2026 ASEE ANNUAL CONFERENCE June 21–24, 2026 | Charlotte Convention Center, NC
Host Sponsor UNC Charlotte
Paper ID #51595

Exploring Student Engagement with AI in a Mechatronics Course

Dr. Hope Leigh Weiss, California State University, Fullerton

Dr. Hope L. Weiss is currently an Associate Professor of Mechanical Engineering at California State University, Fullerton. She earned her B.S. in Mechanical and Aerospace Engineering from Cornell University, and her M.S. and Ph.D. in Mechanical Engineering from the University of California, Berkeley.

Exploring Student Engagement with AI in a Mechatronics Course

Abstract

This study examines student perceptions and use of artificial intelligence (AI) tools in an upper-level mechatronics elective for mechanical engineering students, as well as their broader engagement with AI in other courses. Within the course, students were permitted to use AI tools but must physically implement hardware systems and orally explain their code to verify comprehension. To further examine AI use, a required code conversion assignment asked students to use AI to translate a previous assignment's code between programming environments and reflect on the accuracy, usability, and trustworthiness of the output. Data sources included an anonymous pre-assignment survey (n = 39) with open-ended prompts on perceived advantages and risks of AI, and post-assignment reflections collected through Gradescope (n = 35). Results showed near-universal AI use prior to (94.9%) and during the course (97.4%). Perceived helpfulness was high (mean = 4.08 on a 1–5 scale), while trust in AI-generated technical information was more moderate (mean = 3.15). Open-ended responses most often described AI as a personal tutor and as support for coding and debugging, with primary concerns focused on inaccurate information and overreliance. In the code conversion assignment, students rated AI translation effectiveness highly (mean = 4.20) and reported a high likelihood of future AI use for programming (mean = 4.09), while trust in AI-generated code remained more cautious (mean = 3.34). Overall, results suggest that structured, hands-on assignments can support productive AI use while still requiring verification, debugging, and explanation to promote learning and accountability.

Introduction

Generative artificial intelligence (AI) tools are now widely accessible to students and are increasingly used to support studying, writing, and problem solving. Large-scale surveys report widespread student use of tools such as ChatGPT for tasks including brainstorming, summarizing course materials, and simplifying complex information [1]. These tools are generally viewed as useful for efficiency and idea generation, while also raising concerns related to accuracy, academic integrity, and overreliance [2]. These concerns are particularly relevant in engineering education, where correctness, reasoning, and verification are central learning outcomes.

Prior work in engineering education has examined AI integration from multiple perspectives, including student perceptions, instructional effectiveness, ethical considerations, and professional skill development [3]–[7]. These studies generally find that AI can support problem solving, conceptual understanding, and efficiency when used as a supplemental resource, while also identifying risks related to inaccurate outputs, reduced critical thinking, and overreliance when verification is absent. Many authors note that AI adoption within mechanical engineering curricula is inconsistent.

Student perception studies across engineering disciplines indicate that students value AI for explanation, brainstorming, and debugging, while remaining cautious about unverified results and emphasizing the need for independent reasoning [4], [7]. Course-level implementations of AI in areas such as introductory thermodynamics and senior design further suggest that carefully structured assignments can leverage AI to enhance learning while maintaining rigor [5], [6].

The present study examines student perceptions of AI use in an upper-level mechatronics technical elective for mechanical engineering students. In this course, students build and test physical systems and are required to orally explain their code. A required code conversion assignment asks all students to use AI to translate their working program from a previous assignment from one programming environment to another, test the AI-generated code on hardware, correct errors as needed, and reflect on the reliability and learning value of the output. By combining a pre-assignment survey, open-ended prompts on perceived advantages and risks, and post-assignment reflections tied to a hands-on programming task, this study examines how student perceptions of AI align with an experience that requires verification and explanation.

