## 

Establishing Facts

Prolog programs are text files. This means that we can write them in a text editor like Notepad. Launch Notepad a file to your area as family tree.pl You will have to make sure that you select All Files in the save dialog.

Type up the following facts for the family tree program.

/\* Family Tree Program \*/

/\* facts \*/

male(jack).

male(bill).

male(john).

female(dorothy).

female(evelyn).

female(grace).

female(liz).

parent(dorothy, evelyn).

parent(jack, evelyn).

parent(jack, grace).

parent(bill, liz).

parent(evelyn, liz).

parent(bill, john).

parent(evelyn, john).

The lines which begin and end with asterisks and forward slashes are comments - these are ignored when the program is run. All names must be in lower case - Prolog uses upper case for unknown information. All statements in Prolog end with a full stop.

Establishing Rules

Your program will need some rules if it is going to be used to make decisions. Add the following lines to the bottom of your program.

/\* rules \*/

mother(M,X):-

parent(M,X),

female(M).

father(F,X):-

parent(F,X),

male(F).

The first rule states that M is the mother of X if M is a parent of X and M is female. The father rule works in a similar way.

Testing The Program

Save your program. Launch Prolog.

Click on File, Consult and select the program that you have saved.

We will start by getting our program to find out who the parents of evelyn are. Type the following line in the Prolog window.

mother(X, evelyn).

In our statement, X stands for the information that we don't know. You should be given the answer X = dorothy. Press enter and type the following

father(X, evelyn).

You should get the answer X = jack. Use the program to find out the following.

The father of liz

The mother of john

The father of grace

Adding More Rules

Return to your program and add the following rules.

grandparent(G, X):-

parent(G,P),

parent(P,X).

sister(S,X):-

parent(Z,X),

parent(Z,S),

female(S),

not(S=X).

To get your program to use the new rules, first save your program. Now click on File and Reload Modified Files.

Type in the following

brother(X, liz).

You should get the answer X = john.

Now type in the following

grandparent(X, liz).

After the first answer, press the semi-colon key. You should get the second possible answer.

Using The Anonymous Variable

We can get our program to answer vaguer questions than the ones we have been putting. Suppose we want to find out who all the mothers in the system are. Type the following

mother(X, \_).

The underscore in Prolog is called the anonymous variable. This query should return all of the mothers in the system. Press the semi-colon after each answer.

Creating A User Interface

Expert systems can be made easier to use by creating an interface for the queries that you wish to run. Add the following to the bottom of your program.

findMother:-

write('Whose mother do you want to find? '),

read(Name),

mother(X,Name),

write('The mother of '), write(Name), write(' is '), write(X), nl.

To use this, type findMother. at the prompt. The program should help you with questions. Remember to place a full stop at the end of any line that you enter at the prompt.