I didn’t really run into any super hard problems. It went relatively smoothly through the whole process.

The final schedule changes from run to run, but the one I landed on was fairly well distributed. The schedules always seem to converge on assigning the same teacher for most of the classes and it would schedule one of the classes that had a projected enrollment of 50 to the 450 person lecture hall but that would only happen one time per run.

The only changes that I would make is some of the weights in the fitness function. It prioritizes assigning most of the classes to the same teacher so I would probably not punish for teachers only having 1 or 2 classes scheduled to them.

The algorithm was fairly easy to code and the fitness function wasn’t that hard to translate, but for mine there was no trend really. It would get a random schedule that was really good but these were outliers and no where close to the average of any of the generations. I did make it in about a day and a half, but when I look at my program, I don’t see very many areas that could be improved, at least to the best of my knowledge, without using libraries that are already optimized and specialized in this type of work load.