James Hughes

CPSP339P Final Paper/Reflection

My internship for this semester was to TA for a freshman entry level Engineering course, ENES100, for one section. My weekly hour count ranged from 10-20+ hours a week, with less hours needed later into the semester. My class consists of 40 kids, which I grade most if not all assignments for. I am expected to attend all lectures to help facilitate the class and host a certain amount of office hours every week. This internship is for the fall semester only.

My learning objectives for this internship included: Getting better classroom management skills, gaining better knowledge about programming Arduino and its platform, and lastly getting experience with the OTV Project, which I missed due to COVID restrictions last year. Regarding classroom management skills, I have found the best way to go about *everything* is to always be polite, but firm. If you do have to turn a student down, helping them work through an alternative solution to the design issue is the best thing to do, so that students feel less discouraged from any failures. Oftentimes giving students the tools they need to complete something is the best thing to do, rather than trying to tell them how to do it step by step. As of writing this paper, final OTV challenges have not occurred, and so I cannot guarantee success for every student in this project. However, most teams seem very on track to complete their OTV using the buffer week we have built into the schedule. By running a lecture, I was able to get the experience on coding that I wanted and could easily code something at my normal proficiency level in C++ if I wanted to. OTV experience has been something that I've been picking up as I watch others do the project, but most of it I already had from running robotics teams in the past. The most major learning points have stemmed from the electricity and motors portion of the project, as I did not understand a lot of the things that are required for selecting and using the correct motors, like torque versus rotational speed, or how an H-Bridge channels electricity in alternating directions to reverse motor turn direction.

A little more specification on what teachers want from TF's might be nice. This isn't a huge qualm, but sometimes professors give UTF/LTF's a lot of autonomy on things, to the point that TF's sometimes aren't doing what the establishment had plans for, which is primarily due to communication breakdown in some departments. This only occurred however with the structuring of office hours, which had several changes over the course of the year, some planned and some spontaneous. In the beginning of the year due to communication mishaps I signed up for an extra hour of office hours, despite having a *lot* of grading to do in the first half of the semester. When office hours were established, there were several points at which students needed to swap their hours to fulfill requirements, like Keystone hours, or a certain number of UTFs needed in the lab for safety reasons. Near the end of the year TF's were required to take an extra 2 office hours, but only after redoing the schedule twice in one week in between. I only increased by one hour since I had been doing extra hours and did not know it. Getting an inside look on how professors run their classes and plan things was certainly enlightening, and how teaching from professor to professor can differ heavily based on what the professor thinks is best for the class. Seeing how the leadership structure breaks down was cool, as there is a level of equality and respect between every member of the organizations, regardless of whether the person is a

UTF or a professor, but also that there is partitioning and subcategories of management and communication. There doesn't seem to be really any office politics because of this, there's no fear of looking silly for not knowing something in a classroom setting. For instance, my professor will often ask our on-staff LTF for help with something if he does not know how to solve an issue.

My professor's midterm assessment is accurate for my performance. I make it a point to try and provide explicit feedback on assignments in order to better help students learn the material, rather than punishing students for missed points. This does *not* mean giving them answers directly but reading through their work to find specific concepts or miscalculations to reinforce or correct respectively. While in the lab I just perform general troubleshooting procedures based on what the task is, and if I cannot figure out the problem, I will enlist an LTF (senior TF) to help. Slow grading is also accurate, as this was my first semester on campus, and having to figure out how to manage my time schedule was a challenge all on its own. At some points I was unintentionally sacrificing study time so that I could do grading, but as I got farther into the semester (and our grading policy was a bit more finalized) I was able to work out when I should do grading versus schoolwork. I did try to improve on my grading habits after seeing the report however, usually getting things done in the week of receiving them if I could, and I even distributed some rubrics online to help other TF's that are also working with my professor.

Being a leader in a professional environment requires you to be a leader first even more so than other places I've been. This stems from the environment being made up entirely of adults, but also because students are relying on you for their grade. While ENES100 is not a strenuous class in and of itself, most students do not know that until at least halfway through the course and assume the worst when they do not understand something or make a mistake. Being patient with these mistakes and teaching the students how to fix it is a very important part of the process, regardless of whether you've resolved the issue in your own class. Students in another class may have the same issue and re-explaining or re-establishing rules is a commonality in open lab hours. Once again, patience is the best virtue to have a non-cumulative amount of. Something that is also interesting, and that I may have mentioned elsewhere in this essay, is the command structure, or lack thereof. Everyone in the lab is treated as an equal, nobody is the 'boss' of anyone, only supervisors and advisors. There is a nonverbal agreement between everybody that people do not know everything, and that no one is expected to know everything. Questions are encouraged, and it makes solving issues very easy, as there's almost always someone more senior on staff you can ask.

