

# Teaching Statement

## Yi Lin

I believe that teaching is essential part of a professor's career. I derive my teaching philosophy from my experience as a student, teaching assistant, and lecturer at McGill University.

### 1. Experience

During my graduate studies, I have served as teaching assistant for several courses in different subject areas, including programming languages and techniques (comp202, 302), database systems (comp421), and distributed information systems (comp577, 614, 764). This experience has helped me deepening my knowledge in these areas. By assisting individual students in their assignments and projects, I gained experience in what the common challenges are for most students acquiring knowledge.

I was also invited to give guest lectures in some graduate courses given by other professors on topics related to my research. These include group communication and ordered multicast, peer-to-peer data management, transactions, recovery, caching, and replication. I really appreciate the opportunities to discuss with the graduate students my vision and understanding of these topics. Sometimes I got interesting ideas for my research from the discussions.

As a graduate student, I have been fortunate enough to be offered a lecturer position for an undergraduate course, Computer in Engineering (comp208). I have taught this course twice (Fall 2005 and Fall 2006). The course is intended for students who have diversified backgrounds in engineering (such as mechanical, civil, and chemical engineering) and are interested in scientific computation. The number of students is around 100. In the course, I taught students the concepts and structures of high-level programming, some non-numerical algorithms (such as sorting and searching), some numerical algorithms (such as root finding, numerical integration, and differentiation equations). Teaching this course gave me lots of experience in how to teach an introductory course in computer science. I also learned how to teach large classes which is different from how to tutor students one-to-one or how to give a lecture in a graduate course with only a few students.

### 2. Personal philosophy

My teaching philosophy is to place myself into the position of the student. It is important for me to understand how students learn, and to be aware of the possible difficulties when one is confronted with the material for the first time. My goals of teaching can be summarized as: Firstly, the student should have learned how to solve the basic problem in question with the basic techniques taught in the course. Secondly, the student should have gained an overview of the subjects covered by the course. Thirdly, the student should have gained enough methodology to be able to learn advanced topics independently if necessary. I am a believer of an old Chinese saying: *Giving a man a fish, you feed him for one day. Teaching him how to fish, you feed him for his whole life.*

In the classes I taught I have always tried to meet the goals as mentioned above. For example, in the course Computer in Engineering, I spent considerable time (half a semester) to introduce them to write programs in one language, Fortran, which fits into the students' engineering background. After that, I just spent one week to introduce basic programming in C. I explained to the students the differences between Fortran and C, and pointed out common pitfalls. It turns out that, as long

as students understand how to program in Fortran, it is not difficult for them to transit from Fortran to C. Following this, I also showed students how to implement numerical algorithms in C and how to apply programming to real engineering problems by examples. This requires sophisticated programming skills which might not be understood quickly by students. But students like this because they are convinced that computer programming does matter to their future professional activities.

I also developed my own way to teach students how to learn programming from scratch. In Fall 2005 when I began to teach this course, I used PowerPoint slides to show them how to write a program in proper syntax. However, most students just got lost even with a Hello World program except for those who had some programming experiences before. A lot of things that seem trivial to me, are hard for them to understand. After the first several classes, I decided to show them “live” how to type code, how to compile, and how to debug by directly typing the programs on my computer and show them via the projector. Students named my approach as live programming. Normally in each lecture I began my live programming with one simple example. Then I expanded the example by introducing more and more syntax. My slides were only used for the review purpose by students. All students loved my live programming very much. They told me that they could not link syntax in my slides to programming until I convinced them with live programming.

In my classes, I encourage students to participate actively. For example, I encourage them to ask questions. According to my years’ experience as a student, I think many students refrain from asking questions because they are afraid of appearing stupid in front of their peers. Hence, sometimes I ask some very easy questions on purpose to build their confidence. I always highly comment their questions as “good questions” as long as that would not abuse the words.

I really enjoy seeing that students in my classes are very willing to learn. When my lectures went well, students asked the right questions at the right time. This really stimulates the desire of my teaching and I feel rewarded.

### **3. Teaching interest**

Based on my research and previous experience, I would be interested to teach undergraduate courses in introduction to programming, object oriented programming, data structures, databases, and distributed systems. I would also be happy to offer programming courses to engineering students as I did at McGill University. I can also teach courses in discrete mathematics and operating system. At the graduate level, I would naturally be interested in teaching courses that best suite my research interests. Thus, my first step would be to develop a course on distributed data management.