Homework 10

April 1, 2014

Instruction

Write the answers to all questions in one text file of the name `hwk10.pl`.

Prolog programming

In this homework, you will implement Prolog program to check whether a list of class schedules is legal. Make sure that your program can be loaded into Prolog console and yield the correct output. I have provided a template file with some facts about class rooms, instructors' teaching assignment, and the class requirements.

You should implement the following predicates:

1. Define a predicate is_member(X, L) that returns true iff X is a member of list L.
2. Define a predicate no_time_conflict(T1, T2) that returns true iff there is no time conflict between the time periods T1 and T2.
3. Define a predicate has_instructor(C, I) that returns true iff C is a course taught by the instructor I.
4. Define a predicate no_instructor_conflict(I1, T1, I2, T2) that returns true iff there is no conflict for instructor I1 to teach at time period T1 and for instructor I2 to teach at time period T2. Note that I1 and I2 may be the same instructor. Also note that we use between(S, E) to represent time period starting from S and end at E. We use 24 hour format and use fraction to represent minutes. For example, 13.5 represents 1:30 PM while 11.5 represents 11:30 AM.
5. Define a predicate no_room_conflict(R1, T1, R2, T2) that returns true iff there is no conflict for room R1 to have class at time period T1 and for room R2 to have class at time period T2. Note that R1 and R2 may be the same room.

6. Define a predicate legal_schedule(X) that returns true iff X is a legal schedule considering the capacity and media type of room, the enollment, media requirement, and duration of the class.
7. Define a predicate check_conflict(S, L) that returns true iff S is not in conflict with any schedule in L.

Make sure the predicates can be used in the following test cases:

```
?- is_member(X, [1,2,3]).
X = 1 ;
X = 2 ;
X = 3 ;
false.

?- no_time_conflict(between(9, 10.5), between(11, 12)).
true .

?- no_time_conflict(between(11, 12), between(9, 10.5)).
true.

?- no_time_conflict(between(11, 12), between(11.5, 13)).
false.

?- has_instructor(C, I).
C = c1,
I = i1 ;
C = c2,
I = i1 ;
C = c3,
I = i2 ;
C = c4,
I = i2 ;
C = c5,
I = i3 ;
C = c6,
I = i3 ;
false.

?- no_instructor_conflict(i1, _, i2, _).
true.

?- no_instructor_conflict(i1, between(11, 12), i1, between(9, 10)).
true.

?- no_instructor_conflict(i1, between(11, 12), i1, between(11.5, 13)).
false.
```

```
?- no_room_conflict(r1, _, r2, _).
true

?- no_room_conflict(r1, between(11, 12), r1, between(9, 10)).
true.

?- no_room_conflict(r1, between(11, 12), r1, between(11.5, 13)).
false.

?- legal_schedule(schedule(c1, r3, between(9, 10))).
true.

?- legal_schedule(schedule(c2, r3, between(9, 10))).
false.

?- check_conflict(schedule(c1, r3, between(9, 10)),
     [schedule(c2, r1, between(14, 15.5)),
       schedule(c3, r2, between(14, 15.5)),
       schedule(c4, r3, between(10, 11)),
       schedule(c5, r1, between(11, 12.5)),
       schedule(c6, r2, between(9, 10.5))]
   ).
true .
```