Methods

This study was conducted in EGME 456: Introduction to Mechatronics for Engineers at California State University, Fullerton, an upper-level technical elective for mechanical engineering students during Fall 2026. The course emphasizes mechatronic system integration using an Arduino-based platform, including sensors, actuators, and embedded control. A sequence of laboratory exercises culminates in functioning hardware systems. Artificial intelligence (AI) tools were permitted for coursework, including programming and debugging, but students were required to demonstrate understanding through physical implementation and orally explain their code either through informal one-on-one interactions during laboratory sessions or office hours, or through recorded video submissions demonstrating system operation and code understanding.

The laboratory systems used in this course are Arduino-based mechatronic assignments that require integration of hardware and embedded control logic. In Lab 1, students design a digital input/output system that simulates a theme park ride controller, where pushbuttons control transitions between states (e.g., standby, loading, running) represented by LEDs, including timed sequences and an emergency stop condition. In Lab 2, students develop an analog control system in which sensor inputs are used to control actuator behavior, including mapping measured signals to motor output and implementing both automatic and manual control. These assignments require students to combine hardware wiring with software logic, including conditional statements, timing behavior, and analog signal interpretation. Later laboratories introduce additional complexity, including the use of Arduino libraries and implementation of motor control circuits using transistors. The code conversion assignment was limited to Lab 1 and Lab 2 systems to focus on core control logic and hardware interaction. As a result, successful conversion required not only translation between Arduino and MATLAB syntax, but also correct interpretation of system behavior.

All students completed a required code conversion assignment designed to engage AI tools while requiring verification and debugging. Each student recreated a previously completed laboratory hardware setup, submitted their original working code, and then used an AI tool to translate the code between programming environments (Arduino and MATLAB). Students were instructed not to modify the AI-generated code initially, after which they tested the translated code on hardware, identified errors or mismatches, and corrected the code as needed. Students then completed a structured reflection addressing code functionality, perceived learning benefit, perceived effectiveness of AI for translation, likelihood of future AI use for programming, trust in AI-generated code in engineering contexts, and the extent to which AI supported understanding of program logic and system behavior.

Two primary data sources were used: an anonymous pre-assignment survey administered prior to the code conversion assignment and post-assignment reflections collected through Gradescope. The pre-assignment survey included Likert-scale items and open-ended questions addressing prior AI exposure, AI use in the course, perceived helpfulness of AI, trust in AI-generated technical information, and perceived advantages and risks of AI use in engineering coursework. Likert-scale items used five-point response scales. The exact wording of the pre-assignment survey items and code conversion assignment prompts is provided in Appendix A.

The course enrollment consisted of 37 students. The anonymous pre-assignment survey yielded 39 responses. Because the survey was anonymous and not restricted to a single submission per user, it is possible that one or more students submitted the survey more than once. Duplicate responses could not be identified or removed; however, given the small discrepancy between enrollment and response count and the consistency of themes across responses, this limitation is not expected to meaningfully impact the results. Post-assignment reflections were collected from 35 students. Open-ended responses were analyzed using thematic coding, with responses allowed to contribute to multiple themes. Initial theme groupings were generated using an AI tool and then reviewed and refined by the author.

Results

Pre-assignment survey responses indicated near-universal prior exposure to AI tools, with 94.9% of respondents reporting prior AI use and 97.4% reporting AI use in the course. AI use in the course included multiple purposes, with 92.1% of students reporting use for debugging or troubleshooting code and 68.4% reporting use for explaining code or concepts. AI was rated as generally helpful for the course (Question 9), with a mean perceived helpfulness of 4.08 on a five-point scale. Trust in AI-generated technical information (Question 13) was more moderate (mean = 3.15). The distributions of perceived trust and helpfulness ratings are shown in Figure 1.

