Craig Colegrove

Pete Tucker

CS-372

Homework 5

January 23, 2015

What is good?

As a perfectionist, my code will never be “good.” It might get to a point where it’s “good enough,” but that’s a very different concept. I’m very critical of my code, and hyper-aware of all the different standards and practices I don’t yet know. For example, I know very little about what types of sorting will be efficient for my situation. Because of this, I have never made a program that I find to be “good.” However, this does not mean that I’m never proud of my work. I like programming because it lets me solve problems that would be very difficult to solve manually. When I’m able to make something that could actually be *useful* to people, I’m pleased with my work. That’s why I chose my final project of a class scheduler. It won’t get to the level of details I would wish, but I think that it could help students plan their classes and see where they have free time in their schedules.

I’m not very focused on the aesthetics of my programs, as long as the functionality is not hindered. I much prefer a simple UI that is easy to understand is to a beautiful one that is confusing. Because of this, what I call a good program may be far different from somebody else’s definition of good. I like that CS 171 and 172 left graphics out of the curriculum, because it lets us learn more about the code itself and not be bogged down by how it looks. For me, how useful it is to the user is the most important feature.

As I’ve only taken CS 171 and 172, I believe that my Java final project is the best work I’ve ever done. My previous class projects felt very limited, both by what I knew and by what my partner expected of me. In this class, I’m able to do my own work, and solve problems my own way. Though it’s not there yet, my class scheduler program could become something that I see as good. When I’m working on code, the usefulness of it is what keeps me motivated. It’s no fun making a final project that will never be run once the term is over. In those situations, I tend to stop at “minimally viable,” because I don’t see a need to continue. When I can actually use what I’ve created to help people, however, I am much more motivated to follow through with the project until it approaches my unreachable standards.