Seth Poulsen
sethpoulsen.github.io
sethp3@illinois.edu |(559) 999-1482

Teaching Statement

I have always been thrilled by the joy of learning new things, and I love the opportunity to help others experience the same thing. The goal of a computer science program is that students enter as young adults and leave as competent professionals, equipped to use their talents and skills to have a positive impact on the world. As an assistant professor I look forward to the opportunity of being involved with this transformative process.

My teaching experiences so far have been:

- one semester teaching discussion sections twice a week as the only TA for a calculus class of 15-20 students
- two semesters as a TA for a discrete mathematics course with over 1000 students, working with students solving problems in class, and developing course materials (including Proof Blocks, the subject of my dissertation)
- one semester being the lead TA for a CS1 lab where 30-40 students practiced their introductory programming skills
- mentoring four undergraduate and post-baccalaureate students on research projects

In this statement I will elaborate on how I use evidence-based practices to build inclusive classrooms where every student feels valued and supported.

Teaching Philosophy

I want every student who enters my classroom to feel that I care about them and that the structure and organization of the course will help them to meet their learning goals. To accomplish this I use evidence-based practices in my teaching that help all students perform better, especially students who identify with groups that are underrepresented in computing classrooms.

Set Clear Expectation, Help Students Reach them Unclear expectations is a major source of frustration for students. This negative impact can be worse for students who are members of underrepresented groups in computing and already experiencing additional stressors. In my courses, I strive for students to have clear expectations on what they are supposed to learn and accomplish, and that they know I will support them in reaching those expectations. I will do my best to ensure that my courses have a clear syllabus that shows the learning objectives, the assignments that students will be given, and how they will be graded.

When I was a teaching assistant for discrete mathematics, an ongoing source of frustration for students was their surprise at the reasons that they lost points on the mathematical proofs that they wrote on their quizzes. Each week when preparing to grade the students weekly quizzes, the teaching assistants and instructors would collaboratively construct a rubric. I realized that all of the rubrics followed the same general structure and by giving students more insight into the process of how the rubrics were constructed, we could help them know how they should be studying, and increase their ability to write good proofs.

After reviewing the existing rubrics, I put together what I called the 'generalized proof rubric.' The generalized proof rubric explained the kinds of points all proof rubrics have (e.g. variables are defined before being used, definition of mathematical property is correctly applied in context of the

proof, etc.) and gave examples of the points and how each may be done correctly or incorrectly. I sent it to all the students in the course so that they could have a more clear idea of how their proofs should look.

Active learning Students remember some things from listening to a lecture, but they truly learn by doing. Active learning methods helps all students, but especially those from underrepresented groups in computing, helping to close the achievement gap. In the classrooms I lead, I seek to create an environment where students are given the opportunity to try hard things and solve new types of problems—along with a culture where students know that it is ok to not know the answer, or to be wrong. Of course, students also need some amount of direct instruction of material, and so my classes often involve alternating between giving students instructions, and having them work together on problems in groups.

In the theory-based courses I've taught, I alternate between showing an example of a problem that I solve for everyone to see, and then letting the students work together in groups on similar problems. While they work, I monitor student groups to help them have productive interactions, make sure that all students have an opportunity to contribute, and help students see the next step when they get stuck. Similarly when I teach programming courses I will show students examples, and have students work together on problems.

When I taught the recitation session for a calculus class at Brigham Young University, almost none of the courses in the department used any sort of active learning or in-class group work activities. While students resisted at first, by engaging the students in working on problems in class, I was able to help them have a positive experience working together in groups. The experience taught me that while it takes some amount of work to transform a course from having no active learning materials to getting students to work together effectively, it is worth the effort.

Inclusive Group Formation When working on groups in class activities, in a group with multiple men and only one women, the men may talk over the woman and not give her equal opportunity to contribute. This phenomenon can also happen to people from underrepresented ethnic groups in computing. Thus, when forming groups for in-class activities, I will try to use group formation methods that help all students have the opportunity to participate and learn.

