Teaching Statement

I strive to engage, challenge, and inspire growth in my students. My passion for computer science motivated me to pursue my doctoral education to innovate and explore new ideas, which in turn helped me to improve my teaching and mentoring skills. I believe that every individual with proper mentorship can achieve the goals that might initially seem unachievable. As a young girl interested in STEM, I had many mentors and role models that inspired me to pursue a career in academics and explore my passion for computer science. I want my students to share the same passion and become valuable collaborators in the exploration of theory, research, and practice of computer science. Most students I have taught are now either engineers or academicians, and many of them are still in touch with me. The fact that I am able to contribute back to society, giving knowledge to future generations, gives me immense gratification.

1. Teaching Experience

I have taught undergraduate and graduate students over the past 17 years as (i) Assistant Professor at the University of Tulsa (TU); (ii) Teaching Assistant at the University of Nebraska-Lincoln (iii) Senior Lecturer at the Baba Banda Singh Bahadur Engineering College in India; and (iv) Instructor for Cognizant, a 500-fortune software company. I was nominated for the TU Outstanding Teaching Award by the Effectiveness Committee at the College of Engineering and Natural Sciences in 2021.

2. Teaching Philosophy

My goal as a teacher and mentor is to provide foundations that help students to excel in their professional careers and become exceptional human beings. My educational philosophy is based on four pillars: (i) education and knowledge are critical to success; (ii) taking charge, being more assertive is important; (iii) "the brain is like a muscle" as it responds to mental effort similar to the way our muscles respond to exercise; and (iv) critique the problem/project and not others/team-members, be kind and empathetic.

My role as an educator is to facilitate *accurate mental models.* In my extensive teaching experience, I have seen students devote considerable effort to establishing a mental picture, or mental model, of how the parts of a system (such as a computer program) interact. These mental models serve as simplified causal models that help the student design new systems and predict the behavior of those systems. Therefore, I monitor how students are building mental models, by asking them questions about a computer program's behavior in class and on exams, and to help them repair misunderstandings. A student commented, "*I appreciated your large diagrams, commitment to writing out code … and your commitment to answering questions.*"

I strongly believe that computer science, like a foreign language, is *best learned by immersion*. As such, simulations, debates, brainstorming, role-playing, thought experiments, and interactions with customers and users are a regular part of my classes. The projects/assignments, as well as class exercises, are designed to follow the lecture material to bolster immersive learning. This is evident from one of my student's comments: "*The instructor's use of examples to explain concepts was a nice way to learn. In addition, the use of in-class exercises was fun and a good way to test what we just learned*".

The catalyst of student success is *active learning,* and my lectures usually include in-class activities involving problem-solving and critical thinking. For example, I frequently use *Think-Pair-Share* activities in which students are encouraged to think independently, and then discuss their solutions with their partners, and discuss them with the class. This introduces students to collaboration skills as well as to different ways of solving a problem; hence, building their confidence and respect for different perspectives. A student in one of my classes commented, *"Class activities were helpful and the instructor guidance improved our understanding of the concepts"*

My methods teach students how to *think creatively* and *communicate* effectively. Therefore, my courses integrate *design thinking* to allow students to develop their projects using cognitive, strategic, and practical processes. For example, students practice *divergent thinking* by noting at least 20 project ideas and *convergent thinking* by narrowing this list down to three best ideas and then to one. A student commented "*The instructor encouraged group work and individual creativity at the same time."* Students also learn to communicate effectively with professional writing and presentations.

I strive for a *balance between theoretical and practical approaches* as the impact of a hands-on experience tends to last for a much longer time. Hence, all my courses are project-based where students work on real projects and practice the concepts learned in the classes. The evaluation of the courses is distributed 50-50 so that the students are tested on both theoretical and practical aspects of the course.

