In computer science, there are many problems involving multi-threading and concurrency but relatively few ways of visualizing the solutions. Three classic problems are the Readers-Writers, Producers-Consumers, and Dining Philosophers. When students encounter these problems in a course, they will likely see code for different solutions and think through how the threads would interact in the different cases. However, most material about the problems was lectures and series of static drawings with arrows, making it difficult for students to visualize and experience how the threads should interact.

Therefore, my focus in working with TSGL has been developing visualizations of these classic computing problems. The visualizations are designed to improve students’ understanding of the problems and their common solutions. In Dining Philosophers, we show both working and flawed solutions to demonstrate how some of the most intuitive solutions will prevent every thread from being successful. Readers-Writers shows a couple systems with different strengths.

In the process, we have also restructured much of the library. At the beginning of the summer, it was procedural, so a user would tell the Canvas to draw a shape given certain dimensions and properties. Now, we have made it object oriented, so a user creates the object and sends it to the Canvas for drawing and can erase it from the Canvas later in the program.

Benefits of working on the project include both technical knowledge and the experience of working a full time computing job. Over the past weeks, I’ve noticed how much my knowledge of the computing concepts has grown. Second, as I’m used to taking a variety of courses, I’ve grown through the nature of working on a technical project for 40 hours a week.