Teaching Statement

Satadal "Sata" Sengupta

I view teaching and mentoring as among the most exciting and meaningful parts of an academic career. They allow me to help students grow technically and intellectually. As an educator, I aim to build **inclusive learning environments**, emphasize **hands-on learning grounded in real systems**, and cultivate **intellectual independence in the age of AI**. Whether in the classroom or through one-on-one guidance, my goal is for students to leave with not only stronger technical foundations, but also the curiosity, judgment, confidence, and independence to take on difficult problems for themselves.

Teaching Experience and Approach

My teaching experience spans undergraduate and graduate courses in networking, mobile systems, programming, and basic mathematics. At IIT Kharagpur, I served as a teaching assistant for five semesters across *Computer Networks*, *Performance Modeling of Computer Networks*, and *Smartphone Computing and Applications*, contributing through assessment design, grading, office hours, and guest lectures. At Princeton, I served as a teaching assistant for *Advanced Programming Techniques* for two semesters, mentoring teams on real-world full-stack software development projects. In Fall 2024, I co-instructed *Basic Mathematics* to incarcerated students through Princeton's *Prison Teaching Initiative* (PTI). These experiences have shaped three core elements of my teaching approach.

- **Teach through real problems, projects, and systems.** Students learn best by getting their hands dirty. In networking and systems especially, core concepts become much easier to grasp when tied to systems students already use, such as Netflix streaming or Zoom conferencing. In the networking and mobile systems courses I supported at IIT Kharagpur, lab work asked students to implement protocols and systems from first principles, connecting theory to real implementation. I plan to build on this in my own teaching through testbeds like *CloudLab*, emulators like *Mininet*, and direct interaction with routers, firewalls, and cloud services. I saw the same benefit of real-world, hands-on learning in Princeton where I mentored teams building full-stack applications for real users, including tools for housing management, beekeeping, dating, and course selection. Four of these projects were later adopted by the University. I also saw this benefit in PTI, where I taught basic mathematical concepts through examples from running a farming business, such as addition–subtraction through profit–loss and perimeter through fencing costs.
- **Build an inclusive and fair classroom.** Effective learning begins in a classroom where students feel comfortable asking questions and sharing ideas, regardless of background, preparation, or confidence. This became clear to me at PTI where students brought varied educational histories and learning styles, and the classroom operated under strict restrictions on electronic tools or outside resources. Teaching there pushed me to be deliberate and adaptable. I designed lectures that assumed no prior knowledge, relied heavily on board work, revisited concepts from multiple angles, and used group exercises so students could learn from one another. This was reflected in student feedback, including one comment that this approach helped students persist even when the class became difficult. I saw the same principle again while teaching at Princeton during the pandemic, when students were dispersed across locations and faced unequal access to time and resources. Working with the course staff, I helped adapt expectations and evaluation by understanding each student's constraints and guiding teams to divide work so contributions balanced out fairly over the semester. Going forward, I want to carry this principle into my teaching by designing classrooms that are both welcoming and fair to students with different backgrounds and constraints.
- **Teach for independent thinking in the age of AI.** AI tools should complement learning, not replace thinking. Students should learn to use them well, but they must also reason independently, explain their choices, and avoid over-relying on automation. I saw this clearly at Princeton, where the hardest part of a strong project was not writing code, but choosing the right problem, understanding user needs, and designing a solution that fit real requirements and constraints. LLM tools may speed up implementation, but they cannot decide what is worth building or why. Going forward, I want my assessment to reflect that reality through closed-book assessments, oral exams where feasible,

credit for class participation, and projects that require ownership of both problem and design. I have already used this approach successfully at Princeton and PTI through project discussions, board work, closed-book quizzes, and close attention to class engagement.

Potential Courses

My research and teaching experience prepares me to teach undergraduate courses in areas such as computer networks, networked and mobile systems, network security, and programming. In these courses, I would connect core ideas to real systems and data, from large-scale media delivery and cloud services to emerging applications such as augmented and virtual reality.

At the graduate level, I would be excited to teach advanced courses and seminars on topics such as programmable networks, multimedia applications and networks, and machine learning for networking. These courses would draw on my research experience studying large-scale systems such as Zoom and Netflix, working with Internet measurements and production network traffic, past work on learning-based traffic classification and video streaming, and ongoing work on safe AI for network control. More broadly, I would want students to see networks not just as conduits for traffic, but as a rich systems domain shaped by application demands and operational constraints, and improved through measurement and control.

Mentorship and Advising

My mentoring experience spans undergraduate and graduate students at Princeton and IIT Kharagpur. At Princeton, I have mentored students on routing security, video-quality degradation, and the network behavior of AI chatbots, including work that led to a NINeS paper. At IIT Kharagpur, I mentored students on projects in video streaming, mobile sensing, and applied networked systems, several of which led to a short paper at IFIP/IEEE IM, a poster at ACM MobiCom, and *Best Poster* and *Best Academic Demo* awards at IEEE/ACM COMSNETS. These experiences have shaped my approach to advising, helping students grow not only in technical ability, but also in research judgment, confidence, and independence.

- **Provide guidance while fostering independence.** One of the most important lessons I learned from my own advisors is that good mentoring requires both perspective and restraint. At key moments, they helped me step back from details, reframe problems, and reconnect to the bigger picture without limiting my intellectual independence. I try to bring the same balance to my own mentoring by helping students identify questions that are both technically interesting and realistically answerable, offering strategic guidance when they get stuck, and giving them room to explore ideas, make mistakes, and develop their own research judgment. In my own mentoring, I have already seen how this balance helps students grow into more confident and independent researchers, with publications as one visible outcome. My current work with Veronika, a Bachelor's thesis student studying ChatGPT's network behavior, is perhaps the clearest example yet, with the project now moving toward a submission to IMC, a leading venue in Internet measurement.
- **Connect research to real systems and practice.** Networked systems is an applied field, and many of its best problems come from real systems and operational pain points. My own work has been shaped by those settings. At Netflix, I saw firsthand how research can inform large-scale production systems. My ongoing engagement with NVIDIA has informed my work on hardware-software co-design for applications such as video conferencing and intrusion detection. Through collaboration with Princeton's Office of Information Technology, I learned about operational challenges on campus networks that informed my SIGCOMM and NINeS papers. As an advisor, I would encourage students to pursue internships and collaborations that expose them to operational challenges, and to follow technical writing from organizations such as Cloudflare and Meta on their systems and problems. I want my students to have similarly enriching experiences and to see that some of the most exciting research problems arise where elegant ideas meet messy real-world constraints.

More broadly, I view advising as a way to help students develop not only technical depth, but also the habits that strong research requires: framing problems well, designing sound experiments, writing and presenting clearly, building reliable and reproducible systems, and persevering through setbacks. I want students to leave my group not only with strong research outcomes, but also with the judgment, resilience, and independence to thrive as researchers or engineers in academia or industry.