

Prolog Lab Worksheet

This exercise uses Prolog which comes installed in your CSC481 VM.

Write a Prolog program by creating a file with you favorite **program** editor that contains the following facts:

```
female(pam).  
female(liz).  
female(ann).  
female(pat).  
male(tom).  
male(bob).  
male(jim).  
parent(pam,bob).  
parent(tom,bob).  
parent(tom,liz).  
parent(bob,ann).  
parent(bob,pat).  
parent(pat,jim).
```

Here the predicate `parent(X,Y)` means X is the parent of Y. In order to load this program into Prolog you will have to start Prolog and then either use the 'consult' menu point in the File menu (On Windows you can load the fact database with the menu point File→Consult.) or type:

```
?- consult('<your filename>').
```

when prompted (don't forget the period!).

Exercise 1

Once you have loaded the program pose the following queries:

```
?- female(ann).
```

```
?- female(jim).
```

```
?- parent(X,bob).
```

```
?- parent(tom,X).
```

```
?- parent(X,ann),parent(X,pat).
```

What are the answers to these queries? Beware, for some queries here might be more than one answer. To get all the answers type a ';' and carriage return at the question mark.

Exercise 2

Now, using the parent predicate formulate the following Prolog queries:

1. Who is Pat's parent?

2. Does Liz have a child?
3. Who is Pat's grandparent?

Exercise 3

Given the above facts, extend the program by writing rules defining the following predicates:

sister(X,Y) -- X is the sister of Y.

son(X,Y) -- X is the son of Y.

father(X,Y) -- X is the father of Y.

grandmother(X,Y) -- X is the grandmother of Y.

ancestor(X,Y) -- X is an ancestor of Y.

(Hint: this predicate might come in handy: different(X,Y):- not(X=Y). Some predicate definitions might be **recursive**.)

Demonstrate that your program works by posing the following queries:

4. ?- sister(X,pat).
5. ?- sister(X,Y).
6. ?- son(jim,X).
7. ?- father(X,bob).
8. ?- grandmother(X,ann).
9. ?- ancestor(X,jim).