Part I
Using the SWI-Prolog or a prolog compiler/interpreter of your choice, design and implement a
deductive database system for exploring direct and indirect family relationships. As discussed in class,
logic programming in Prolog is a declarative rather than procedural process. You simply express
what you know in terms of facts or "assertions" and rules or "relationships", in other words you
declare what the problem is and what the desired solution should be like. You don't provide an exact
and step by step solution. Your program uses the facts and built in deductive rules that you have
provided, along with the backtracking, resolution and unification facilities built into the language to
search the database and provide a solution. Your program should respond to the following
assertions, relations and queries:

Assertions:

female(name).
male(name).
born(name,year).
marrried(name,name).
father(father_name,child_name).
mother(mother_name,child_name). (or implicitly defined).

Relations:

mother(X,Y) :- ....
parent(X,Y) :- ....
sisters(X,Y) :- ..... /* Two Sisters */
brothers(X,Y) :- ..... /* Two Brothers */
sibling(X,Y) :- ..... 
grand_father(X,Y) :- ..... 
grand_mother(X,Y) :- ..... 
cousin(X,Y) :- ..... 
uncle(X,Y) :- ..... 
aunt(X,Y) :- ..... 
generational_gap(X,Y) :- ..... /* defined as 20 years age difference */
teenager(X) :- ..... 
senior_citizen(X) :- ..... 
any_relationship(X,Y) :- .....
Queries:

Any of the above assertions or relationships can be specified in the form of a query. Queries can be of two forms, logical (returning true or false) or queries that return one or more values.

What to hand in?

1) Program listing. (fully documented)

2) Program output. Fully test your program for direct and indirect relationships. Note any surprising or interesting family relationships that are deduced as the result of the inferencing mechanism that you have developed. (your output may be captured using the script command.)

3) A neatly drawn family tree.

4) Be prepared to demonstrate your program.

5) Staple your paper. (No loose papers)

Extra Credit: (2 pts.)

Explore the possibility of automatic assertions and rule generations. Note that this is an effective means of improving the performance of your database. (i.e. when a new fact or rule is deduced through inferencing, resolution, unification and backtracking, one may choose to add it to the existing knowledgebase.)

Part II: ABA