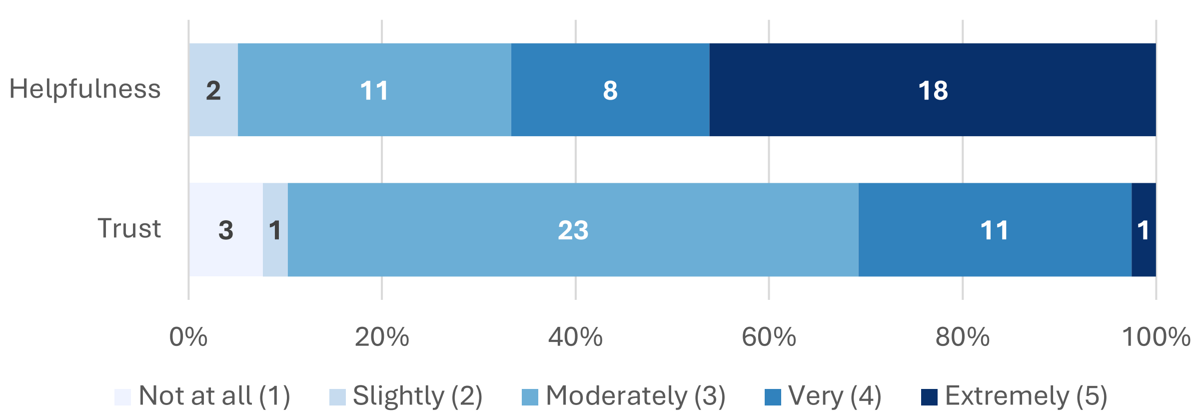

Figure 1. Pre-assignment survey results for perceived helpfulness of AI in the course (Question 9) and trust in AI-generated technical information (Question 13).

In response to the open-ended prompt asking about advantages of AI in engineering coursework (Question 20), AI was most frequently described as an always-available resource for explanation and clarification. Coding support and time savings were also commonly cited, while fewer responses emphasized brainstorming and information organization. Theme frequencies for perceived advantages are shown in Figure 2. One student summarized the tutoring role of AI by describing it as "like a constantly open office hours."

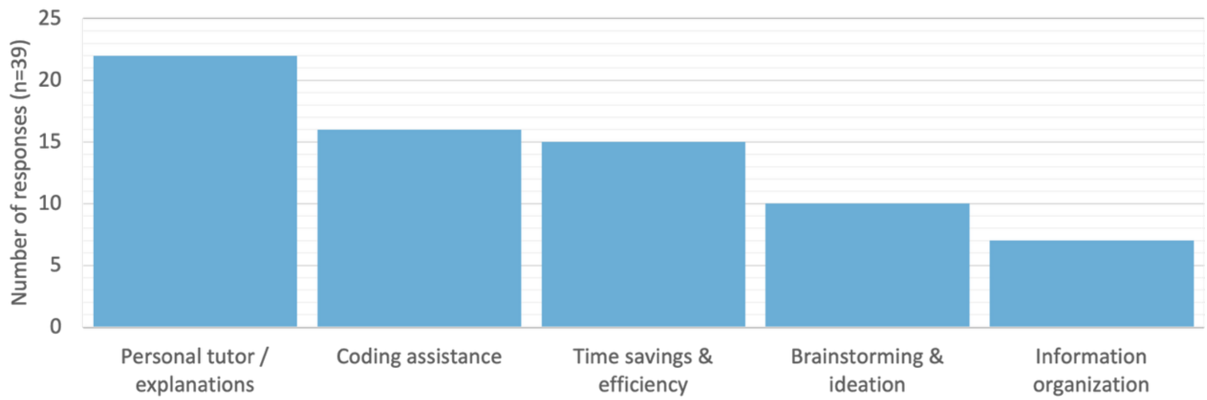

Figure 2. Theme frequencies for perceived advantages of AI in engineering coursework (Question 20).

Responses to the prompt asking about drawbacks or risks (Question 21) emphasized two dominant concerns: inaccurate information and overreliance. AI outputs were frequently described as incorrect or misleading, including hallucination-like behavior. A second recurring concern was reduced learning when AI is used as a substitute for reasoning or practice. Additional themes included reduced critical thinking, academic integrity risks, and loss of creativity. Theme frequencies for perceived drawbacks and risks are shown in Figure 3.

