My teaching philosophy centers on the idea that students are active learners who need to be engaged with the material to truly understand and retain it, and that we as computer science teachers must work to improve teaching in the field. I work to facilitate and guide the learning process by creating a supportive and inclusive environment that encourages student engagement and participation. I believe that students learn best when they are actively involved in the learning process and can see the relevance of what they are learning to their careers and goals.

In my career as a computer scientist, I have seen first-hand how students learn the minimum to complete an assignment and do not end up retaining much information. Therefore, my primary objective for the course when redesigning Rapid Prototyping was for the students to be able to laser-cut and 3D print five years from now with confidence in their ability to both use the tools and design and create prototypes using those tools. We taught them how to laser cut and 3D print for the purpose of designing prototypes and simulating motion, along with using more advanced tools like Rhino, Kangaroo 2, and Grasshopper. I chose to design the course with backwards integration, building backwards from what I wanted the students to learn.

To achieve this goal, I incorporated interactivity into lectures, hands-on activities, and real-world examples. In my second lecture in Rapid Prototyping, I taught the students about four-bar linkages, and showed them numerous examples of how gears are used in the real world (like prosthetic knee joints, the pantograph, and cars). They were then tasked with building four bar linkages on a simple automaton in SolidWorks, which we walked through together to learn mechanical mates and rotational motion in SolidWorks. Several lectures later, when I asked the students a question in class about whether they could think of examples of complex linkages, students readily remembered details of four-bar linkages because they were able to connect it to real-world uses. These in-class questions reinforced their knowledge and gave me a sense of what the students needed help with.

At the conclusion of class, I had the students respond to overarching questions regarding what they were struggling with, what they felt confident about, and what they were eager to learn. My aim was to gauge their overall confidence and reinforce the idea that they were making progress in mastering certain SolidWorks concepts, activating their prior knowledge.

Another objective I had for the class was for students to be able to understand how they can use these skills of variable usage, flexible code, and object-oriented programs in the real world. The entire course is hands-on; each class, the students build something new. This flipped classroom approach allows them to continue building the muscles of parametric and object-oriented programs in an interactive way that not only encourages skill retention through repetition but also builds confidence. When I taught my first lecture on Design Tables in SolidWorks, I had students build a plate with holes that varied based on the variables and parameters they set. I connected this to other computer science classes they learned and concepts they already knew like variable-setting for ease of reproducibility and automation.

To connect concepts to the real world, I created worksheets and discussion questions for lectures. I had the students practice the steps of iterating through design and prototyping steps, so that they could use the skills they were learning in the real world. The worksheets outlined an ill-defined problem, such as designing a Voronoi-holed vase, and a list of questions that an expert would ask him/herself to approach the problem, such as "What are the key design considerations to build this in Grasshopper?" and "How can these considerations be balanced to create an effective design?" This allowed the students to exercise the skills they would need in their future careers. Because the professor told me that in prior years students often took the class with friends or helped those sitting next to them, I had students work in pairs or groups so they could efficiently work kinks out together.

A third objective I have is to help students apply algorithmic, computational, and logical problem-solving skills to the challenges they encounter in computer science. Through the courses I teach, I want to guide students toward acquiring and honing professional and personal skills of programming and design iteration by thinking critically and creatively.

In computer science, students are often discouraged when they get stuck on difficult and new concepts. Therefore, I endeavor to cultivate a growth mindset through problem-solving in my students, which inspires them to be persistent and resilient when faced with obstacles. I seek to do so using collaborative techniques. I do this because many CS courses don't explicitly allow for collaboration, but in practice, and in the real-world, collaboration is key to problem solving.

To implement this teaching philosophy and the learning objective of building resilience, I use a variety of teaching methods such as project-based learning, flipped classrooms, and peer instruction. In my computer science courses, I emphasize the importance of teamwork and collaboration by assigning group projects and encouraging peer review and feedback.

I changed a few lectures to be student-led: students each chose a new 3d printing-related technology to present, which they were graded on as a new segment of their course grade. They found this fun and were able to deep-dive on interesting technologies of their choice, ranging from 3D-printed skin to pasta shapes; this helped them build confidence with the topic.

When I teach, students have frequent opportunities to 'turn and talk,' making sure that their neighbors and themselves are on the same page and helping one another solve any problems they encountered. This allows them to identify common problems in a program, even if they themselves don't have the problem. It also allows them to develop problem-solving skills without asking instructors, which they were prone to do at the beginning of the semester.

I also believe in using iterative, multi-part assessments that are aligned with the course objectives and provide students with meaningful feedback to improve their learning. Aside from peer feedback that happens in real-time during the flipped classroom, I allow for numerous check-ins with myself so that I can ensure that everyone understands the subject matter. During my lectures, I had 30-second student check-ins where I had students ask questions, walked around their computers to see that they had all created the same objects, and let those lagging catch up. This gave me real-time feedback on what the students needed more time on, what they were stuck on, and clarifications that were needed. For example, a student typed in an equation incorrectly and needed help debugging; I found her bug, an extra parenthesis, and this question allowed other students to find similar bugs in their own code.

At the end of class I also sought to identify areas of confusion to address them in future lectures. By asking about their interests, I hoped to foster their engagement and enthusiasm for the subject matter. This approach allowed me to gather a more comprehensive understanding of how the students were comprehending the material, beyond what I observed during class. For example, one question I asked was, 'What can you do now that you couldn't do before?' This promotes metacognition and allows me to see in which areas students are becoming more confident.

My teaching philosophy is centered around creating an inclusive and collaborative learning environment that encourages students to think critically and develop problem-solving skills. I believe that the most effective way to teach computer science is through active learning, where students are encouraged to apply theoretical concepts to real-world problems and work collaboratively to find solutions. I also believe in the importance of providing students with personalized feedback and support, so that they can develop their skills and confidence in a safe and supportive environment. I am committed to continuous growth and development as a teacher, and I look forward to working with and learning from my students.