Developing Secondary Mathematics Prospective Teachers' Confidence in Using Programming in Mathematics Classrooms: Benefits of the Use of Large Language Models

Hyejin Park and Eric Manley (Drake University)

The Association of Mathematics Teacher Educators' (2022) position statement on technology states that mathematics teacher educators (MTEs) should provide teacher candidates opportunities to explore and use different mathematics technologies, including computer programming and robotics, to promote their understanding of mathematical concepts. It also emphasizes the critical role of MTEs in supporting teachers in effectively integrating technology into mathematics teaching. Studies have shown that a variety of barriers often prevent teachers from actually implementing Computer Science (CS) learning in their classrooms, including limited teacher preparation and professional development and low teacher self-efficacy (Celepkolu et al., 2020; Vivian et al., 2020). Studies on the integration of CS learning into mathematics curricula have shown that teachers need more support in actually connecting the mathematics and CS concepts (e.g., Kim et al., 2022; Stigberg & Stigberg, 2020). Our recent study (Park & Manley, 2024) shows that engaging pre-service teachers (PSTs) in programming-integrated mathematics learning activities promoted their understanding of mathematics concepts. It also led to an increase in their confidence in using block-based programming to teach mathematics. However, we observed that their confidence in using text-based programming remained lower, with some PSTs commenting on their inability to create text-based programs that can support students' mathematics learning.

Large Language Models (LLM), like those behind ChatGPT, have demonstrated the ability to generate correct programming code in response to natural-language prompts (Chen et al., 2021). This raises the question of whether LLM-generated code can be used to support teachers in creating programming-integrated mathematics learning activities and, therefore, increasing teacher confidence in implementing such activities in their classrooms. To examine this, we engaged PSTs in task design activities in which they created a programming-embedded mathematics learning task with ChatGPT assistance.

Descriptions of the Geometry Content Course

During the Fall 2023 semester, we designed a series of programming-embedded geometry learning (PGL) in-class activities, and we used them in a geometry content course for secondary mathematics teachers. We designed and revised the PGL materials throughout the semester, considering how students reacted to each PGL activity. Seven prospective secondary mathematics teachers were enrolled in the course during that semester, and four (Ellie, Grace, Charlie, and Lynn) voluntarily participated in our study. All were undergraduates except Charlie, a Master of Arts in Teaching student. Charlie had some teaching experience as a volunteer, though none of the other participants had any practicum experience. Before taking the geometry

content course, they all took an Introduction to Computer Science course where they learned about text-based programming using Python, a required course for secondary mathematics education majors. We engaged students in block-based and text-based programming activities by implementing the PGL materials while learning geometry and proving. See Table 1 for the sequence of programming-embedded activities. After experiencing designing, revising, interpreting, and evaluating different programs throughout the class, we asked students to create programming-embedded geometry learning tasks and use them to teach their peers as part of the final project assignment. For this work, we engaged students in discussing the connections between K-12 CS and mathematics practices and the relationships between programming and proving. After this discussion, we introduced them to ChatGPT as an AI assistant to generate program code. We discussed things they should consider before writing code with ChatGPT assistance, such as a learning goal and the kinds of programming that might be suitable for their tasks. Students tried to come up with some programs by prompting ChatGPT for assistance and began brainstorming about how to design their tasks. Students were asked to use ChatGPT only to generate computer code, not to create a lesson plan. Students implemented their tasks by teaching their classmates as if they were the target students. Each student prepared teaching materials (e.g., student worksheets or slides) and was given 15 minutes to teach. Afterward, they wrote a reflection paper about the experience.

Table 1

Programming-Integrated Geometry Learning Activities Used in the Geometry Content Course

Week	In-class Activities	Assignment
2-3	Interpreting, revising, evaluating, and designing block-based (EdScratch) programs for triangle explorations and constructions[1]	Drawing and exploring shapes using Scratch
4	Interpreting a Python turtle graphics program: The application of the polygon angle sum theorem	Writing a reflection on the use of programming as a geometric tool based on experience in class activities
5	Assessment: Interpreting, revising, evaluating, and designing text-based (Python turtle) programs for triangle explorations and constructions; and comparing the similarities and differences of Scratch, EdScratch, and Python Turtle Graphics.	
14	Discussion of CS learning and using ChatGPT to generate computer code	Designing programming-embedded geometry learning tasks with ChatGPT assistance
15	Teaching: Task Implementation	Preparing to teach using the task created & Writing a reflection on the task design and implementation activities

[1] These activities were created using Park et al.'s (2023) tasks published through the MTLT journal.

Changes in PSTs' Confidence in Using Programming in Mathematics Classrooms with ChatGPT's Assistance: Ellie's Case

In this section, we will use Ellie's case to describe the positive impact of these PGL activities in increasing teachers' teaching efficacy. Before taking the geometry content course, Ellie had prior experience with programming in introductory-level college CS classes. After completing the programming-embedded activities for almost five weeks from the beginning of the semester, when asked to reflect on her thoughts on programming integration in teaching mathematics, she expressed skepticism about that idea, rating her confidence level in teaching programming as follows: "My confidence level in teaching a block-based program would be around a 2 (out of 3 points), but *my confidence level for a text-based program would be more around a 1*." However, we noticed that her overall confidence rating evolved after completing the task design activities with ChatGPT assistance. In her task design process, Ellie even showed flexibility in programming choice and picked a programming application that she thought would be relevant to use as a tool for supporting students' learning of mathematics better, taking into consideration the target mathematics topics she would teach. She shared in her reflection after completing the task design and implementation activities:

> I wanted to create a task about transformations on geometric shapes, so I thought I would use block-based programming. However, *I decided to change my topic to focus on different angles and the Alternate Angles Theorem. I then decided to use Python because I figured that would be the best programming option for creating lines and angles using Turtle*.

