My objective of teaching and mentoring is (a) to teach students to fish rather than give them a fish, and (b) to inspire students to pursue their professional career, by (1) developing students' internal skills (i.e., critical thinking and fundamental concepts), (2) emphasizing the importance of what, which, and when, in addition to how and why, and (3) sharing encouraging true stories and the vision of promising research directions, by means of lecturing, mentoring, and writing.

In 2003, I served as a teaching assistant for an Evolutionary Computation course in Taiwan. From 2006 to 2008, I served as the teaching assistant for E-commerce technologies for four semesters at NC State in US, where I received outstanding teaching assistant award. There are several things I have learned from these experiences.

I believe the *internal skills* (e.g., in computer science, fundamental concepts such as formal languages and object-oriented concepts) are more important than the *external skills* (e.g., techniques and tools such as linear regression, A* search, and C++). In Chinese martial arts, internal skills refer to the development of coordination of human bodies in order to excel in mechanics such as blood circulation, muscular movement, and balance. External skills, on the other hand, refer to any particular categories of martial arts. Internal skill practitioners are known to be good not only at bringing their external skills to bear but also at picking up new martial art category quickly. Computer science embraces tons of external skills. These external skills advance quickly. In addition to teaching the latest external skills, I would emphasize the corresponding internal skills. For example, besides syntax, I would emphasize object-oriented concepts when teaching Java. Besides various learning algorithms, I would emphasize the general idea of statistical learning when teaching machine learning. I believe, similar to martial arts, developing such fundamental internal skills enables students to excel at the external skills they have known, and also to pick up or even design new skills quickly.

Understanding *what* the purpose of a technique is, and *when* and *which* technique should be chosen is as important as knowing *how* and *why* it works. The former enables students to apply the technique wisely, whereas the latter enables students to implement it. Computer science is a comprehensive and fast-developing discipline. Students keep learning new techniques and forgetting those they have learned. It is almost impossible for students to remember all the details of how and why. Yet, compared to memorizing how and why, it is easier for students to understand what, which, and when. As a teacher, I would highlight the pros and cons of the techniques, identify the differences between similar techniques, and point out which technique is preferred in what situation. For example, in addition to how a red-black tree works, I would emphasize how a red-black tree is different from other tree structures, how it is compared against B-trees and binary search trees, and when to apply red-black trees rather than the others. In homework assignments and exams, rather than having students to go through step-by-step details of the techniques, I would provide realistic problems, ask students to choose appropriate techniques, and explain their choices. I believe understanding what, which, and when is as important as knowing how and why. The latter is essential for implementation, but without understanding the former, students would not be able to apply what they have learned wisely.

I believe students need inspiration as much as knowledge. People do things well not only because they are good at those things, but also because they are passionate about those things. From my teaching and mentoring experience, students' passion usually comes from the inspiration of people around them. Teaching can be inspiring. For example, when teaching artificial intelligence, I would share encouraging stories of IBM Watson, talk about fascinating applications of facial recognition, and describe a promising vision of future intelligent world. Besides these stories and future plans, I would also provide a concrete picture of current state of the art, and point out practical research opportunities that can fulfill these ambitions.

Another lesson I have learned is the difference of learning styles between American and Asian students. In US, it is very common to see the interactions between teachers, teaching assistants, and students either during class and office hours, or on discussion boards. A teacher can easily evaluate the learning process of the students based on their explicit feedback. By contrast, in Asian countries such as Taiwan, most students are either shy or afraid of asking questions. When the students struggle with the materials, they tend to seek assistance from classmates and friends rather than from their teachers, possibly because of the social barrier between professors and students. The learning process of these students is hard to evaluate without exams and homework assignments. Too many assignments and exams may put too much pressure on the students and give them an incentive to plagiarize. Also, sometimes both American and Asian students may be under the wrong impression that they understand the materials. Thus, as a teacher, I would design nongraded quizzes depending on how much ideas of students' learning progress I have. These quizzes are designed for the students to make mistakes, and help both the teachers and students evaluate what have been learned.

There are also other important teaching philosophies that I believe including frequent adjustments according to student needs and feedback, and balance between individual work and teamwork. Teaching is a great opportunity of learning. I would be proud of teaching and mentoring students and seeing them to approach their successful careers.

For teachable courses, I would like to design a multiagent coordination course that focuses on the coordination using trust and reputation mechanisms in distributed systems. I am interested in teaching courses including multiagent systems, service-oriented computing, artificial intelligence, data mining, and machine learning. I can also teach basic classes such as probability and statistics, data structure, algorithms, graph theory, and discrete mathematics.