Prolog Lab 2

University of Wales, Cardiff

This exercise is not assessed, but it is in your interest to practise this, because SOON there might be one that is.

1. Consider the following program :

```
parent_of(mary, peter).
parent_of(james, peter).
parent_of(mary, eunice).
parent_of(james, eunice).
parent_of(mary, lois).
parent_of(james, lois).
parent_of(marion, mary).
parent_of(timothy, mary).
parent_of(paul, timothy).
parent_of(marion, anna).
parent_of(jack, jill).
parent_of(mick, liam).
parent_of(clare, steve).

male(peter).
male(james).
male(timothy).
male(paul).
male(steve).
male(jack).
male(mick).
male(liam).

female(mary).
female(eunice).
female(lois).
female(marion).
female(anna).
female(clare).
female(jill).

mother_of(Mother, Person) :-
 parent_of(Mother, Person),
 female(Mother).
```

```
mother_of2(Person) :-
 parent_of(Mother, Person),
 female(Mother).

grandad_of(Granpa, Person) :-
 parent_of(Parent, Person),
 parent_of(Granpa, Parent),
 male(Granpa).
```

1. Who is mary's mother, and eunice's grandfather.
2. Modify the *grandad_of* relation for a *grandma_of* relation.
3. What is the difference between the two mother_of relations.
4. Write predicates and clauses for the following :
   child_of(Child, Person)
   which succeeds if **Child** is a son of **Person**.

   son_of(Son, Person)
   which succeeds if **Son** is a son of **Person**.

   brother_of(Brother, Person)
   which succeeds if **Brother** is a brother of a **Person**
5. Can you draw the tree to show how backtracking occurs in this case ? Also, how are multiple solutions obtained ? A diagram to show the workings of this program would be useful to visualise program behaviour.

Note: the facts within the database may have to be expanded for some of the relations to be written.

Omer Rana
February 1997.