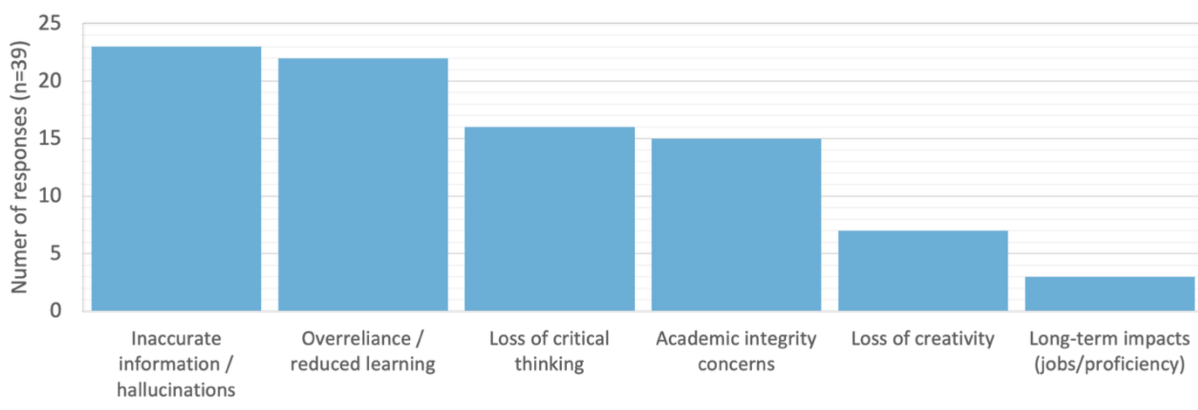

Figure 3. Theme frequencies for perceived drawbacks or risks of AI in engineering coursework (Question 21).

Post-assignment reflections (n = 35) indicated that AI-assisted cross-language code translation was generally perceived as effective, while still requiring verification. On the effectiveness item (Question 5.6), 51.4% selected "Extremely effective" and 25.7% selected "Very effective," yielding a mean rating of 4.20. A high likelihood of future AI use for programming (Questions 5.5) was also reported (mean = 4.09). Perceived learning benefit (Questions 5.4) was more moderate (mean = 3.09), with "Slightly helpful" being the most common response. Trust in AI-generated code (Question 5.7) remained cautious to moderately positive (mean = 3.34), with responses concentrated in the middle categories. Moderate perceived support for understanding programming logic and system behavior (Question 5.8) was observed (mean = 3.06). Effectiveness and trust ratings are shown in Figure 4.

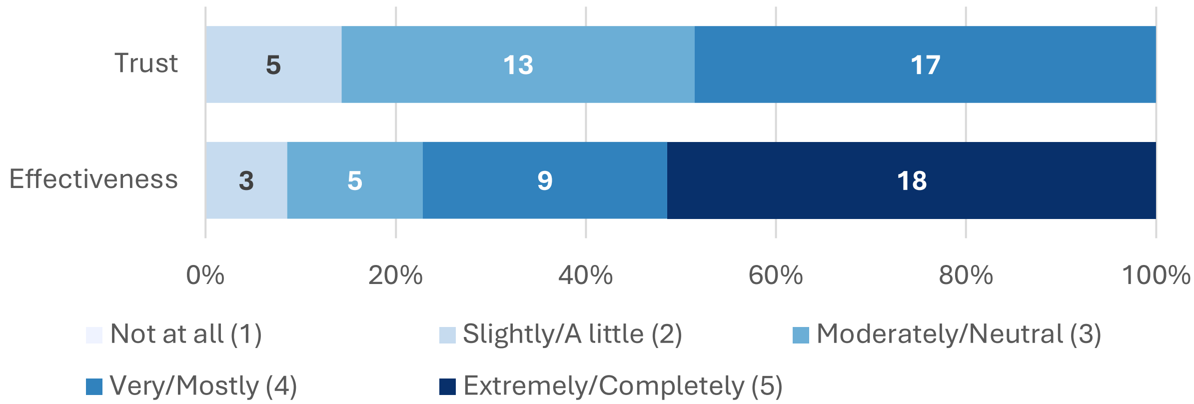

Figure 4. Code conversion assignment Likert-scale responses for perceived effectiveness of AI-assisted code translation (Question 5.6) and trust in AI-generated code (Question 5.7) (n = 35).

