Teaching Statement

I am particularly eager to transfer the knowledge acquired during my PhD by teaching subjects related to my area of expertise. So far I have taught courses like Computer Networks, Operating Systems, Object Oriented Programming with C++ for the UG students during my tenure at SRM-University, AP. where I received a student feedback rating of 4.21 for the Computer Network course. In my current assignment at IDRBT, I am instructing courses on Internet technologies for the PGDBT students where I have received 3.82 and 4.2 course feedback rating. Additionally for the last 2 years I am teaching Advanced Operating Systems to the M.Tech students at University of Hyderabad. I rely on a more hands-on approach during my teaching, for example in my Advanced Operating Systems course I often use code snippets from either XV6 or Linux Kernel repositories which encourages the students to look into the actual OS implementation to correlate concepts with the actual implementations. I use similar strategies to teach different network protocols as well.

Given an opportunity, I would be interested in teaching the following courses.

1. **Advanced Operating Systems Design:** This course aims to provide participants with practical, hands-on experience delving into the internals and advancements of operating system concepts covered in undergraduate OS courses. Beginning with an in-depth exploration of the Linux operating system, participants will engage in practical exercises focusing on the latest developments in the Linux Kernel. Topics covered include process handling, process scheduling, memory management, system calls, and interrupt handling specific to Linux-based operating systems. Next we'll move to the concept of virtualization which will cover the design of Xen virtualization platforms, memory management aspects during virtualization, various concepts of lightweight virtualization, and so on. We'll also discuss and try out containers -- the latest developments in virtualization technology. Then we'll discuss the file system details starting from Unix Fast file system, Log structured file system, networked file system (NFS) and Google's data center file system. Finally we'll touch upon some more advanced topics like Trusted Execution Environments (TEE), Embedded operating systems like optimizations done in Android design, kernel level sandboxing, and various applications of machine learning applications over computer systems. We'll also cover different innovative technologies like automated log analytics, and distributed processing of large scale data have added various new and challenging flavors to this classical computer science topic.
2. **Computer Networks:** In recent years computer networking has gone through a lot of changes. With the advent of advanced concepts like network functions and software defined networking, the layer-wise study of networking is fading. However, I think the core concepts of protocol stacks can help students realize the complexity of networked systems. Therefore, given an opportunity, I would like to organize the subject using a two-pass method where the first pass is reserved for the top-down approach and the second pass would be dedicated to the bottom-up approach. The top-down approach can help the students become familiar with the individual concepts like application-specific overlay networks, reliable and unreliable delivery, congestion control, inter & intra-domain routing, link layer discovery mechanisms, etc. During the second pass, the students can learn about the individual protocols, their history, and how they impacted the network control mechanisms. Finally, this course must contain hands-on sessions on real and emulated environments to enhance students' interest.
3. **Trends in Modern System Design:** This course deals with advanced topics in the field of computer networks and Operating systems targeted towards more interested students. I want to focus on the traditional design and the evolution of Operating systems and computer network architectures during this course. I intend to cover a few advanced concepts like eBPF vs LKM based approaches, clean slate Internet architectures, programmable data & control planes, network function virtualization, dynamic adaptive video streaming, etc. Given the opportunity, I would like to design this course to make students aware of the current research trends. With the rising popularity of machine learning-based approaches in the research trends, I would like to

touch upon cognitive radio and reinforcement learning-based distributed scheduling problems. The course also consists of multiple hands-on exercises on modern system emulation tools like Mininet/mininetWifi, Docker with Kubernetes, CloudSim/iFogSim, P4, etc., to provide the students an actual flavor of the domain.

4. **Modern Distributed Platforms:** Recent trend in distributed computing largely relies on the edge-cloud continuum. This course will be beneficial for the students who are interested to learn about the fundamentals of modern distributed computing paradigms. In this course I would like to start with the fundamentals of cloud computing and slowly move towards the edge, fog, droplet computing paradigm. I would like to touch upon the classical research problems like scheduling and task off-loading challenges in this context. I also think a basic idea about the following technologies will also be useful; which would be the part of the curriculum as well. Namespaces, Docker, AWS, GCP, Kubernates, TSN

Apart from these subjects, classical subjects like the introduction to computing, operating systems, and distributed systems are necessary to understand computer science fundamentals. Given the opportunity, I am also ready to teach these subjects with minor preparations so that both the students and myself can upgrade our understanding.