Being an educator and a lifelong learner is not only my childhood dream but also my eternal goal. Over the years, I have been privileged to meet many brilliant teachers and students who take passionate interests and participation in my success. Carrying such support forward drives me to pursue a faculty position. My philosophy as an educator is to create an *academic permaculture* where a sharing and caring ecosystem nurtures respect and reciprocity in human relationships. Following the design principles David Holmgren articulated in his permaculture book [1]: 1) **observe and interact**, 2) **creatively use and respond to change**, 3) **use and value diversity**, 4) **design from patterns to details**, 5) **apply self-regulation and accept feedback**, 6) **integrate rather than segregate**, I detail my existing efforts to do so.

1 TEACHING EXPERIENCE

My formative experience as a teaching assistant for Prof. Tiffany Barnes's Discrete Math class at NC State University demonstrated that the *essence of teaching* is to **observe and interact**. I prepared myself by attending the classes and finishing the homework before giving detailed instructions to students. I disseminated my insights and knowledge to students by explaining the discrete math terminologies and reasoning the solutions to specific problems. My goal was to foster a friendly environment with might and main to assist students to succeed. I still proudly remember that I helped a student who constantly attended my office hours to pass the final exam even though she failed the midterm. At the end of the semester, Prof. Barnes appraised me as "a consistent and respectful team player".

In the subsequent semester, working as a head teaching assistant for the Discrete Math class taught by Prof. Dennis Bahler, I understood that the *essence of teaching* is to **creatively use and respond to change**. Since my accumulated familiarity with course-related materials from previous experience, I was responsible for developing exam questions, moderating discussions on Piazza, and hosting review sessions. I designed questions with the consideration of relevance, inclusion, and creativity. For example, I applied summation, product, and power functions to induction problems. To better strengthen students' problem-solving skills via trial and error, I proposed multiple sample tests, which raised heated discussions on specific problems on Piazza. I made a provisional decision to host additional review sessions to clear students' confusion. I found it particularly rewarding to see the attentiveness of students and their built-up confidence after the sessions. As Prof. Bahler said, "it takes a village" to manage a large class with 200 students, but I couldn't resist the happiness and fulfillment from the communications and interactions with the students.

My most recent experience has been with Advanced Distributed Systems, a graduate-level class taught by Prof. Xiaohui Gu. It taught me the *essence of teaching* as to **use and value diversity**. I helped reconstruct the class syllabus with an updated list of classic and state-of-the-art papers, connecting the canonical distributed concepts with cutting-edge cloud computing technologies. Students transitioned the learned knowledge from courses into concrete and applicable term projects, where diversified topics are proposed—from VM/container management to anomaly detection, from log analysis to privacy-preserving consensus. I tried my best to give concrete feedback to students' paper review assignments, help them set up evaluation environments, and suggest proper workload benchmarks for their implemented systems. These efforts were well-paid when I saw them presenting papers and writing project reports with creativity and logical reasoning. I especially appreciated the students' diversified opinions and unique mindsets, igniting unstoppable sparks of interest in today's fast-pacing technologies.

2 MENTORING EXPERIENCE

I have had the fantastic opportunity to work with several talented students. My mentoring philosophy is to help students develop their own professional identity through the process of conducting

research. By contributing engineering effort, empirically exploring workable solutions, and eventually delivering well-crafted systems, students get more proficient at the skills and more confident with the research space. Once they are ready to dive into more challenging tasks, I help them target a specific research problem, adopt appropriate methods to address it, and organize their findings into technical reports, open-source tools, or published papers. Here, I summarize my experiences with three students.

Pavithra Iyer, a NC State University MSc student, began working with me on the DScope project in 2017. This experience taught me the *essence of mentoring* as to **design from patterns to details**. By imitating my proposed testing model and following a few samples I gave her to run DScope on the Hadoop and Cassandra systems, Pavi broadened the testing scope on more open-source cloud systems by compiling source code and constructing testing scripts. With the growing familiarity of the project, she helped validate the effectiveness and accuracy of DScope by manually checking true and false positives. Her hard work formed an irreplaceable part of our evaluation, making it possible for DScope to be accepted to SoCC 2018. After graduation, Pavi joined Google as a software engineer.

Sam Cheng, a junior Ph.D. student from the University of Illinois Urbana-Champaign worked with me on the risky script project in 2020. This experience made me understand the *essence of mentoring* as to **apply self-regulation and accept feedback**. Sam constructed several automated scripts to crawl, pre-process, and extract Shell scripts from StackOverflow. He then empirically categorized the extracted scripts with their risky patterns. He found that many extracted risky scripts can be detected by the state-of-the-practice tool ShellCheck. To better understand the risk script patterns in the wild and the coverage of ShellCheck, Sam suggested extending our benchmarks by exploring Github. Following the same methodology, he found that many risky scripts in Github were semantic bugs, which cannot be identified by ShellCheck's syntax checking. With all the findings, Sam presented them in a poster.

Bekir Turkkan, a senior Ph.D. student from the University at Buffalo worked with me on the GreenABR project in 2021. This experience underlines the *essence of mentoring* as to **integrate rather than segregate**. With my system background and Bekir's machine learning background, we integrate reinforcement learning into designing an energy-efficient video streaming system. I helped him prototype the design, evaluate it with multiple videos in different genres, and document the techniques into a submitted conference paper. Bekir is continuing to work with me until the end of 2021 to develop more exciting projects.

3 TEACHING INTEREST

Given my background and experience, I am well prepared to teach operating systems, distributed systems, and program analysis. I would love to teach courses in a lecture-driven way covering canonical concepts and a discussion-centric way following cutting-edge technologies. I would make hands-on projects an essential of these courses, starting from small ones to comprehensive and even publishable ones. I would also enjoy hosting advanced graduate courses, seminars, and reading groups in my research areas.

The *principles of permaculture* will continue inspiring me to become a committed faculty, who will 1) value the diverse background of students, 2) design courses from patterns to details, 3) adjust course contents by introducing modern techs to students, 4) observe students attentiveness and interact with them to understand their needs, 5) constantly self-evaluate and openly accept students' feedback, and 6) integrate interdisciplinary knowledge to courses.

REFERENCES

[1] David Holmgren. 2002. Permaculture: Principles and Pathways Beyond Sustainability. (2002).