Human-human-AI pair programming

Feature	Justification	Example
AI as a technical copilot		
Providing hints	Providing hints is useful when the students are stuck, which allows them to make progress [11]. Giving hints is preferable to giving the solutions directly because prior studies mentioned that some users prefer to still have control while being assisted by a copilot ("Do it with me instead of do it for me" [10]). Additionally, providing code solutions directly will reduce the code ownership and understanding, and increase overreliance [7]	Send hints to the navigator first so that the navigator can discuss it with the driver
Providing decomposed solutions when users are still stuck	If students are still stuck after being provided with hints, then the AI could give a solution to move on to progress the learning process [11]. While giving the solution, it is better to decompose it into multiple steps to ensure students have better code understanding [7].	Give the step-by-step solution with an explanation on the chat
Issue check after each code block	In pair programming, the navigator has a role to review the code while the driver is coding. Pointing out the issue directly can reduce the cognitive burden in the end when issues accumulate. Moreover, Immediate feedback has been shown to improve student learning [9]	Intervene the navigator after 5s: *"This line has some issues, what do you think?"*
AI as a communication facilitator		
Proactively help when stuck (silence ≥ 30 s or ≥ 3 identical errors)	Prior works mentioned the Four Goals of Productive Discussion (FGPD), which includes that people should actively share their thoughts and engage with each other [1]. Hence, silence for too long is not productive.	Intervene both users: *"Would you like to share some of your thoughts?"*

		If users are still stuck, then proceed to give hints to both
Addressing misdirection (When discussing the wrong idea for ≥ 30 s)	Prior studies have empirically shown that redirecting students back to the right direction can improve dialogue productivity [1]	Intervene both users: *"I think there might be some issue with idea X, since it is …., what do you think?"* If users are still stuck, then proceed to give hints to both
Dealing with imbalance (One student dominating too much)	One significant challenge of pair programming is the potential skill level mismatch between partners. When there is a considerable disparity in programming abilities, the collaboration may become less effective, with the more skilled student potentially dominating the process or becoming frustrated with their partner's pace [2].	Intervene the silent user: *"What do you think about your friend's idea?"* Intervene the dominant user: *"Pause for a moment and summarise your idea in one sentence so your partner can follow."*
AI as a learning facilitator		
Planning	Researchers theorize that problem-solving has four distinct principles: understanding the problem, devising a plan, carrying out the plan, and reflecting on the outcome [3, 4]. Hence, understanding the problem and having a plan before writing the code directly is important [3]. However, novice programmers often do not do enough planning [3] and have unorganized thought processes [5]. Having a solid plan is even more important when coding with AI, as students often face metacognitive difficulties [6]	Intervene both users: *"Can you describe your approach in steps before coding?"* If the plan does not sound good, challenge both users: *"This part might cause an issue because …, what do you all think?"*
Reflection-in-action to assess code understanding and develop metacognition while using an AI assistant	Prior work shows that overreliance on AI can cause students to lose their understanding of the code [7]. This lack of code understanding could present issues in maintainability and validation [8].	*"Can you walk me through what your code does up to this point?"* *or*

		"Why did you choose that approach?",
Reflection-on-action after completing the task	"Reflection activity allows students to conceptualize the experience and raise their knowledge to a higher-level meta-cognitive aspect." [12]. This is especially important when using AI helps in the learning process to ensure code understanding [7].	Ask both users to explain and discuss

References

[1] Investigating the Impact of a Collaborative Conversational Agent on Dialogue Productivity and Knowledge Acquisition
https://link.springer.com/article/10.1007/s40593-025-00469-7

[2] The impact of AI-assisted pair programming on student motivation, programming anxiety, collaborative learning, and programming performance: a comparative study with traditional pair programming and individual approaches
https://stemeducationjournal.springeropen.com/articles/10.1186/s40594-025-00537-3

[3] Exploring Differences in Planning between Students with and without Prior Experience in Programming

[4] G. Polya. How to solve it: A new aspect of mathematical method, volume 85. Princeton university press, 2004.

[5] On the Design and Development of a UML-Based Visual Environment for Novice Programmers
https://www.informingscience.org/Publications/234

[6] "It's Weird That it Knows What I Want": Usability and Interactions with Copilot for Novice Programmers
https://arxiv.org/pdf/2304.02491

[7] Assistance or Disruption? Exploring and Evaluating the Design and Trade-offs of Proactive AI Programming Support
https://arxiv.org/pdf/2502.18658

[8] Jenny T. Liang, Chenyang Yang, and Brad A. Myers. 2024. A Large-Scale Survey on the Usability of AI Programming Assistants: Successes and Challenges. In Proceedings of the IEEE/ACM 46th International Conference on Software Engineering (Lisbon, Portugal) (ICSE '24). Association for Computing Machinery, New York, NY, USA, Article 52, 13 pages. doi:10.1145/3597503.3608128

[9] How Helpful do Novice Programmers Find the Feedback of an Automated Repair Tool? https://arxiv.org/pdf/2310.00954

[10] https://aprilwang.me/assets/pubs/CHI25_automation.pdf

[11] An Evaluation of Data-Driven Programming Hints in a Classroom Setting https://pmc.ncbi.nlm.nih.gov/articles/PMC7334677/#:~:text=working%20without%20instructor%20assistance%20,with%20little%20additional%20instructor%20effort

[12] Chang, B. (2019). Reflection in learning. Online Learning, 23(1), 95-110. doi:10.24059/olj.v23i1.1447 https://files.eric.ed.gov/fulltext/EJ1210944.pdf