

# Exercise 5 Solutions

Due at 3:30 PM (on paper), April 16, 2009.

Please type your answers

---

The questions below refer to the following database:

```
:- dynamic pop/2.
```

```
capital(france,paris).  
capital(uk,london).  
capital(peru,lima).  
capital(chile,santiago).  
capital(japan,tokyo).  
capital(china,beijing).
```

```
europe(france).  
europe(uk).
```

```
samerica(peru).  
samerica(chile).
```

```
asia(japan).  
asia(china).
```

```
% population in millions
```

```
pop(paris,10).  
pop(london,8).  
pop(lima,8).  
pop(santiago,5).  
pop(tokyo,28).  
pop(beijing,12).
```

1. (5 pts) Write a goal to find the French capital.

```
capital(france,X).
```

2. (5 pts) Write a goal to find the country whose capital is Lima.

```
capital(X,lima).
```

3. (10 pts) Write a goal to find the capitals of Asian countries.

```
asia(X), capital(X,Y).
```

4. (10 pts) Write a goal to find cities with a population of at least 10 million.

```
pop(X,Y), Y >= 10.
```

5. (15 pts) Write a new rule that defines a sister city as a city with the same population as another city but on a different continent.

```
sister(S,T) :- pop(S,P), pop(T,P),  
               capital(X,S), capital(Y,T),  
               not((europe(X), europe(Y))),  
               not((samerica(X), samerica(Y))),  
               not((asia(X), asia(Y))).
```

*There are several variations on this that are acceptable. Hopefully writing this rule convinces you that the structure of your facts matters a lot. For example, consider if we had chosen to have a continent/2 functor instead of separate functors for each continent.*

6. (10 pts) Suppose a new census reveals that the population of Lima has increased to 9 million people. Write the two lines necessary to change the database at interpretation time to remove the old population fact and add the new one.

```
retract(pop(lima,8)).  
assert(pop(lima,9)).
```

*Note that for the retract statement, we don't have to specify the old population 8. We can put a variable there and have the interpreter unify it to 8. We can even use the placeholder “\_” in that spot.*

---

7. (20 pts) The following is small database from the book that we talked about in class:

```
rainy(seattle).  
rainy(rochester).  
cold(rochester).  
snowy(X) :- rainy(X), cold(X).
```

Load the database into the Prolog interpreter. Give the results for a trace of the goal `snowy(C)`. Explain the meaning of `call`, `exit`, `redo`, and `fail`.

```
[trace] ?- snowy(C).  
  Call: (7) snowy(_G335) ? creep  
  Call: (8) rainy(_G335) ? creep  
  Exit: (8) rainy(seattle) ? creep
```

```
Call: (8) cold(seattle) ? creep
Fail: (8) cold(seattle) ? creep
Redo: (8) rainy(_G335) ? creep
Exit: (8) rainy(rochester) ? creep
Call: (8) cold(rochester) ? creep
Exit: (8) cold(rochester) ? creep
Exit: (7) snowy(rochester) ? creep
C = rochester.
```

*For a each goal,*

*call* begins searching for clauses that unify with the goal

*exit* indicates the goal is satisfied, sets a place marker at the clause and binds the variables appropriately

*redo* retries the goal, unbinds the variables and resumes search at the place marker

*fail* indicates no more clauses match the goal

**8. (5 pts)** Examine the tic-tac-toe example on page 576 of the book. Suppose we change the rule

```
move(A) :- good(A), empty(A), !.
```

to read

```
move(A) :- good(A), empty(A).
```

What will be the effect of the change?

*Actually, there is no observable change as seen by those who actually tried it out in the interpreter. The call to move is followed by another cut in the makemove rule. If you also remove that cut, you will see the computer make a series of moves to finish out the game without giving the user a chance to make any more moves.*

**9. (20 pts)** Create your own database of at least 10 facts and/or rules. The database can be related to a topic of your choosing: sports, entertainment, literature, and hobbies. Show at least 3 interesting queries on your database.

Answers may vary (and often amuse).