CM0377 Lecture 3: Families program again! (family.pl)

```
parent_of(bill, jim).
parent_of(jane, jim).
parent_of(bill, albert).
parent_of(jane, albert).

parent_of(jim, alice).
parent_of(jane, alice).
parent_of(jim, fred).
parent_of(jane, fred).
parent_of(jim, john).
parent_of(jane, john).

male(bill).
male(jim).
male(albert).
male(fred).
male(john).

female(jane).
female(alice).

father_of(F, Ch):-
      parent_of(F, Ch), male(F).

grandfather_of(Gf, Gc):-
      parent_of(Gf, Ch), parent_of(Ch, Gc), male(Gf).

brother_of(B, Sib):-
      parent_of(P, B), parent_of(P, Sib), male(B), B\==Sib.

sister_of(S, Sib):-
      parent_of(P, S), parent_of(P, Sib), female(S), S\==Sib.
```

owns.pl

```
owns(bill, book('A.N. Other', 'The laws of confusion', 'Thomson', 1993)).
owns(bill, car('Vauxhall', 'Cavalier', 87000, 1992)).
owns(marion, book('A.C. Jones', 'Why I am almost perfect', 'Bluffem press',
1998)).

display_item(book(Auth, Title, Pub, Yr)):-
      write('a book: '), nl,
      write(Auth), nl, write('"'), write(Title),
      write('" Publisher: '), write(Pub), write(' Year: '), write(Yr), nl.

display_item(car(Manuf, Mod, Miles, Yr)):-
      write('a car: '), nl,
      write(Manuf), write(' '),
      write(Mod), write(' Mileage: '),
      write(Miles), write(' Year: '), write(Yr), nl, nl.

go:-
      owns(X, Item),
      write(X),
      write(' owns '),
      display_item(Item), fail.

go.
```

new_fam.pl:

```
parent_of(bill, list(jim, list(albert, empty))).
parent_of(jane, list(jim, list(albert, empty))).
parent_of(jim, list(alice, list(fred, list(john, empty)))).

male(bill).
male(jim).
male(albert).
male(fred).
male(john).

female(jane).
female(alice).

father_of(F, Ch):-
      parent_of(F, List), member(Ch, List), male(F).

grandfather_of(Gf, Gc):-
      parent_of(Gf, List1), member(Ch, List1),
      parent_of(Ch, List2), member(Gc, List2),
      male(Gf).

member(X, list(X, _)).
member(X, list(_, Others)):-member(X, Others).
```

new_fam1.pl:

```
parent_of(bill, [jim, albert]).
parent_of(jane, '.'(jim, '.'(albert, []))).
parent_of(jim, [alice, fred, john]).

male(bill).
male(jim).
male(albert).
male(fred).
male(john).

female(jane).
female(alice).

father_of(F, Ch):-
      parent_of(F, List), member(Ch, List), male(F).

grandfather_of(Gf, Gc):-
      parent_of(Gf, List1), member(Ch, List1),
      parent_of(Ch, List2), member(Gc, List2),
      male(Gf).

member(X, [X|_]).
member(X, [_|T]):-member(X, T).
```

Last modified 5th February 2001