**Practical No : 03**

**Problem Statement:**

1) Write a Prolog Program for basic relationship rules. (slide No-5)

2) Write a Prolog Program to add relations - Uncle, Aunt, Nephew, Niece,

grandfather, Grandmother, Grand Child, Cousin in previous program.

3) Write a Prolog Program for recursive rule for relationship ( slide No-8,9)

4) Write a Prolog Program for recursive rule for directed graph (slide No- 11)

**1) Write a Prolog Program for basic relationship rules. (slide No-5)**

**Code:-**

parent(elizabeth,charles).

parent(philip,charles).

parent(elizabeth,anne).

parent(philip,anne).

parent(elizabeth,andrew).

parent(philip,andrew).

parent(diana,william).

parent(charles,william).

parent(diana,harry).

parent(charles,harry).

female(elizabeth).

male(philip).

male(charles).

female(anne).

male(andrew).

female(diana).

=>

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

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

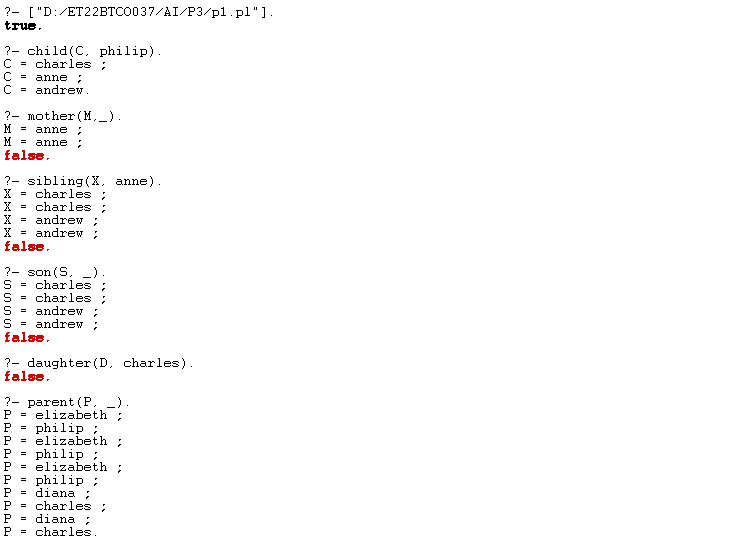
father(X,Y) :- parent(Y,X),male(X).

son(X,Y) :- child(X,Y),male(X).

daughter(X,Y) :- child(X,Y),female(X).

sibling(X,Y) :- parent(Z,X),parent(Z,Y),X\=Y.

**Output:-**



**2) Write a Prolog Program to add relations - Uncle, Aunt, Nephew, Niece, grandfather, Grandmother, Grand Child, Cousin in previous program**

**Code:-**

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

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

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

son(X,Y) :- child(X,Y), male(X).

daughter(X,Y) :- child(X,Y), female(X).

sibling(X,Y) :- parent(Z,X), parent(Z,Y),female(Z),not(X=Y).

brother(X,Y):-male(X),sibling(X,Y).

sister(X,Y):-female(Y),sibling(X,Y).

uncle(X,Y):-parent(Z,Y),

sibling(X,Z),male(X).

aunt(X,Y):-parent(Z,Y),sibling(X,Z),female(X).

nephew(X,Y):-male(X),(uncle(Y,X);aunt(Y,X)).

niece(X,Y):-female(X),(uncle(Y,X);aunt(Y,X)).

**Output:-**



**3) Write a Prolog Program for recursive rule for relationship ( slide No-8,9)**

**Code:-**

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

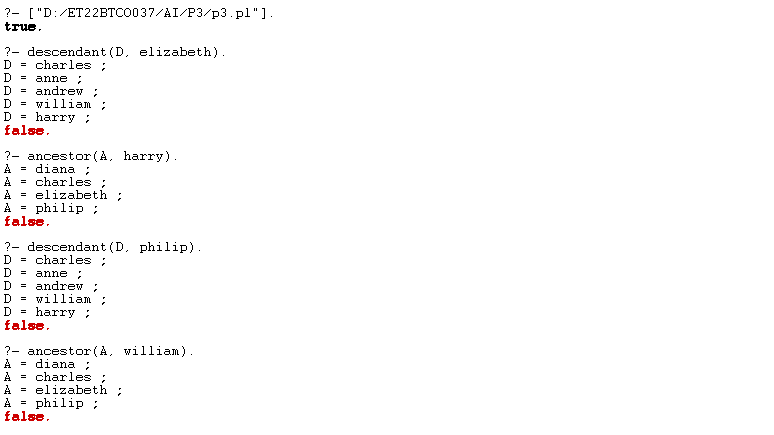
descendant(X, Y) :- child(X, Y).

descendant(X, Y) :- child(X, Z), descendant(Z, Y).

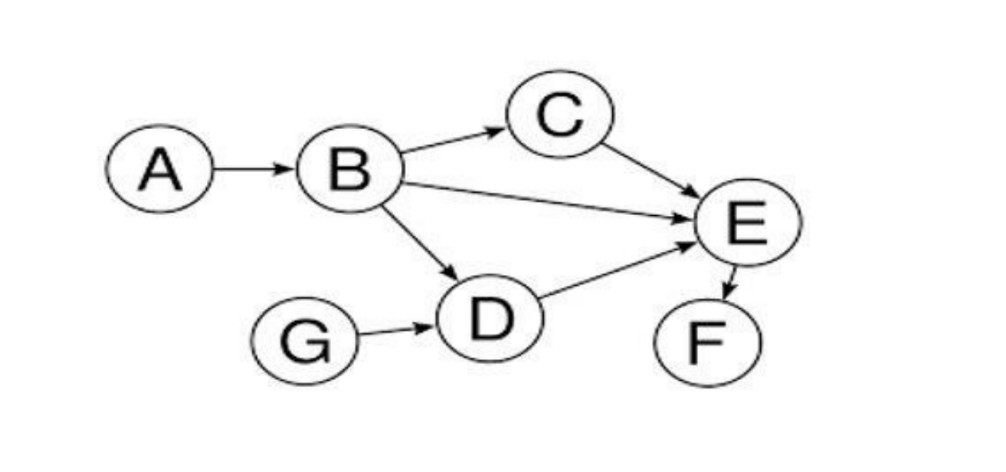
ancestor(X, Y) :- parent(X, Y).

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

**Output:-**



**4) Write a Prolog Program for recursive rule for directed graph (slide No- 11)**



**Code:-**

edge(a,b).

edge(b,c).

edge(b,d).

edge(b,e).

edge(c,e).

edge(d,e).

edge(e,f).

edge(g,d).

connected(X,Y) :- edge(X,Y).

connected(X,Z):- edge(X,Y),connected(Y,Z).

**Output:-**

