Wentworth Institute of Technology
COMP3071 – Intro To Programming Languages
Fall 2016, Kreimendahl

Programming Assignment 2

This assignment is intended to work with a larger codebase in Prolog and apply knowledge of a declarative language to a problem that may occur in the real world.

1. Repository

First, fork the repository at `https://gitlab.com/comp3071fk/pa2` and use git to clone the repository to your local machine. Modify the `pa2.pl` file in the repository and when you are ready to submit, use git to add, commit, and push the file. Make sure that your submitted code is all in `pa2.pl` in your forked version of `PA2`, not a different file or project. Add me to the project on gitlab as a member with developer access so I have access to your code.

2. Specifications

The purpose of this assignment is to implement a database for students in classes. The supplied knowledge base will have facts about students, courses, classrooms, and class times. Your task is to implement some predicates to get information from the database.

```
when(1000,10).
when(1200,12).
when(3400,11).
when(3350,12).
when(2350,11).
```

```
where(1000,dobbs102).
where(1200,dobbs118).
where(3400,wentw216).
where(3350,wentw118).
where(2350,wentw216).
```

```
enroll(mary,1200).
enroll(john,3400).
enroll(mary,3350).
enroll(john,1000).
enroll(jim,1000).
```

2.1. Language

This assignment should be completed in Prolog. Please add sufficient comments to your code so that it's clear to a smart but uninformed reader what your intention is with a code block.

2.2. Predicates

Define the following predicates by writing a rule or rules for them. The examples are just that – examples. You will have to write some facts to test your predicates correctly and completely.

schedule/3

The schedule predicate should take the parameters of `(student, classroom, time)`. With this predicate, we can form queries such as `?- schedule(mary, P, T).` and get results: `P = dobbs118 T = 12; P = wentw118 T = 12;` This predicate can also query a classroom's usage with something

like `?- schedule(S, wentw216, T).` This will show all the students in a classroom and the times that they are there.

usage/2

The usage predicate gives all the times that a classroom is in use. For example, `?- usage(wentw216, T).` would result in: `T = 11; T = 12; .` The query `usage(X, 11).` should return all of the classrooms that are in use at 11.

conflict/2

A conflict exists if two courses are using the same classroom at the same time. `?- conflict(X, dobbs118).` should return false. The query `conflict(X,Y).` should return all room conflicts in the database.

meet/2

This predicate will decide if two students can meet each other, according to their schedules. There are two ways that two students can meet: either they can meet by being enrolled in the same course, or they can have different classes in the same classrooms at adjacent times (off by an hour). (For this predicate, you only need to return true for one ordering of the query. For example, `meet(mary, john).` might be true, or `meet(john, mary).` might be true.

2.3. Testing

The above knowledge base is also in pa2.pl. You will probably want to add facts to make for more robust predicate testing. I will be testing your submission on both this small database and a larger one, with more students, courses, and rooms.