I encourage *knowledge-sharing* of real-world software problems and their implications. Students in my classes research and present (i) bad software designs and bugs ("Software blunders" in *CS 4503*) that led to catastrophes to human life, financial losses, and ethical issues; (ii) technological advances that have transformed human lives ("Wowed me" in *CS 4513)*; (iii) latest or upcoming transformative research ideas ("Technological advances" in *CS 5863/CS 7863);* and (iv) the awareness of good ("hall of fame") and bad design ("hall of shame" in *CS 4053/ CS 6863*). This helps students learn the tenets of a good/bad design, and to become cognizant of software issues as well as implications in CS and non-CS fields.

3. Curriculum Development Efforts and Courses Taught

At TU, I have taught courses related to my research interests, i.e., Software Engineering (SE), Human-Computer Interaction (HCI), and the fostering of coding skills for novice programmers (students). This has given me the opportunity to apply experiences from research to teaching and vice-versa, reducing learning barriers and honing the coding skills of students.

In my six years at TU, I have developed and taught eight courses related to my research. Current courses I teach are:

- *CS 4503: Senior Software Projects I (Fall 2016-2021)* and *CS 4513: Senior Software Projects II (Spring 2020-2021) – class size: 25 to 74 students.* These are mandatory courses for senior students and act as a transition from novice programmers at university to professional developers in the software industry. The courses teach and evaluate students on SE concepts and processes. Students complete a yearlong project, write reports, and give presentations with a project vision, goal(s), list of features implemented, project demo, burndown charts about their progress, a review report, and a retrospective report. The projects are evaluated by me every 2-3 weeks during the semester and by external judges. In *CS 4503* the judges include software professionals from industry*;* e.g., in Fall 2020 judges were a software architect, a manager, a researcher, and a developer from software industries in Texas, California, Norway, and India. In addition, judges gave their expert feedback on the project's idea, design, architecture, and students' presentation styles. In *CS 4513*, judges include non-computer science professionals from academia or industry. For example, in the Spring of 2021, judges were experts from Political Science, Business, Psychology, Economics, and Biology. During their presentations, students used creative strategies to communicate to a wider audience as in the TV show Shark Tank (innovation, usability, and motivation). Students commented:

  > *"I've learned about project maintenance in the development life cycle. I've also learned about gathering and developing specifications, along with what software requirements specification document looks like. I've learned about design specification and class diagrams."*
  >
  > *"She did well in checking in with all of the groups and providing excellent feedback and suggestions for improving upon the projects Organized schedule, she is on time and well assignment provided."*

"instructor made sure the student were attentive, participating and contributing to each class and were not just passively riding the class."

● *CS 4053/CS 6863: Interaction Design (Spring 2016-2020, Fall 2020, 2021) – class size: 10 to 16 students.* This is an undergraduate and graduate course that teaches foundational concepts of HCI. This course has students from mixed backgrounds; they differ by (i) school experience: sophomores to seniors; (ii) work experience: none to multiple internships; and (iii) majors: CS and non-CS (Arts, Engineering, Management, and Science). Hence, I adapt the course material and lecture examples to address diverse student backgrounds. Students commented:

"Understanding of how humans interact with systems; - Expanded my analytical abilities; - developed an understanding of the design process and applied various methods and tools currently used; - understood ways of evaluating designs; - expanded my understanding of developing and conducting studies."

"-Had a very good understanding of all the material and was able to convey that knowledge in class -Did a very good job of relating material to real-world applications and job opportunities."

"- The instructor presented material that are applicable to students whether planning on going to industry or academia; - The instructor expanded on some ideas and cited numerous research papers that back up the material and help students who wants to learn more."

● *CS 5863/CS 7863: User-Centered Design and Research (Fall 2015, 2019; Spring 2021) – class size:5 to 7 students.* This is a special topic HCI course for undergraduate and graduate students. The course lectures emphasize major classes of inventions in user interface software and technology. The semester-long project intends to teach students to identify open questions and use theories of user interfaces to describe, analyze, and critique user interface technology. Students commented:

"I felt happy that I accomplished many tasks this semester. I have read and discuss 12 research paper. I have done 9 different presentation. I have written 18 short reports. I have received useful feedbacks from the instructor"

"1. How to research (from finding good papers on a subject to setting up a well-designed experiment) 2. Types of interfaces and ways to use them 3. How to write (and find in other papers) a problem statement 4. How to evaluate other research papers and come up with new research ideas 5. Steps in formulating a user-centered product/software"

"the instructor did great job. I don't think I can expect anything better than what she has done during this semester."

