Jose M. Faleiro — Teaching Statement

Over the past two decades, computer science has had an increasingly large role to play in people's daily lives. With this role only set to increase in the future, computer science *education* will have an important impact on not just practitioners, but – through technology they build – the public at large. For this reason, I believe that the effective communication, via teaching, of both technical know-how and a systemic understanding of technology's role in society is critical to drive positive impact in the world. The opportunity to teach is therefore one of my main motivations for seeking an academic position.

As a PhD student, I have served as a teaching assistant on advanced classes on operating systems and database systems. I have also thrice served as a teaching assistant on larger undergraduate classes on introduction to programming and software engineering. While teaching the operating systems class, I held weekly office hours, prepared and graded exams, and taught lectures (on multi-core synchronization, read-copy-update, and scheduling). Additionally, I helped students build a multi-core operating system kernel in C and assembly. In the database systems course, I helped design the syllabus, gave multiple lectures (on multi-version concurrency control, main-memory multi-core database systems, and large-scale distributed analytics), designed a hands-on experimental lab, and mentored students' research projects.

In future teaching work, I plan to incorporate the lessons I learned from this experience. I learned the importance of **experimenting with real code** in systems-oriented classes. For instance, I designed a hands-on lab on database process models in which students conducted a set of experiments to document the tradeoffs between multi-threading, multi-processing, and their pooling variants. The lab proved popular with students, and has subsequently been incorporated in later offerings of the class. I also learned the importance of a **comparative approach** to learning in systems-oriented classes. For instance, while teaching lectures in the database systems class, I deliberately compared approaches to the same problem, say database indexing, in early database systems with those of modern systems. A comparative approach helps students put prior research in the appropriate context, and distinguish between the fundamental constraints of a problem from those that might be subject to change, such as constraints that arise due to hardware performance characteristics or application-specific requirements. Comparative approaches to learning can therefore help students develop critical thinking when doing research and building systems.

As a new faculty member, I would be qualified to teach classes on database systems, big data systems, data science, distributed systems, operating systems, systems programming, data structures, and introductory computer science. At the graduate level, I would be happy to lead seminars on modern data management systems, systems for machine learning, and distributed systems.

Four of the undergraduate students I taught in the database systems class worked with me on their undergraduate thesis projects. I worked with students to define a well-scoped problem, and then mentored them while they implemented and evaluated a prototype of their solution. For instance, in the Fall of 2016, I mentored Stanislaw Swidwinski, an undergraduate student who worked on workload-sensitive transaction re-ordering mechanisms. Stan was able to show that transaction re-ordering could improve a database system's throughput by over order-of-magnitude on conflict heavy workloads. Seeing these undergraduates make tangible progress on challenging problems (some of which are the subject of active state-of-the-art research) was extremely rewarding. As a faculty member, I plan to continue mentoring undergraduates and give them opportunities to work on problems that are part of my broader research agenda.

I also spent time mentoring junior graduate students Joshua Lockerman and Juno Kim at Yale, and Chenggang Wu at UC Berkeley. I worked with Josh and Juno on a large distributed systems project. Juno was a first-year PhD student at the time, and I worked with him to design applications and experiments that best showed the performance characteristics of our system. Josh was the lead student on the project, and I advised him on design and implementation, evaluation, and writing. I mentored Chenggang on a project which explored the design space of building a coordination-free key-value store. I advised him on experimental evaluation, writing, and building the system itself. The resulting work for both projects is under review for publication.

Finally, I strongly believe in outreach to practitioners in industry. I have given onsite talks at various venues, including Microsoft Research, Amazon Web Services, and Apple. In order to reach a broader audience, I have also given talks about my research at two major practitioner focussed conferences, StrangeLoop in 2017 and Ricon in 2015. These conferences attract hundreds of participants every year, and provided an excellent platform to make my research accessible to a broader audience. I plan to continue this practitioner outreach as a new faculty member.