Her overall experience with this task design project was positive, and she reflected, "ChatGPT was very helpful for me in creating my task. I have a background in programming, but *I don't think I would have been able to efficiently create a program to demonstrate what I wanted it to do*." She also expressed confidence in integrating programming, saying:

> I would say my confidence level for teaching programming in future math classrooms would be about a 2 (out of 3 points). *I feel confident about my knowledge in programming*, I think the only thing holding me back from being a 3 is that *I'm still not super familiar with teaching because I haven't done any practicums*.

PSTs' Confidence in Using Programming and Their Choices of the Programming Language and ChatGPT Usage

Ellie's case illustrates how ChatGPT can improve teacher confidence in using programming in the mathematics classroom. Like Ellie, Grace also found ChatGPT could complement her programming skills, so she used it to generate text-based code for her geometry learning task. After the task design and implementation activities, she reflected on how using ChatGPT to write her code made her life less stressful. Grace wrote that

> Writing code from scratch can be difficult, but if I see an example of what it is supposed to look like, it makes it easier for me to understand. *By having ChatGPT write the code, I can spend more time on my given topic and less time trying to figure out how to write the code*.

So, for Ellie and Grace, ChatGPT was a useful tool for creating text-based programs "effectively," which they wanted to use in their mathematics lessons.

However, two other PSTs decided not to rely on ChatGPT's assistance to create a program(s) for their programming-embedded geometry learning tasks. We noticed that their choice was related to their confidence in their programming knowledge and abilities to create programs. For instance, Lynn found that block-based programming worked well for her task, which was the programming environment she felt confident using. Therefore, Lynn felt no need for ChatGPT assistance, so she chose not to use it in her task design process, making block-based Scratch programs by herself. Charlie also started with high confidence in her programming skills and knowledge. She expressed "quite" confidence (3 out of 3 points) in making blocks and text-based programs. Her programming choice for her task design was text-based programming with Python Turtle graphics. So, like Lynn, Charlie also decided not to rely on ChatGPT to generate the code in her task design. However, in her written reflection after completing the task design, she shared, "It would have been useful to ask it [ChatGPT] for the most simplistic code to start, plus a reusable looping code."

Suggestions on Supporting Teachers to Incorporate Programming in Teaching Mathematics

We were able to see the potential use of ChatGPT in reducing teachers' burdens of creating programs by themselves, which may increase teachers' openness to incorporating programming in their mathematics classrooms. However, for effective use of programming in mathematics classrooms, it is necessary to provide learning opportunities like the PGL activities through which teachers can build their knowledge about interpreting, revising, designing, and evaluating programs in mathematics learning contexts, enabling them to see how programming can support mathematics learning or vice versa. Without this knowledge development support, teachers could not use ChatGPT-generated code effectively for programming-integrated mathematics teaching.

References

Association of Mathematics Teacher Educators. (2022). *Position of the association of mathematics teacher educators on technology.* Retrieved from https://amte.net/sites/amte.net/files/AMTE%20Technology%20Statement%20Oct%202022_0.pdf.

Celepkolu, M., O'Halloran, E., & Boyer, K. E. (2020). Upper elementary and middle grade teachers' perceptions, concerns, and goals for integrating CS into classrooms. In *Proceedings of the 51st ACM technical symposium on computer science education* (pp. 965–970). https://doi.org/10.1145/3328778.3366937

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., ... & Zaremba, W. (2021). Evaluating large language models trained on code. https://doi.org/10.48550/arXiv.2107.03374

Kim, C., Gleasman, C., Boz, T., Park, H., & Foutz, T. (2022). Learning to teach coding through argumentation. *Computers and Education Open*, *3*, 100–107. https://doi.org/10.1016/j.caeo.2022.100107

Park, H., Boz, B., Sawyer, A., & Willingham, J. C. (2023). Triangle explorations and constructions using robots. *Mathematics Teacher: Learning and Teaching PK-12*, *116*(5), 392–398. https://doi.org/10.5951/MTLT.2021.0313

Park, H., & Manley, E. D. (2024). Programming-integrated mathematics learning for future elementary teachers and non-STEM majors. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (pp. 1014–1020). https://doi.org/10.1145/3626252.3630908

Stigberg, H., & Stigberg, S. (2020). Teaching programming and mathematics in practice: A case study from a Swedish primary school. *Policy futures in education*, *18*(4), 483–496. https://doi.org/10.1177/1478210319894785

Vivian, R., Quille, K., McGill, M. M., Falkner, K., Sentance, S., Barksdale, S., ... & Maiorana, F. (2020). An international pilot study of k-12 teachers' computer science self-esteem. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education* (pp. 117–123). https://doi.org/10.1145/3341525.3387418