

# Viewing Learning Through a New Lens: Game Programming with Python Through Meno

Kadyn Dienst

December 2, 2024

As Meno, I would find myself questioning not only the process of learning but also my ability to tackle the challenges of college. Much like Socrates challenged Meno to think deeply about what it means to “know,” my year one course, Game Programming with Python, has caused me to reflect on my preparedness, motivation, and understanding of how knowledge is gained. These reflections align closely with the theme of COR 100: to gain a understanding of college and the skills needed to succeed.

When I began, I felt uncertain about my ability to succeed in college. While I was motivated by a passion for problem-solving, programming was a unique challenge that tested my perseverance. Early on, I encountered setbacks like syntax errors and things such as loops. These moments were frustrating but also showed potential for growth. Every small success, like completing a number-guessing game, showed me that I could rise to the demands of college through determination and practice.

I realized that motivation by itself wasn't enough; preparation and structure were equally important. I needed to embrace the process of trial and error, a lesson that resonated with Socrates' approach to questioning in Meno. Just as Socrates encouraged Meno to question his assumptions, I learned to confront my uncertainties in programming, using them as opportunities to build skill.

The problem of "how we know what we know" is central to both Meno and my journey in game programming. Initially, programming felt like an insurmountable puzzle. I didn't know what I didn't know—a paradox Meno himself grapples with when he asks Socrates, "How will you look for something when you do not know at all what it is?" (Plato, trans. 1994). This paradox mirrors the learning process in programming, where the path to understanding is unclear until you take the first step.

My breakthrough moments came when abstract concepts became concrete through practice. For example, the concept of loops was confusing in theory but became clear when I used them to create a scoring system in a game. This hands-on experience helped me "know what I knew" in a way that felt both practical and philosophical. It also highlighted the importance of reflection and iteration in deepening understanding—lessons that apply not only to programming but also to the broader process of learning itself.

One thing from Meno that resonated with me was Socrates' assertion that "searching and learning are, as a whole, recollection". This idea suggests that learning is not about gaining new knowledge but about uncovering what we already know. In programming, this became clear when I realized that many solutions came not from memorizing syntax but from applying logical thinking skills that I had before this course.

Debugging, for me, became a good example of recollection. Each error message felt like a question, challenging me to reconsider my approach. With each correction, I wasn't just fixing mistakes; I was rediscovering patterns of logic and problem-solving I had used in other areas of life. This process reinforced the idea that learning is less about filling gaps and more about connecting what we already know to new contexts.

Reflecting on my journey through Game Programming with Python, I see how it has shaped my academic preparedness, challenged my assumptions about learning, and deepened my understanding of how we come to "know." The course mirrored the themes of COR 100 by encouraging me to engage with the paradoxes of knowledge and embrace the process of discovery. Just as Meno found himself questioning and learning through Socratic dialogue, I have grown through the challenges and triumphs of programming. While I have a lot left to learn, I am motivated by the fact that knowledge is not finite but a lifelong process of growth.