My lectures are based on foundational concepts (both theoretical and practical) and examples/experiences of the latest technological advances in industry and research. A student commented, "*She is always keeping her students up to date with the era of technology by talking about cutting edge technologies in computer science field."* My lectures, examples, and tests/assignments emphasize the importance and challenges of understanding and designing for diverse users and customer needs. For example, in Senior Projects, a student team created an educational app for deaf people.

Tests and assignments do not necessarily assess a student's ability or potential. Hence, I evaluate students' performances using multiple evaluations including: (i) semester-long projects; (ii) project presentations; (iii) presentations on positive/negative implications of software design/implementation; and (iv) in-class exercises. The undergraduate courses are specifically evaluated with written exams, quizzes, and assignments while graduate courses have reading assignments related to research papers from the ACM digital libraries. While evaluating course work, I use a "growth mindset" that focuses on developing intelligence through effort and practice. A student commented that they learned, *"the ability to analyze*

some hard computing problem. the ability to communicate effectively with others. the ability to listen, discuss, read, and present my ideas and assignment's solutions in front of others. learning more information about design and research skills."

In group-based projects (*CS 4503, CS 4513, and CS 4053/CS 6863)*, students are encouraged to (i) work on real-life innovative projects; (ii) use tools and techniques to collaborate, communicate and coordinate, as that is the heart of global software development; and (iii) complete a Likert scale survey for peer-evaluations of team members. Further, students also self-report their project contribution, motivating them to become effective team members. In CS 4503, a student commented "*I LIKE THIS CLASS IT'S PRETTY GREAT. But actually, working with a team and learning about how the industry handles software development is fascinating, and working on more open-ended personal projects has been liberating.*"

I have designed two advanced courses, namely *CS 4863/CS 6863: Empirical Software Engineering (Spring 2016)* and *CS 7863: Advanced Software Engineering (Fall 2016).* These graduate courses are seminar-based, where the first half of the class includes lectures followed by research paper discussions and student presentations. Classes are structured to promote close interactions with students and discussions of the latest research developments and industry practices. The goal is to promote critical thinking by engaging students to read, present, evaluate, and discuss existing research from recent premier conferences and journals. I also co-developed *CS 1003: Code@TU (Summer 2017)* a three-week workshop on Internet of Things, Security, and Big Data for high school students and future TU students.

I also taught an introductory programming course, *CS 2003: Fundamentals of Algorithm and Computer Applications (Spring 2017-2019)*. This course encourages creativity, fosters team skills, and develops self-confidence in students, which are important contributors to retention in the computer science program. I take measures to identify students that may need extra assistance before it's "too late." I strive for a high level of interaction (e.g., by preparing and asking questions) to ultimately get students to ask their own questions. I use scaffolding learning and problem solving to engage students, allowing them to break down problems and understand the bigger picture. A student commented that I "*Was always willing to help and clarify things that may have been confusing or unclear. Was very precise with what she wanted from us which made it easier to concentrate on completing assignments*."

