In what ways do students utilise AI tools for programming tasks, and how do they perceive their effectiveness? Students primarily use AI for code generation (41.76%), debugging (37.36%), and code explanation (31.87%). They perceive these tools as effective for faster problem-solving and increasing efficiency compared to traditional methods, although concerns about accuracy are prevalent.

RQ1.3: What are students' attitudes toward allowing or restricting AI tools in programming education? There was a statistically detectable but modest shift from 2023 to 2024, with the majority preference moving from fully allowing AI (57.14% in 2023) to partially allowing it (57.14% in 2024). This indicates a developing preference for a balanced approach that leverages AI benefits while managing risks.

Overall, students generally hold a positive yet increasingly cautious perspective towards AI tools for programming, recognising their utility while also advocating for controlled usage and acknowledging potential drawbacks like inaccuracy and over-reliance (RQ1).

The study also examined how students' opinions on AI tools differ across different years of study (RQ2).

RQ2.1: Are there significant differences in AI tool adoption and perception between undergraduate and graduate students? Yes, statistically detectable differences were observed in 2024, primarily between first-year graduate and first-year undergraduate students, with a small-to-moderate effect size. First-year graduate students (GL1) showed higher adoption for programming tasks (97.62%) compared to first-year undergraduates (UL1: 76.48%). GL1 students also expressed greater concern regarding over-reliance and the potential negative impact on independent reasoning.

RQ2.2: How do students at different study levels (e.g., first-year vs. second-year undergraduate) differ in their use of AI for programming? Second-year undergraduates (UL2) used AI for programming more frequently (87.74%) than first-years (UL1: 76.48%), though this difference was not statistically significant. UL2 students also appeared least concerned about over-reliance and lack of independent reasoning compared to other groups.

Overall, students' opinions and usage patterns regarding AI tools do appear to differ by level of study, with the most pronounced differences between first-year graduate students and first-year undergraduate students in terms of adoption rates and concerns about learning impacts.

The research also investigated how students' perspectives on AI-assisted programming changed between two consecutive years (RQ3).

RQ3.1: Have students' perceptions of AI tools evolved over time, and if so, in what ways? Yes, perceptions appear to have evolved. The most notable change was the shift in attitude regarding allowance for programming assignments, with a move towards favouring partial allowance and restriction in 2024 compared to 2023.

RQ3.2: Are there any trends in AI tool adoption between the two consecutive survey years? Yes, there was a substantial descriptive increase in overall familiarity with and adoption of various AI tools, with specific tools like ChatGPT seeing major growth. The use specifically for programming assignments also trended upwards.

The descriptive trends suggest an evolution in students' AI usage and attitudes between 2023 and 2024, but the non-equivalent groups mean that these patterns should not be interpreted as definitive longitudinal change. The clearest year-on-year pattern is a statistically detectable, though modest, shift from fully allowing AI tools towards preferring partial allowance for programming assignments; even here, however, non-equivalent samples and possible repeated respondents warrant caution. Nonetheless, this research indicates faster integration of these tools into academic practice alongside a developing critical awareness regarding their appropriate use (RQ3).

6. Conclusion

This research explored the use of AI tools by students in computer programming education and their perspectives on this use, building upon previous work (Novak & Maček, 2024).

This study underscores the growing integration of AI tools into programming education. While students recognise the value of AI in enhancing efficiency and supporting learning, their perspectives have evolved towards a balanced, partial-allowance model. This model aligns with a view of AI as a supportive tool whose use must remain transparent and verifiable. As AI tools continue to evolve, it is crucial for educators to establish clear guidelines and interventions that promote responsible use, ensuring that students can use these tools effectively without compromising the development of critical thinking and problem-solving skills. Instead of a single, institution-wide policy, guidelines should be tailored to the course level and learning outcomes. For foundational programming courses, where the goal is skill acquisition, policies might restrict AI-generated solutions and require students to demonstrate independent reasoning through in-class work and brief viva voce code walkthroughs. For more advanced courses, where AI increasingly reflects industry practice, policies can be more permissive but should still require disclosure of AI use and traceable code provenance (e.g. via version-control histories).

To maintain academic integrity, assessments should place greater weight on the learning process and on students' ability to verify, critique and adapt AI-assisted code, for example through assignments that involve debugging or improving AI-generated solutions. Recognising that effective GenAI use is a skill, the curriculum should include instruction on prompt engineering and the critical evaluation of AI outputs. This empowers students to use AI as an effective co-pilot rather than an unreliable oracle, mitigating risks associated with inaccurate 'hallucinated' information.