Discussion

The results indicate that students primarily view AI as an always-available support resource for explanation and coding assistance, while also recognizing that AI outputs should not be treated as authoritative. Caution was expressed in contexts where correctness is not immediately verifiable, such as theory or analysis. The code conversion assignment reflects these perceptions by requiring students to test AI-generated code on physical hardware and correct errors, thereby emphasizing verification and debugging as core engineering behaviors. Although students rated AI translation effectiveness highly, reported learning gains were more moderate, suggesting that AI can accelerate production of functional code without necessarily yielding equivalent conceptual understanding. Requiring physical implementation, oral explanation, and structured reflection appears to support accountability and encourage critical evaluation of AI reliability.

Instructor Reflections

Early in the semester, many students struggled to orally explain their code despite having functional programs, particularly when describing control logic and programming constructs. Over four laboratory assignments, students demonstrated increased confidence in explaining program flow and software–hardware interactions. This observation is consistent with pre-assignment survey results, which showed an increase in self-reported confidence in writing and explaining code from a mean of 2.49 before the course to 3.95 by the time of the assignment after the four laboratory assignments.

During their code explanations, students were often encouraged to use AI as a learning aid rather than a solution generator, such as rewriting unfamiliar constructs using simpler logic. Post-assignment reflections indicated that AI provided moderate support for understanding programming logic and system behavior for this assignment, suggesting that while AI was effective for generating and refining code, its contribution to deeper conceptual understanding was more limited in this context. This pattern suggests that future iterations of the assignment could benefit from additional scaffolding that emphasizes explanation and reasoning, as well as extension to systems involving Arduino libraries, which introduce additional abstraction and may present different challenges for AI-assisted code translation.

The overall classroom atmosphere appeared more positive than in previous offerings, with greater student engagement in debugging and iteration. However, multiple changes occurred concurrently, including the explicit allowance of AI tools and a shift to a competition-based semester project, and cohort-specific dynamics likely contributed. This course was the first taught by the author to explicitly allow AI use, and the results will inform how AI is incorporated in future offerings of this course and other courses taught by the author. These observations therefore cannot be attributed solely to AI integration but suggest that transparent AI policies and structured accountability through implementation and explanation may support student engagement and confidence.

References

[1] D. Ravšelj, D. Keržič, N. Tomaževič, L. Umek, N. Brezovar, N. A. Iahad, et al., "Higher education students' perceptions of ChatGPT: A global study of early reactions," *PLoS ONE*, vol. 20, no. 2, p. e0315011, 2025

[2] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, et al., "ChatGPT for good? On opportunities and challenges of large language models for education," *Learning and Individual Differences*, vol. 103, p. 102274, 2023.

[3] R. D. Manteufel, "Assessing the impact of artificial intelligence on undergraduate mechanical engineering education," in *Proceedings of the 2025 ASEE Annual Conference & Exposition*, Arlington, TX, USA, 2025.

[4] S. A. Wilkerson, S. F. Kiefer, and Y. T. G. de Melo, "Reflections on artificial intelligence use in engineering courses," in *Proceedings of the 2025 ASEE Annual Conference & Exposition*, Montreal, QC, Canada, 2025.

[5] J. D. Christopher, V. I. Bongioanni, and L. V. Scharff, "Applications of artificial intelligence in an introductory thermodynamics course," in *Proceedings of the 2025 ASEE Annual Conference & Exposition*, Montreal, QC, Canada, 2025.

