1. **Introduction**. What is the problem or task you are trying to solve? What methods are you using to solve them? (This may be answered by the assignment itself, but write it in your own words.)

The task that I had was to create a prolog program that looks at some of the CS classes offered at Brown and can output whether the class is an intro class, intermediate class, or upper level class based on various criteria such as its prerequisites and whether it is offered in the fall or the spring. Additionally, the program can output whether or not a student can take a certain class based off of the classes that they have already taken. Or, more broadly, it can output which students are eligible to take a class or all of the classes that a certain student is eligible to take.

2. **Methods**. What implementation choices did you make? How did you decide on those choices? How did you evaluate or test your system?

When coding this project, I first started by interpreting the given graph of all of the CS classes, which showed their various prerequisites. I made a fact for each course, which included the name of the course and its prerequisites. I then made facts for which courses were offered in the fall vs. the spring semester. I then made separate rules for whether or not a class is an intro class, an intermediate class, or an upper level class. I then made facts for various students and the classes that they have taken. I made a rule for whether or not a student can take a class based on whether or not the list of classes they have taken contains the prerequisites for the given course. I decided on these choices because the assignment explicitly stated that each of these rules were required. For the can_take rule I made special versions for cs141 and cs126 because the assignment specified to “hard code” them. The original can_take rule works for every other class in the database. I tested my system by testing a case that I expected to be true and a case that I expected to be false for each rule, to make sure that they worked properly.

3. **Results**. Did your system work? How well? Provide tables / numerical results if appropriate.

My system did work well. It correctly answered the provided questions/scenarios to test (in section 3.2). One thing is that I did have to hard code in the classes that each person has taken instead of interpreting the scenarios given. For example, question 3 says that “**sheryl** Sandberg is eligible to take all upper level classes.” Instead of making a fact saying that **sheryl** can take cs166 and cs126 (the upper level classes) and then being able to know all of the classes that she has taken from that, I simply listed out all of the non-upper level classes in a has_taken fact.

4. **Discussion**. Based on your results, were your methods and choices appropriate for the task? What would you do differently next time? Or if you had more time? Are there any ethical implications of your system that others should be aware of?

Refer to the methods section for the first question. If I were to complete this project again/if I had more time I would add a way for the program to output which other classes a student can take based off of one class that they can take and the classes that they've already taken. For example, maybe a student has taken cs22 and they can take cs33. One of the prerequisites for cs33 is cs16, which along with cs22 fulfills the prerequisites of cs141. In other words, the program would output that the student can take cs141 given that they have taken cs22 and can take cs33. There are no ethical implications to be aware of; the program does not know a student's age, gender, race, socioeconomic status, etc. It only knows the classes that a student has taken, so it cannot discriminate against someone for other factors.