Acknowledgements

This work was supported by the Croatian Science Foundation under the project number IP-2025-02-4117.

References

Ahmed, Z., Shanto, S. S., & Jony, A. I. (2024). Potentiality of generative AI tools in higher education: Evaluating ChatGPT's viability as a teaching assistant for introductory programming courses. STEM Education, 4(3), 165–182. https://doi.org/10.3934/steme.2024011

Ali, W., Alami, R., Alsmairat, M. A. K., & AlMasaeid, T. (2024). Consensus or Controversy: Examining AI's Impact on Academic Integrity, Student Learning, and Inclusivity Within Higher Education Environments. 2024 2nd International Conference on Cyber Resilience (ICCR), 01–05. https://doi.org/10.1109/ICCR61006.2024.10532968

Asgari, M., Tsai, F.-C., Mannila, L., Strömbäck, F., & Sadique, K. M. (2024). Students' perspectives on using digital tools in programming courses. Discover Education, 3(1), 57. https://doi.org/10.1007/s44217-024-00144-4

Aydemir, H., & Seferoğlu, S. S. (2024). Investigation of users' opinions, perceptions and attitudes towards the use of artificial intelligence in online learning environments. European Journal of Open, Distance and E-Learning, 26(s1), 70–92. https://doi.org/10.2478/eurodl-2024-0007

Ayyoub, A., Khlaif, Z., Hamamra, B., Bensalem, E., Mitwally, M., Sanmugam, M., Fteiha, A., Joma, A., Bsharat, T. R. K., Eidah, B. A., & Khaldi, M. (2025). Drivers of Acceptance of Generative AI Through the Lens of the Extended Unified Theory of Acceptance and Use of Technology. Human Behavior and Emerging Technologies, 2025(1). https://doi.org/10.1155/hbe2/6265087

Bain, G., & Barnes, I. (2014). Why is programming so hard to learn? In Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education (p. 356). ACM. https://doi.org/10.1145/2591708.2602675

Balogh, A. (2024). Educational Innovation of Using Artificial Intelligence in University Education: A Comprehensive Student Survey. International Journal on Lifelong Education and Leadership, 10(2), 55–65. https://doi.org/10.25233/ijlel.1528746

Cosma, G., & Joy, M. (2008). Towards a Definition of Source-Code Plagiarism. IEEE Transactions on Education, 51(2), 195–200. 10.1109/TE.2007.906776

Denecke, K., Glauser, R., & Reichenpfader, D. (2023). Assessing the Potential and Risks of AI-Based Tools in Higher Education: Results from an eSurvey and SWOT Analysis. Trends in Higher Education, 2(4), 667–688. https://doi.org/10.3390/higheredu2040039

Ellis, M. E., Casey, K. M., & Hill, G. (2024). ChatGPT and python programming homework. Decision Sciences Journal of Innovative Education, 22(1), 4–15. https://doi.org/10.1111/dsji.12306

Farhi, F., Jeljeli, R., Aburezeq, I., Dweikat, F. F., Al-shami, S. A., & Slamene, R. (2023). Analyzing the students' views, concerns, and perceived ethics about chat GPT usage. Computers and Education: Artificial Intelligence, 5, 100180. https://doi.org/10.1016/j.caeai.2023.100180

Guzdial, M. (2010). Why is it so hard to learn to program. In A. Oram & G. Wilson (Eds.), Making Software: What Really Works, and Why We Believe It (pp. 111–124). O'Reilly Media.

Hou, I., Mettille, S., Li, Z., Man, O., Zastudil, C., & MacNeil, S. (2024). The effects of generative AI on computing students' help-seeking preferences. In Proceedings of the 26th Australasian Computing Education Conference (ACE '24) (pp. 39–48). Association for Computing Machinery. https://doi.org/10.1145/3636243.3636248

Hou, I., Nguyen, H. V., Man, O., & MacNeil, S. (2025). The evolving usage of GenAI by computing students. In Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2 (pp. 1438–1439). ACM. https://doi.org/10.1145/3641555.3705266

Hamamra, B., Khlaif, Z. N., Mayaleh, A., & Baker, A. A. (2025a). A phenomenological examination of educators' experiences with AI integration in Palestinian higher education. Cogent Education, 12(1). https://doi.org/10.1080/2331186X.2025.2526435

Hamamra, B., Khlaif, Z. N., & Mayaleh, A. (2025b). Between promise and precarity: Palestinian university educators' experiences with AI integration in higher education. Interactive Learning Environments, 1–16. https://doi.org/10.1080/10494820.2025.2547264