[6] W. Li, Y. Hung, G. Altuger-Genc, S. Zhang, A. Tatoglu, and Z. Zhang, "Integrating AI/ML learning in senior projects for mechanical engineering technology students," in *Proceedings of the 2025 ASEE Annual Conference & Exposition*, Montreal, QC, Canada, 2025.

[7] M. N. Sakib, M. Manzo, and R. C. Yalamanchili, "Navigating the impact of AI in engineering education: Enhancing learning outcomes and addressing ethical and assessment challenges," in *Proceedings of the 2025 ASEE Annual Conference & Exposition*, Arlington, TX, USA, 2025.

Appendix A: Survey and Assignment Prompts

The following prompts are reproduced verbatim from the survey and assignment, with formatting condensed for readability.

A.1 Pre-Assignment Survey (Anonymous Google Form)

1. What is your academic year?
   ( ) Junior ( )Senior ( ) Graduate student
2. Have you taken any other courses that explicitly allow or discuss AI tool use?
   ( ) Yes ( ) No ( ) Not sure
3. Have you used AI tools (e.g., ChatGPT, Copilot, Gemini, etc.) before this class (EGME 456)?
   ( ) Yes ( ) No
4. If yes, what AI tools have you used? (Select all that apply)
   ( ) ChatGPT ( ) Copilot ( ) Google Gemini ( ) Claude ( ) Other (please specify)
5. For what purposes have you previously used AI tools? (Select all that apply)
   ( ) Writing/explaining code ( ) Debugging ( ) Writing reports ( ) Studying concepts ( ) Brainstorming ( ) Other (please specify)
6. Have you used AI tools for this class so far?
   ( ) Yes ( ) No
7. If yes, what AI tools have you used in this class? (Select all that apply)
   ( ) ChatGPT ( ) Copilot ( ) Google Gemini ( ) Claude ( ) Other (please specify)
8. For what purposes have you used AI tools in this class? (Select all that apply)
   ( ) Debugging ( ) Translating code ( ) Writing reports ( ) Explaining concepts ( ) Brainstorming ( ) Other
9. How helpful has AI been for you in this class so far?
   (1 = Not helpful at all, 5 = Extremely helpful)
10. How confident are you in writing and explaining your own code before this course?
    (1 = Not confident at all, 5 = Very confident)
11. How confident are you in writing and explaining your own code now?
    (1 = Not confident at all, 5 = Very confident)
12. Do you attribute any perceived improvement in your ability to write and explain your own code to AI?
    ( ) Yes – mostly because of AI ( ) Yes – partly because of AI ( ) No – improvements were mostly from practice, feedback, or explaining my code ( ) No – I did not notice improvement ( ) Unsure
13. How much do you trust AI-generated technical information?
    (1 = Not at all, 5 = Completely)
14. How likely are you to use AI tools for programming or problem-solving in the future?
    (1 = Very unlikely, 5 = Very likely)
15. Have you used AI tools in any other Mechanical Engineering courses?
    ( ) Yes ( ) No
16. If yes, in which types of courses have you used AI tools? (Select all that apply)
    ( ) Programming or computational courses (e.g., MATLAB, Python) ( ) Design or CAD courses ( ) Laboratory or experimental courses ( ) Theoretical or analysis-heavy courses ( ) Writing-intensive or communication-focused courses ( ) Other (please specify)
17. In your other Mechanical Engineering courses, how have instructors handled AI tool use? (Select all that apply)
    ( ) Encouraged by the instructor ( ) Allowed but not discussed ( ) Discouraged or prohibited ( ) Unclear or not mentioned

18. How often do you use AI tools for coursework outside this class?
    ( ) Never ( ) Once or twice a semester ( ) Occasionally (a few times per course) ( ) Frequently (for most assignments) ( ) Almost always
19. How do you feel about using AI tools in your engineering education overall?
    (1 = Very negative, 5 = Very positive)
20. What do you see as the biggest advantages of using AI in your engineering coursework?
21. What do you see as the biggest drawbacks or risks?