Interaction with Students Outside the Class: My commitment to engage, challenge, and inspire students is not limited to the classroom. Beyond the classroom, I support students by (i) being available for their queries regarding assignments, projects, jobs, and challenges by emails or one-to-one meetings; (ii) adapting to their needs, as I believe "*Change is the law of life;*" (iii) resolving group conflicts, encouraging them to focus on the success of the project and keeping their individual differences aside; and (iv) being compassionate during students' personal problems and supportive by writing recommendation letters. A student commented, "*Dr. Kuttal was always accessible outside of class, and would always address questions in a timely manner, not only that, but the material for tests was always communicated very well.*"

Industry Partnerships: Strong industry partnerships are powerful engines of innovation and economic growth and with this aim, (i) I collaborate with industry researchers and invite industry experts to discuss their experiences in my classes. One student said, *"I appreciated bringing in GitWit* [a design company]*, and it was beneficial to hear feedback*;" (ii) visit industries and experts to learn the latest practices. For example, in the summer of 2018, I visited Varain Systems in Palo Alto. Meeting with the director of the UX team, UX researcher, software architecture, and project manager helped me learn SE processes and design processes followed in their company. In the future, I plan to visit different software companies yearly to enrich my knowledge and share these experiences with students; and (iii) discuss interview questions and experiences. As a student said, "*Discussing applications of the concepts to interview-like questions were extremely helpful, and I know these applications will help me in interviews in the future.*"

Integrating Diversity into Education: Students in my classes work on diverse teams. This helps to encourage students of mixed skill-sets (e.g., coding, management) to learn from each other's skills, and collaborate and partner with other diverse peers (e.g., gender, ethnic groups) who can provide a different perspective. The students also learn to consider the diverse population of users for design thinking, creativity, and creating software applications. I regularly participate in NCWIT summits and engage NCWIT resources to retain female and underrepresented (including rural) students. I adapt my (i) teaching strategies and course materials based on the latest research; (ii) mentoring models by determining the most effective mentoring strategy for each student; and (iii) communication techniques; for example, I moved to *discord* for remote communication as it is used by the current generation. In my HCI classes, students learn to find and fix gender-related biases in software design using the GenderMag method. In 2018, I was a finalist for the TU Women and Gender Studies Program's Linda J. Lacey Award for Mentoring Excellence.

4. Graduate/Undergraduate Research Student Mentorship

My collaboration with students translates into research. My research students work on a number of problems at the intersection of HCI, SE, and AI. I inspire students in my research lab to work independently and in groups, tackle open-ended problems, conduct experiments and analyze data, and develop effective communication skills. Currently, I am supporting or have supported 5 graduate (3 MS and 2 Ph.D.) and 29 undergraduate student-researchers since starting at TU in 2015.

I have had undergraduate students work alongside graduate students to help support and sometimes initiate research. I typically mentor 4-5 undergraduate students a year, some as early as their freshman year. I also work with students from diverse disciplines (e.g., Psychology and Mechanical Engineering). The students working in my lab establish close relationships with external researchers (e.g., IBM TJ Watson, USA, Microsoft Research Cambridge, UK, and Laureate Institute for Brain Research, USA) and academicians (e.g., University of California - Irvine, University of Oregon, Oklahoma State University, North Carolina State University, University of Nebraska-Omaha, and University of Tartu - Estonia). This collaboration provides unique opportunities to students, broadens their horizons, fosters strong relationships, and makes their research truly relevant. To date, 5 undergraduate students have collaborated with IBM researchers. The students also interact with other researchers by presenting research papers at key HCI and SE venues. I have published 12 research articles with 17 undergraduate students at TU, and one of my mentees (first-generation hispanic woman) is pursuing a PhD at Colorado School of Mines and another pursued MS at University of California-Irvine (first-generation Asian).

I work hard to maintain a diverse and close-knit research group. To date, I have supported eight female, three Hispanic, and fourteen Asian students through my research. The research conducted in collaboration with two undergraduate students (both first-generation women of color) received an Honorable Mention award at the ACM CHI Conference. To support undergraduate research in my lab, I have secured funding from Tulsa Undergraduate Research Challenge (TURC), federal work-study program, and NSF REU.

5. Summary

Teaching has been the most satisfying experience and has given me joy, knowing that I have positively influenced students' lives. I aim to inculcate problem solving, creativity, design thinking, planning and collaboration as part of their rationale. I look forward with great excitement to continuing my academic career to teach, mentor, and research.