Language Assignment #3: Prolog

Issued: Thursday, October 13
Due: Thursday, November 3

Purpose

This assignment asks you to begin using a logic programming language named Prolog. Prolog was designed by Alain Colmerauer, from Aix-Marseille University, in 1972.

Documentation

Prolog lecture slides are at:

```
~buff/classes/354/pub/slides/slides-prolog.pdf
```

Prolog is described in Section 12.2 of our textbook.

The onyx cluster has a Prolog compiler, which is documented at:

```
http://www.gprolog.org
```

and demonstrated by:

```
~buff/classes/354/pub/sum/prolog
```

You can also find various Prolog tutorials on the Internet, for example:

```
http://www.cpp.edu/~jrfisher/www/prolog_tutorial/contents.html
```

Assignment

Write and fully demonstrate a program that selects a set of acceptable meeting times for a set of people. Each person provides a set of free time slots. For example:

```
~buff/classes/354/pub/la3/data.pl
```

shows that Bob has two one-hour time slots that are free, one of which is 7:00AM–8:30AM.

Hints and Advice

Start with a simpler problem. A skeletal solution is:

```
~buff/classes/354/pub/la3/meetone.pl
```

When complete, this program will print the names of people who can meet from 8:30AM–8:45AM. This will be Ann, Carla, and Dave. If the "query" was for 5:30AM–6:45AM there would be no solutions.

Then, extend your solution to the complete problem. A skeletal solution is:

```
~buff/classes/354/pub/la3/meet.pl
```

When complete, this program will print a list of compatible meeting times for Ann, Bob, and Carla: 8:00AM–8:30AM and 10:00AM–10:15AM.

Given two times slots, there are three ways in which they can overlap. Draw pictures. You'll need a rule for each way. Use the `\==` predicate to avoid zero-length meetings.

You'll also need a predicate for comparing two times (e.g., `lte`).

Test your solution thoroughly, by modifying `data.pl` and the list of people who want to meet.