**Tripp Mims and William Loving**

The problem that we aimed to solve was organizing classes based on a limited number of professors, timeslots, and rooms. This seemed interesting as it was applicable to UVA for organizing the classes within certain rooms of Rice Hall for Computer Science classes for example.

To begin there are number of classes and two lists with elements as for each time slot and room assigned to class such that with being the number of available time slots and being the number of available rooms. Both time slots and rooms are constrained by these values. In addition to this, no two rooms can be the same during the same time slot, so a constraint is added such that . Finally, a professor cannot teach two classes that are at the same time so for a professor of a class (denoted by ) . These are the constraints that are then searched to find a solution in z3.

The hardest part of this was finding all of the constraints that needed to be satisfied. Adjusting the constraints to finally formalize the problem in a way that z3 to solve was difficult as putting them into a formalization from natural language was not straightforward. This was interesting as having to ensure a professor was not double booked was one that was initially overlooked, and it became a problem that was interesting to solve. Evolving the constraints to finally get what was needed felt like a puzzle that needed to be solved. The other constraints were relatively easy as they were intuitive. In addition to this, creating code with z3 was easy as what was needed was easy to find especially after looking into the sudoku solver.