A.2 Code Conversion Assignment Prompts (Gradescope Assignment)

<u>Part 1 – System Setup</u>

Recreate your mechatronic system from Lab 1 or Lab 2. Make sure it's properly wired, functional, and running with your original Arduino or MATLAB code.

1.1 Upload your original working code as a reference before conversion. Make sure your comments in your code indicate whether it was Lab 1 or Lab 2 near the top of your code.

1.2 Write a short note explaining whether you used AI tools (e.g., ChatGPT, Copilot, etc.) to help write or debug your original Lab 1 or Lab 2 code.
If yes, briefly describe how you used AI (for example: to fix syntax errors, generate initial code structure, suggest debugging steps, etc.). AI use is allowed in this class, this question is for transparency only and will not affect your grade.

<u>Part 2 – Prediction</u>

Before using AI, think critically about how your code might translate between Arduino and MATLAB.
Write a short paragraph (5–8 sentences) predicting:

- Whether you expect the AI to successfully convert your code.
- Which parts of the code might not translate well (e.g., syntax, timing functions, serial communication).
- What debugging or modifications you might expect to need.

<u>Part 3 – AI Code Conversion</u>

Use an AI assistant (e.g., ChatGPT, Copilot, or another LLM-based tool) to convert your original code:
From Arduino → MATLAB or MATLAB → Arduino.
Your prompt should clearly instruct the AI to convert your code from Arduino to MATLAB or MATLAB to Arduino. You can copy/paste your conversation into a document or take screenshots and save them as a single PDF.

3.1 Save the prompt as a pdf and upload it below:

3.2 Upload the generated code (as a .m or .ino file). Do NOT modify the code initially.

<u>Part 4 – Testing and Debugging</u>

1. Test the converted code on your hardware setup.
2. If it doesn't work:

- Identify specific issues.
- Modify or debug the AI-generated code to make it functional.
- Clearly note what changes were required and why.

If you needed to edit the code, upload the new code here:

<u>Part 5 – Reflection Questions</u>

Reflect on your experience using AI for cross-language code translation.

5.1 Was your prediction accurate? Compare your expectations to what actually happened.

5.2 Did the AI-generated code work immediately? Describe how well the AI code ran without changes.

5.3 What parts of the code did not translate correctly, if any? And what changes did you make to get it working, if needed?

5.4 How much did using AI in this assignment help you learn about programming or engineering systems?
Select one option below and briefly explain your reasoning in a few sentences — for example, whether this experience increased your understanding, raised new questions, or changed how you think about coding.
( ) Not at all helpful ( ) Slightly helpful ( ) Moderately helpful ( ) Very helpful ( ) Extremely helpful

5.5 How likely are you to use AI tools for programming in the future based on this assignment?
Select one option below and briefly explain your reasoning in a few sentences — for example, whether this experience increased your confidence, raised new concerns, or changed your perception of AI's usefulness for coding.
( ) Very unlikely ( ) Unlikely ( ) Neutral ( ) Likely ( ) Very likely

5.6 How effective do you think AI was in helping you translate or improve your code for this assignment?
Select one option below and briefly explain your reasoning in a few sentences — for example, whether it saved time, improved your results, or required significant correction.
( ) Not effective at all ( ) Slightly effective ( ) Moderately effective ( ) Very effective ( ) Extremely effective

5.7 After completing this assignment, how much do you trust AI-generated code in engineering contexts?
Select one option below and briefly explain your reasoning in a few sentences — for example, whether your trust increased, decreased, or stayed the same based on the AI's accuracy and reliability.
( ) Not at all ( ) A little ( ) Neutral ( ) Mostly ( ) Completely

5.8 To what extent did using AI help you understand the programming logic or system behavior?
Select one option below and briefly explain your reasoning in a few sentences — for example, whether the AI's output clarified your understanding, confused you, or had little effect.
( ) Not at all ( ) A little ( ) Somewhat ( ) Quite a bit ( ) A great deal