Idroes, G. M., Noviandy, T. R., Maulana, A., Irvanizam, I., Jalil, Z., Lensoni, L., Lala, A., Abas, A. H., Tallei, T. E., & Idroes, R. (2023). Student Perspectives on the Role of Artificial Intelligence in Education: A Survey-Based Analysis. Journal of Educational Management and Learning, 1(1), 8–15. https://doi.org/10.60084/jeml.v1i1.58

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 [Preprint]. arXiv. https://arxiv.org/abs/2410.06865

Khlaif, Z. N., Ayyoub, A., Hamamra, B., Bensalem, E., Mitwally, M. A. A., Ayyoub, A., Hattab, M. K., & Shadid, F. (2024). University Teachers' Views on the Adoption and Integration of Generative AI Tools for Student Assessment in Higher Education. Education Sciences, 14(10), 1090. https://doi.org/10.3390/educsci14101090

Khlaif, Z. N., Hamamra, B., Bensalem, E., Mitwally, M. A., & Sanmugam, M. (2025). Factors influencing educators' technostress while using generative AI: A qualitative study. Technology, Knowledge and Learning, 1–26.

Lochmiller, C. (2021). Conducting Thematic Analysis with Qualitative Data. The Qualitative Report. https://doi.org/10.46743/2160-3715/2021.5008

Mah, D.-K., & Groß, N. (2024). Artificial intelligence in higher education: exploring faculty use, self-efficacy, distinct profiles, and professional development needs. International Journal of Educational Technology in Higher Education, 21(1), 58. https://doi.org/10.1186/s41239-024-00490-1

Mahmoud, A. B., Kumar, V., & Spyropoulou, S. (2025). Identifying the Public's Beliefs About Generative Artificial Intelligence: A Big Data Approach. IEEE Transactions on Engineering Management, 72, 827–841. https://doi.org/10.1109/TEM.2025.3534088

Novak, M. (2020). Effect of source-code preprocessing techniques on plagiarism detection accuracy in student programming assignments.

Novak, M., Joy, M., & Kermek, D. (2019). Source-code Similarity Detection and Detection Tools Used in Academia. ACM Transactions on Computing Education, 19(3), 1–37. https://doi.org/10.1145/3313290

Novak, M., & Maček, M. (2024). Student perspectives on using Artificial Intelligence tools for solving programming assignments. In N. Vrček & D. Korent (Eds.), Proceedings of the 35th Central European Conference on Information and Intelligent Systems - CECIIS 2024 (pp. 351–363). University of Zagreb Faculty of Organization and Informatics.

Nowell, L. S., Norris, J. M., White, D. E., & Moules, N. J. (2017). Thematic Analysis. International Journal of Qualitative Methods, 16(1). https://doi.org/10.1177/1609406917733847

Okulich-Kazarin, V., Artyukhov, A., Skowron, Ł., Artyukhova, N., & Wołowiec, T. (2024). Will AI Become a Threat to Higher Education Sustainability? A Study of Students' Views. Sustainability, 16(11), 4596. https://doi.org/10.3390/su16114596

Ou, A. W., Stöhr, C., & Malmström, H. (2024). Academic communication with AI-powered language tools in higher education: From a post-humanist perspective. System, 121, 103225. https://doi.org/10.1016/j.system.2024.103225

Prather, J., Denny, P., Leinonen, J., Becker, B. A., Albluwi, I., Craig, M., Keuning, H., Kiesler, N., Kohn, T., Luxton-Reilly, A., MacNeil, S., Petersen, A., Pettit, R., Reeves, B. N., & Savelka, J. (2023). The robots are here: Navigating the generative AI revolution in computing education. In Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education (ITiCSE-WGR '23) (pp. 108–159). Association for Computing Machinery. https://doi.org/10.1145/3623762.3633499

Rajendran, V., & Ramasamy, R. K. (2024). Artificial Intelligence Integration in Programming Education: Implications for Pedagogy and Practice (pp. 197–206). https://doi.org/10.1007/978-981-97-7794-5_16

Saari, M., Rantanen, P., Nurminen, M., Kilamo, T., Systä, K., & Abrahamsson, P. (2024). Survey of AI Tool Usage in Programming Course: Early Observations (pp. 182–191). https://doi.org/10.1007/978-3-031-48550-3_18

Smolansky, A., Cram, A., Raduescu, C., Zeivots, S., Huber, E., & Kizilcec, R. F. (2023). Educator and student perspectives on the impact of generative AI on assessments in higher education. In Proceedings of the Tenth ACM Conference on Learning @ Scale (pp. 378–382). ACM. https://doi.org/10.1145/3573051.3596191

Yilmaz, R., & Yilmaz, F. G. K. (2023). Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning. Computers in Human Behavior: Artificial Humans, 1(2), 100005. https://doi.org/10.1016/j.chbah.2023.100005

Appendix A: Questionnaire (English translation)

Introductory text: This survey is part of the study *"Students' views on AI tools in the context of solving programming assignments"*. The survey is anonymous and the data will be reported in aggregated form only.

