Everyone deserves the opportunity to find out if they want to be a computer scientist. Unfortunately, ensuring everyone has that chance is not as simple as creating large, inspiring introductory courses. Creating an inclusive program requires careful decisions in every aspect of our courses and in all the course we teach.

Our field has systemic issues that we cannot fix overnight, but we can take steps to make our courses inclusive to everyone. I follow the established best-practices for making classrooms inclusive; we've discussed them at our CS Ed reading group and I learned more from Colleen Lewis's panels at the most recent SIGCSE. I even shared some of those best-practices with new graduate students at UW's TA conference this past summer. But building an inclusive course goes beyond what we do directly in a classroom. Two particular methods that make my classroom more inclusive are having a carefully-chosen, trusted course staff and responding to feedback.

Students benefit from seeing people "like them" leading their courses. At UW I have been teaching over summers, which means the pool of potential TAs is limited. Despite that I've worked to ensure we have people with different backgrounds on our staff. For Summer 2019 I was teaching a course for non-majors. I made sure we had some non-CSE majors as TAs, so that our students (many of whom were rejected from becoming CSE students) would **see** that people like them have succeeded. Having non-major TAs also helps our materials; TAs with experiences closer to our students' experiences are better-situated to make sure our materials are relevant and useful to them.

Throughout my courses, I work to make sure my TAs are comfortable giving feedback and contributing to the course. In my first staff meeting, I tell them I want their opinions on the course. In our last staff meeting we do a retrospective of the quarter: discussing what worked and what did not and where we can improve. Getting multiple people to look at materials is critical. Everyone has different blind spots. The more people that look at our materials, the fewer blind spots we have left. None of our homework assignments or section handouts were released without at least three people looking at them. That many pairs of eyes reduces mistakes, but more importantly it brings in multiple perspectives.

One of the keys to getting TAs to give feedback is to empower them when they do. That means I listen and pause to consider what my TAs have suggested (that sounds simple, and it is; but it makes a difference). When TAs volunteer to draft homework problems or solutions, I encourage them to do so – even when that causes me to spend more time editing than if I had just written it myself. My senior TAs were comfortable and capable of doing everything from suggesting schedule changes to generating new content for review sessions to writing homework problems.[1]

So far, I have mostly spoken in generalities – about processes rather than actual events; let me give an example. In UW's data structures courses, we spend the last few weeks on graph algorithms. The learning objective I care about the most in those courses is "graph modeling." Given a scenario, describe a graph and an algorithm to run on it that will solve the given problem. The skill is to extract from the "real world" the necessary graph and then from that graph the answer to the problem.

I first wrote these problems for Summer 2018, for our majors course. In our homework and section practice materials, I made the scenarios in the same way I had always seen them written: long stories, with (occasionally elaborate) jokes thrown in. I got feedback after the quarter ended (from one of my TAs, and from students through that TA) that those questions had been a challenge for some of our students. The long block of text simply took a very long time to read through for some of our students.

---

[1] I can't take credit for creating that culture; the real culture building was done by Kasey Champion, who is an expert at empowering her TAs. Those TAs had worked alongside me for a few quarters, so they knew I was serious about maintaining the ethos she built into the course. I intend to build that same culture wherever I go.

The length of the problems and complicated jokes were a mistake. While the problems were important, the particular instantiation was not inclusive of students who took longer when reading.[2]

When I taught data structures again (this time for non-majors, which has a large population with English as a learned language), I still believed graph modeling was a critical skill, but wanted to do a better job of asking the problems. I was particularly careful about the topics chosen. Instead of an elaborate joke, I went with a tiny one. I wrote problems about lemonbikes (a parody of limebikes that are ubiquitous on our campus, and a reference that would be immediately understood by anyone who had spent more than a few days in Seattle). Moreover we dedicated significant effort to minimizing the amount of text to be read. Compared to the previous quarter, we were able to reduce the length of that type of problem by about 40%.

The previous paragraph switched to first person plural for good reason. The final editing of that problem was done in a meeting with me and a few senior TAs – because getting it clear and concise was worth dedicating significant full-staff effort. At training for UW's TA conference, there was a phrase that encapsulated a lot of what I've written here: "inclusive teaching is effective teaching." Changing the wording makes the graph modeling problems better for everyone, not just the particular population we knew was adversely affected. Having multiple eyes on assignments reduces the chances of technical errors, not just non-inclusive or irrelevant examples. Creating an environment where feedback is welcome, actively soliciting that feedback, and acting on it make my courses more inclusive, and those practices make them better for everyone.

---

[2]The largest affected population that quarter was our international students, but it would also affect students with certain disabilites and students from other underrepresented groups who are more likely to have English as a learned language.