%CS_125 Logic Programming
%Spring 2002
%Solutions to Coursework 1

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Question 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%(a)
possible_suspect(fred).
possible_suspect(mary).
possible_suspect(jane).
possible_suspect(george).
%(b)
crime(robbery1, john, tuesday, park).
crime(assault1, mary, wednesday, park).
crime(robbery2, jim, wednesday, pub).
crime(assault2, robin, thursday, park).
%(c)
was_at(fred, park, tuesday).
was_at(fred, pub, wednesday).
was_at(fred, pub, thursday).
was_at(george, pub, tuesday).
was_at(george, pub, wednesday).
was_at(george, home, thursday).
was_at(jane, home, tuesday).
was_at(jane, park, wednesday).
was_at(jane, park, thursday).
was_at(mary, pub, tuesday).
was_at(mary, park, wednesday).
was_at(mary, home, thursday).
%(d)
jealous_of(fred, john).
jealous_of(jane, mary).
owes_money_to(george, jim).
ownes_money_to(mary, robin).
%(e)
motive(P, V) :- jealous_of(P, V).
motive(P, V) :- owes_money_to(P, V).
prime_suspect(P, C) :-
possible_suspect(P),
crime(C, V, D, L),
was_at(P, L, D),
motive(P, V).
%(f) Queries and answers.
/*
?- consult('lp-cwisol.txt').
% lp-cwisol.txt compiled 0.00 sec, 10,136 bytes
Yes
?- prime_suspect(fred, robbery1).
Yes
?- prime_suspect(george, robbery2).
Yes
?- prime_suspect(P, robbery1).
P = fred ;
No
?- prime_suspect(P, robbery2).
P = george ;
No
?- prime_suspect(P, assault1).
P = jane ;
No
?- prime_suspect(P, assault2).
No
?- prime_suspect(fred, C).
C = robbery1 ;
No
?- prime_suspect(george, C).
C = robbery2 ;
*/

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Question 2
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%(a)
family(L) :-
L = [person (anne,f,[paul,peter]),
     person (ben,m,[harry,robert]),
     person (diana,f,[harry,robert]),
     person (edward,m,[{}]),
     person (elizabeth,f,[anne,ben,edward,homer]),
     person (george,m,[elizabeth]),
     person (harry,m,[{}]),
     person (homer,m,[beatrice,patrick]),
     person (jill,f,[beatrice]),
     person (mark,m,[paul,peter]),
     person (mary,f,[elizabeth]),
     person (phillip,m,[ben,edward,homer]),
     person (paul,m,[{}]),
     person (patrick,m,[{}]),
     person (peter,m,[{}]),
     person (robert,m,[{}]).
(b)

parent(X,Y) :-
  family(L),
  member(person(X,_),Children,L),
  member(Y,Children).

female(X) :- family(L), member(person(X,f_),L).

male(X) :- family(L), member(person(X,m_),L).

mother(X,Y) :- parent(X,Y), female(X).

father(X,Y) :- parent(X,Y), male(X).

grandparent(X,Y) :- parent(X,Z), parent(Z,Y).

ancestor(X,Y) :- parent(X,Y).

ancestor(X,Y) :- parent(X,Z), ancestor(Z,Y).

(c) Queries and answers

/*

?- mother(X,robert).
X = diana ;
No

?- parent(anne,X), parent(mark,X).
X = paul ;
X = peter ;
No

?- grandparent(X,ben).
X = george ;
X = mary ;
No

?- ancestor(X,harry).
X = ben ;
X = diana ;
X = elisabeth ;
X = george ;
X = mary ;
X = phillip ;
No
*/

chalkboard(L) :-
  L = [pizza(hawaii, HT1, HT2, HP),
       pizza(marco_polo, MT1, MT2, NP),
       pizza(pepperoni, PT1, PT2, PP),
       pizza(supper_supreme, ST1, ST2, SP),
       pizza(ninja, NT1, NT2, NP)],
  HT1 = mussels,
  member(HP,[’7.00’, ’8.50’, ’10.00’]),
  MT2 = tomato,
  member(HT1,[mussels,prawns,salami,tuna]),
  member(pizza(_,ham,_,’8.50’),L),
  PP = ’7.00’,
  member(ST2,[avocado,corn,olive,tomato]),
  member(pizza(_,tuna,corn,F),L),
  member(P,[’5.00’, ’7.00’, ’8.50’, ’10.00’]),
  member(pizza(_,T1,olive,’5.00’),L),
  member(T1,[ham,mussels,prawns,tuna]),
  member(pizza(_,_,pineapple,Q),L),
  member(Q,[’5.00’, ’6.50’, ’7.00’, ’8.50’]),
  permutation([ham,mussels,prawns,salami,tuna],
              [HT1,MT1,PT1,ST1,NT1]),
  permutation([avocado,corn,olive,prawns,tomato],
              [HT2,MT2,PT2,ST2,NT2]),
  permutation([’5.00’, ’6.50’, ’7.00’, ’8.50’, ’10.00’],
              [HP,MP,PP,SP,NP]).

permutation([],[]).

permutation(L,[X|P]) :- sel(X,L,L1), permutation(L1,P).

sel(X,[X]|L),L.

sel(X,[Y]|L),[Y|L1]) :- sel(X,L,L1).

/*

%- chalkboard([P1,P2,P3,P4,P5]).

P1 = pizza(hawaii, mussels, avocado, ’10.00’)
P2 = pizza(marco_polo, salami, tomato, ’6.50’)
P3 = pizza(pepperoni, tuna, corn, ’7.00’)
P4 = pizza(supper_supreme, prawns, olive, ’5.00’)
P5 = pizza(ninja, ham, pineapple, ’8.50’);
No
*/