Q1. Faculty *Single-response.*

Select or write the name of the faculty you are attending.

□ University of … □ Other: ______

Q2. Level of study *Single-response.*

Select your current level of study.

□ Undergraduate □ Graduate (Master's)

Q3. Study programme *Open-ended.*
Please write the study programme (major) you are attending.

Q4. Year of study *Single-response.*
Select or state the year of study you are currently in.
□ 1st year – undergraduate □ 2nd year – undergraduate □ 3rd year – undergraduate
□ 1st year – graduate (Master's) □ 2nd year – graduate (Master's)
□ Other: ______

Q5. Average grade *Open-ended (numeric).*
What is your current average grade on your study programme?

Q6. Prior use of AI tools *Single-response; required.*
Have you tried using AI tools such as ChatGPT, Microsoft Copilot, etc.?
□ Yes □ No

Q7. AI tools used *Multiple-response.*
Please indicate which AI tools you have used.
□ ChatGPT □ Bing Chat □ BlackBox AI □ Microsoft Copilot □ DALL·E □ Bard □ Other: ______

Q8. Use of AI tools for non-programming assignments *Single-response.*
Have you used AI tools for assignments that are **not** programming (e.g. seminar papers, essays, etc.)?
□ Yes □ No

Q9. How AI tools were used for non-programming assignments *Open-ended.*
In what ways have you used AI tools for assignments that are **not** programming (e.g. seminar papers, essays, etc.)? Please describe in as much detail as you can.

Q10. Use of AI tools for programming assignments *Single-response.*
Have you used AI tools for solving programming assignments?
□ Yes □ No

Q11. How AI tools were used for programming assignments *Open-ended.*
In what ways have you used AI tools for solving programming assignments? Please describe in as much detail as you can.

Q12. General stance on allowing AI tools for programming assignments *Single-response.*
Regarding programming assignments, do you think that AI tools should:
□ Be fully allowed for solving assignments (please explain in the next question)
□ Be completely prohibited for solving assignments (please explain in the next question)
□ Be partially allowed and partially prohibited (please explain in the next question)

Q13. Explanation of your stance on allowing or prohibiting AI tools *Open-ended*
Do you think that AI tools should be allowed for solving programming assignments, or should they be

prohibited? Please explain your reasoning and, if possible, specify for which types of assignments they should or should not be used.

Q14. Overall impression of AI tools in the study context *Open-ended.*
What is your overall impression of these AI tools? What advantages and disadvantages do you see in the context of studying?

Q15. Additional comments *Open-ended.*
If there is anything else you would like to add, please write it here.

Appendix B: Exploratory 2 × 2 comparisons of AI tool use between 2023 and 2024

AI tool	2023 users n (%)	2024 users n (%)	OR (2024 vs. 2023)	95% CI for OR	p-value (two-sided)
ChatGPT	62 (73.8%)	181 (99.5%)	64.23	8.48 – 486.42	< 0.001
Microsoft Copilot	12 (14.3%)	100 (54.9%)	7.32	3.72 – 14.40	< 0.001
Bing Chat	8 (9.5%)	68 (37.4%)	5.67	2.58 – 12.46	< 0.001
DALL-E	1 (1.2%)	42 (23.1%)	24.90	3.36 – 184.29	< 0.001
BlackBox AI	3 (3.6%)	22 (12.1%)	3.71	1.08 – 12.77	0.047
Gemini	4 (4.8%)	40 (22.0%)	5.63	1.94 – 16.32	< 0.001
Other	9 (10.7%)	8 (4.4%)	0.38	0.14 – 1.03	0.091

$N_{(2023)} = 84$; $N_{(2024)} = 182$.
Percentages are the proportion of students in each year who reported using the tool at least once (multiple-response item; percentages do not sum to 100%).
Odds ratios (OR) > 1 indicate higher odds of reporting use in 2024 than in 2023; OR < 1 indicates lower odds.
p-values are based on 2 × 2 tests of association between year (2023 vs 2024) and use of each tool (yes/no), using Pearson's chi-square test with Yates's continuity correction.
Bonferroni-adjusted p-values are computed for a family of seven per-tool comparisons (α_adj ≈ 0.05/7 ≈ 0.007).