

Teaching Philosophy

I am passionate about the computer sciences and strive to foster an inclusive learning environment where I can share my excitement for challenging ideas and problems, while being respectful of, and responsive to, student needs. While I do not think every individual *must* become a computing professional, computation is increasingly a part of every field; therefore, I strive to ensure that my students feel that they *belong* & are supported in this field, so that they don't abandon this challenging & rewarding pursuit for the *wrong* reasons. I utilize a number of tactics in pursuit of these objectives:

I establish context and communicate my passion. Students that view course material as relevant and interesting are more likely to ask clarifying questions, think deeply about the implications of concepts, and explore material & applications beyond the classroom. I demonstrate relevancy by connecting course material to current events and industrial/research applications. It has also been my experience that passion for a subject is infectious, and that students, in part, engage course material because a teacher exhibits excitement and interest. Thus I often share relevant problems and issues from my own experience, which includes academic/industrial research and entrepreneurial pursuits. I find that engaging in discussion of real-world problems, especially those in which I am personally invested, reinforces course material and encourages students to think about how they will apply computing to pursue their own goals and have an impact on the world around them. Finally, since student success is a partnership, I find that students are more willing to challenge themselves when they see my own level of investment in their success. So to better engage my students, I am often open about my own commitment to the incredibly challenging, impactful, and fun practice of effective teaching.

I encourage a balance of individual exploration and active facilitation. One of the distinctive characteristics of the computer sciences is the relatively low level of resources and expertise needed to experiment with new technologies, contribute to real-world projects, and learn from practitioners. Thus, I actively encourage students to quickly prototype their ideas, contribute to software projects they value, and seek guidance from peers and relevant experts. However, to be lifelong learners and collaborative practitioners in this rich field, my students must also be skillful as analytical & design thinkers, iteratively engaging with complex ideas & methods. It has been my experience that exploration facilitates learning motivation, breadth, and speed, while deep reflection & consideration promotes understanding, growth, and quality of outcome. Thus I regularly walk a fine line, deftly wielding instructional scaffolding and other resources to facilitate individualized & experiential student learning.

I communicate clear expectations and provide frequent feedback. Students feel motivated & respected when they understand the goals of a class, can visualize their path to achievement, and are provided clear sources of useful feedback & support along the way. I invest time and attention to clearly and consistently communicate my expectations involving learning objectives, assignment quality, class participation, and a schedule of deadlines (that, whenever possible, I adhere to without modification). Setting a clear and challenging bar motivates students to focus their energy on mastering material, as opposed to juggling administrative and/or guessing instructor priorities. I regularly begin sessions with an overview of upcoming tasks, responses to common student questions, and a short-to-long-term contextual map of topics to come. In my experience, once we address these pragmatic sources of anxiety, the class as a whole can focus and engage the material of the day. I also strive to provide diverse, frequent sources of feedback, such as clear answers to questions during lecture/online forums, innovative forms of automated (if not partial) grading, one-on-one discussion during office hours/electronic communication, and fast, but informative, turnaround in assessment. Quickly and effectively closing the feedback loop catches conceptual inaccuracies early and helps to make students feel supported in their learning journey.

I adapt to student needs. Students come to a classroom with a variety of needs that stem from diverse backgrounds/interests, distinct understanding of prior material, and differing learning styles. I continually adapt course material/presentation and classroom management in an attempt to create an optimal environment for each student and the class as a whole. For example, to develop a sense of belonging for each member of the class (and instruction team), I craft lecture language & assessment problems to highlight historically underrepresented groups (e.g., using gender-balanced pronouns, or non-binary when possible; using names that are not all coming from a single cultural context, such as white American/European). To engage students, I enhance traditional lecturing with active learning techniques, such as Think-Pair-Share and role-play (e.g. physically acting out algorithms). I also strive to develop a supportive classroom environment by incorporating collaborative small-group activities and both encouraging and rewarding student involvement. To touch upon a variety of learning styles, I seek out and share a variety of resources pertinent to course topics, including online videos/talks/lessons, academic papers, as well as open-source projects. Additionally, I have found that many students benefit most from small-group and/or one-on-one interaction during discussions/labs, as well as office hours. These personalized sessions are some of the most challenging and rewarding for me as an educator, and I strive to ensure that no student walks away without added comprehension of, and appreciation for, classroom material, as well as the computer sciences as an intriguing and rewarding pursuit.