

Teaching Statement
C. Paul Cook

I believe the best way to engage students is through enthusiasm and passion for the subject at hand. By getting students excited about what they are learning, putting forth the effort necessary to grasp complex ideas will come naturally to them. Indeed the teachers who have made the strongest impression on me as a student have exhibited genuine excitement for their subject of teaching; when I teach, I strive to similarly connect with my students through my own enthusiasm for computer science.

This is an especially exciting time for computer science education, particularly at the introductory level. The ability to easily obtain massive amounts of data is changing the nature of inquiry in many fields across both the sciences and liberal arts, such as astronomy, linguistics, and sociology. Members of fields outside of computer science can increasingly benefit from an understanding of computer science and computer programming in order to cope with such vast and ever-expanding quantities of data. More and more, programming skills are required to effectively analyze data; however, and perhaps more crucial for researchers outside of computer science, an appreciation for the principles of computation, and in particular, what can be easily computed, is necessary to be able to formulate appropriate research questions. By promoting computer science and effectively teaching to beginning students, I believe computer science educators can have a tremendous impact on human knowledge.

Because I believe that even a limited experience in computer science can benefit a student's understanding of other fields, I am most passionate about teaching students who are new to computer science and programming. Moreover, I am especially interested in teaching those who are established in other fields, and do not view themselves as "computer scientists". This is a reflection of my research interests, which bring together ideas from computer science, linguistics, and lexicography.

My passion for teaching beginners is demonstrated by the courses I have chosen to be a teaching assistant (TA) for as a graduate student at the University of Toronto (UT). In particular, I have TAed numerous sections of UT's introductory computer science course, Introduction to Computer Programming. I have also TAed sections of several other courses for beginners, including Computer Science for the Sciences, an introductory course intended for students outside of computer science; The Why and How of Computing, a course for those with little experience with computers; and Mathematical Expression and Reasoning for Computer Science, an introductory course on discrete math and logic. My primary responsibilities as a TA have been holding tutorials for sections of 20–30 students, and working one-on-one with students in computer programming labs. In addition, I have been asked to give guest lectures

when the instructor was not available, and to act as a head TA to co-ordinate the activities of other TAs. I have also graded many assignments and exams. The effectiveness of my teaching is shown in my most recent teaching evaluation where students consistently rated my teaching as “very good”. Nevertheless, I recognize that as I am teaching others, I too am learning to teach, and I am therefore committed to improving my teaching. As evidence of this, I completed the Teaching Assistant Training Program Certificate at the University of Toronto, which involved attending a series of pedagogical seminars and having my teaching evaluated.

I am qualified to teach a variety of computer science courses, including all those typical of the first two years of an undergraduate computer science degree. I am also well-suited to teaching upper-level undergraduate or graduate courses on computational linguistics and natural language processing. Furthermore, I would be very interested in developing new courses. Below are two ideas for courses that I believe might be of interest to many students outside of computer science.

Computer Science for Non-Computer Scientists:

This course would be intended for upper-year undergraduate students with majors in data-driven fields outside of computer science. It would present principles of computer science, similar to those in an introductory computer science course, but cater to more mature students. Examples for assignments and projects would illustrate practical applications of the concepts being learnt for tasks related to data analysis.

Text Processing for the Humanities and Social Sciences:

This course would be intended for upper-year undergraduate students with majors in areas in the humanities and social sciences that make use of linguistic data. The focus of this course would be on practical skills for processing and analyzing text. The assignments would help students develop the skills necessary to ask and answer their own questions about how language is used.