In addition to these precautions, I reach out and ask my students how their groups are working for them, so that they have the opportunity to express concerns that they have. When I was a teaching assistant for discrete mathematics at the University of Illinois, students were assigned groups at the beginning of the semester, with the intention of keeping them in the same groups for the duration of the semester. One of my students was very excited about working with other students in his group, but he came from an ethnic group underrepresented in computing (Hispanic) and had a different background and experiences than the other students in his assigned group. He tried to engage students in his group, but his group members did not respond well (this was exacerbated by the nature of online course discussions during the COVID-19 pandemic). After talking with him individually, I was able to get him moved to a different group. I gave all of my students the opportunity to express concerns about their groups, and I was also able to make other changes to the groups that helped productivity for all of the groups involved. After having such a positive experience in shifting students around to improve group composition, I helped my peers who were teaching assistants for other sections of the course to do the same.

Reaching Students Individually I want all my students to know I care about them, and want them to succeed. When I first meet a new group of students and introduce myself to them, I communicate

that I care about them and their success. When teaching Calculus, I was able to use information I gathered from grading student assignments to know what material students needed further exposure to in order to succeed. In this way, I was able to give individualized help to student.

Over my years as a teaching assistant, I have had multiple students come to me with personal issues that were affecting their performance in the course. I feel so grateful that I was able to create an environment where students were comfortable coming to me for help, and that I was able to take an opportunity to help them as individuals.

Continuously Improving Some of the best instructors I've had were so good because they were constantly responsive to the feedback that they received from students, whether that be changing an assignment when realizing that it placed an unnecessary burden on some students, or altering course policies in response to new understanding of how they affect students. They also continuously monitor course data, taking note if students from certain demographics struggle with certain parts of the course, so that they can take action to remedy the situation. Some of my instructors had an anonymous form open throughout the semester, so that any student could submit anonymous feedback to them at any time, a practice that I will replicate in my own courses. Even in semesters when I never made use of the form, the fact that it existed helped me to feel more comfortable in the course because I knew that the instructor cared about their students and wanted to listen.

Research Mentoring

I mentored four students on research projects during my PhD, resulting in two papers where my research student was the first author. With each of my students, I have helped them leverage their talents and skills to make meaningful contributions in research, while also guiding them through acquiring new skills. Two of my students were international students, one a woman, and one a non-traditional student in the University of Illinois' iCAN computing accelerator program, for students with a degree in a non-computing discipline to transition into computing.

I keep an active list of research projects that students I advise can take ownership of. Different projects on my list require different skills, ranging from educational data analysis, designing user interfaces for learning software, integrating software into existing learning systems, and reading and extracting information from education research papers. Thus, any student I mentor can both contribute in a meaningful way with their skills and interests, as well as have the opportunity to learn new skills. Having a variety of projects for students to pick from also helps students take ownership and feel excited and motivated by their projects. Here I will give examples of a few of the positive experiences I had mentoring students.

Two of my students, Ridha and Mei, conducted data analysis about they types of mistakes that students make while writing database queries in the MongoDB or Cypher query languages. Ridha took ownership of the MongoDB analysis and Mei took ownership of the Cypher query language analysis. As they worked on their projects, I met with them individually and as a group each week to help them with their data analysis code, and discuss the best way to classify student errors in the data set. Under my direction, they each wrote up their work as the lead author, and both presented it at ITiCSE, an ACM conference on computer science education. Mei's work also won 2nd place in the student research competition at SIGCSE. In addition to mentoring Ridha and Mei on their research, I was able to help each of them contemplate the next step of their career, and both have since entered graduate school in computing. I feel honored to have had an impact on them in their career development, and I am excited to continue having this sort of impact on students throughout my career.

One of my other students, Jackson, was part of the University of Illinois' iCAN program. As such, Jackson was a non-traditional student who had spend the start of his career as a high school teacher. Jackson was able to use his prior experience to help in writing educational materials for a study we were planning, and at the same time I helped him get his programming skills up to speed. Later, Jackson personally thanked me for teaching him HTML as it had helped him contribute meaningfully to his group projects in class. Soon after, Jackson started an internship and then a full time job working as a software developer.

Conclusion

Teaching, mentoring, and watching students grow has been such a joy for me. Over the last few years I've learned many methods for helping students of all backgrounds succeed in courses and in research, and I will continue to learn more. I look forward to continuing to grow in my teaching and mentorship experiences as an assistant professor.