%parent(X, Y) holds true when X is a parent of Y

parent(tom, jim);

parent(tom, liz);

parent(jim, bill);

parent(liz, anne);

parent(liz, bill);

parent(mary, jim);

parent(mary, liz);

parent(valerie, bill);

%male(X) holds true is X is male

male(jim);

male(tom);

male(bill);

%female(X) holds true is X is female

female(anne);

female(liz);

female(mary);

female(valerie);

% mother(M,C) holds when M is the C’s mother: a mother is a female parent

mother(M,C):- parent(M,C), female(M);

%father(F,C) holds when F is C’s father: a father is a male parent

father(F,C):-parent(F,C), male(F);

%offspring(C,P) holds when C is P’s offspring, which means that P is C’s parent

offspring(C,P):-parent(P,C);

%parentS(M, F, C) holds when M and F are two different parents of C

parentS(M,F,C):-parent(M,C), parent(F,C), notequal(M,F);

% notequal fails (it is not true) when the arguments are the same; otherwise it succeeds (i.e., it is true)

notequal(X,X):-!, fail;

notequal(\_, \_);

%siblings(X, Y) is true when X and Y have the same parents

siblings(X, Y) :- parentS(M,F,X), parentS(M,F, Y), notequal(X,Y);

%sisters(X,Y): sisters are female siblings

sisters(X, Y) :- female(X), female(Y), siblings(X,Y);

%brothers(X,Y): male siblings

brother(X,Y) :- male(X), male(Y), siblings(X, Y);

%grandparent(G, X): G parent of a parent of X

grandparent(G, X) :- parent(G, P), parent(P, X);

%grandmother(GM, X) : female grandparent

grandmother(GM, X) :- grandparent(GM,X), female(GM);