Even though I took ENES100 online in my freshman year, the class still allowed me to learn the basic skills and curriculum associated with the class, but not necessarily how to carry out the OTV production process. I do have prior experience, before college I was in two different robotics teams, which helped me learn more about the integration process and methods for making an efficient robot. Other skills related to performing certain tasks like woodwork, 3D printing, or coding, I learned from experience in other projects, not classes. In my last internship (which was not eligible for Scholars, since it was in my freshman year) I was primarily coding in C#, while the computer brains for OTV's are programmed in C+. While the coding languages are not the exact same, they are similar in structure, generalized knowledge on coding

methodologies will transfer between languages regardless. In the same internship I also got a lot of hands-on experience with setting up and maintaining 3D printers. Knowing exactly how a 3D printer produces its products was directly applicable to advising on optimizing and making prints more efficient for teams, as 3D printed custom parts are an important part in constructing OTV's to perform certain tasks. Lastly, the second most important construction material was wood, and my grandfather taught me a lot of woodshop skills, or tricks to form a part without sacrificing a piece of wood's integrity. Of course, safety with all these machines is also important in keeping my students safe, which I made sure to enforce protocols for rigidly, or even do things for my students that required a little more precision. All of these skills greatly improved my ability to teach the class with little prior instruction, as I was even able to teach my professor some tricks. There are of course things I wish I knew before doing the internship.

In the technical department I wish I would have had better retention regarding electric circuits, and more knowledge on motor selection. Motor selection is not necessarily due to ENES100's curriculum, as online learning didn't require me to select a motor or work with one on any of the projects. Not having this knowledge did not hurt me too badly but having to ask an LTF or my professor for help whenever motor questions were asked was a little bit troublesome. While I did not retain knowledge on circuits from ENES100 very well, our projects often did not require any balancing for what parts we were using, as the power draw was either too low, or the circuitry too simple. As such, I had an almost equivalent amount of knowledge on electronics as the students I was teaching. This was rectified later in the semester by LTFs performing some electronics informational presentations.

The internship was overall a success for me, as I met my general and specific goals. While my lecture on coding was expectedly dry, I have rarely had students come to me with coding basics questions in my class, as they know how to better access resources to solve their problems. I have also included my Slides presentation on my E-Portfolio, as I find this was a large growth area in my internship, relative to both public speaking and my coding skills. Having to run a lecture forced me to step out of my comfort zone, but also to teach myself the coding syntax that I did not focus as much on when I initially took ENES100. One thing that I found extremely rewarding was to teach students skills that were not in the curriculum explicitly, like how to learn coding languages or use the woodworking tools. Being able to directly apply skills learned in my internship or prior clubs and experiences really makes me feel like taking the time to learn them paid off, and not just for my own personal projects, but others as well. Sharing knowledge like how to research coding issues, how to learn code syntax quickly, how to organize code, or even more mundane things like 'should I cut against or with the woodgrain' are all skills that may not be apparent to students for a long time but will give them a leg up in technical areas against their peers. As for learning goals specifically, if I don't know how to do something and I ask my LTF or professor to help, I try to stick around and see what they teach. Doing this I have learned a lot about parts of the OTV project I missed during online class, especially about motors and overall system integration.

While being in the lab for two hours is pretty draining, I really liked the job, and would definitely do it again. Being a TF is not as stressful as I thought it would be, and the nature of

ENES100 is a lot more conducive to teaching, rather than punishing students who do not immediately grasp concepts. The grading schedule near the beginning of the year is very heavy, but now knowing more about scheduling in college I am confident that I could handle the workload. Although I will be back to full credit load next semester and will not be able to be a TF, in later years where my credit load is less vigorous, I will certainly be a TF for ENES100 again. Being a TF is a pretty good way to make money, and while the time commitment can be hefty on some weeks, it is something that I really like doing as it is very rewarding personally, and I will do it again if possible.