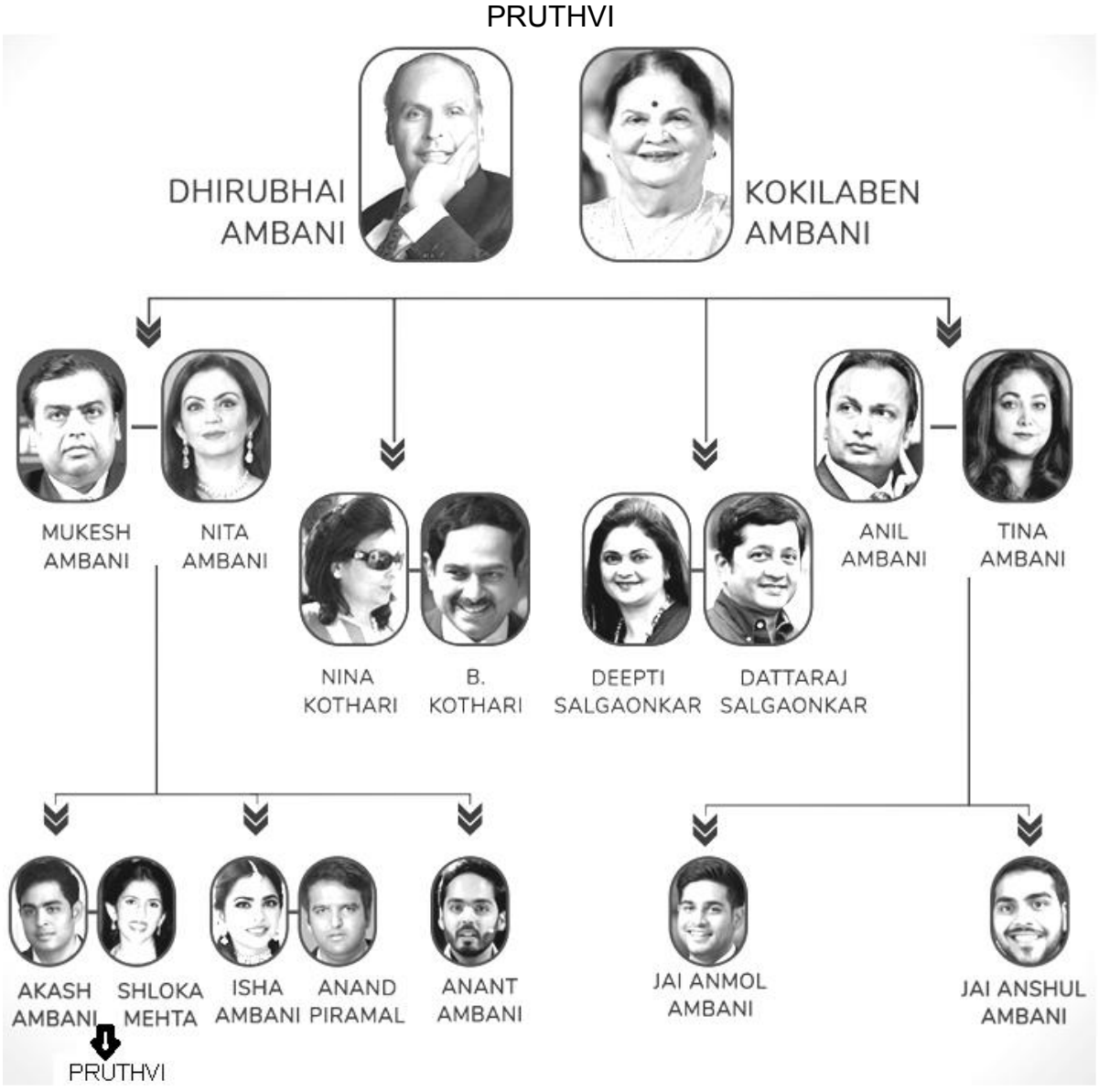
M.C.A Semester – II

Roll No :	**1233**
Name of Student :	**Kirti vichare**
Subject :	MCAL21 Artificial Intelligence and Machine Learning Lab
Module:	Module1 : Logic programming with Prolog
Date:	05/05/2022

Q. 1) Write a PROLOG program using following family tree/knowledge base. Write rules for **grandfather, grandmother, sister, cousin, brother, uncle**, **aunty** and **sibling** relations and query the database.

sister(X,Y) Sister of X is Y , brother(X,Y) brother of X is Y, sibling(X,Y) sibling of X is Y

uncle(X,Y) uncle of X is Y, aunty(X,Y) aunty of X is Y, cousin(X,Y) cousin of X is Y



The knowledge base is as follows:

```
male(dhirubhai).
male(mukesh).
male(anil).
male(kothari).
male(dattaraj).
male(akash).
male(anand).
male(anant).
male(jaianmol).
male(jaianshul).
male(pruthvi).
female(kokilaben).
female(nita).
female(nina).
female(deepti).
female(tina).
female(shloka).
female(isha).

fatherof(mukesh,dhirubhai).
fatherof(nina,dhirubhai).
fatherof(deepti,dhirubhai).
fatherof(anil,dhirubhai).
fatherof(akash,mukesh).
fatherof(isha,mukesh).
fatherof(anant,mukesh).
fatherof(jaianmol,anil).
fatherof(jaianshul,anil).
fatherof(pruthvi,akash).

motherof(mukesh,kokilaben).
motherof(nina,kokilaben).
motherof(deepti,kokilaben).
motherof(anil,kokilaben).
motherof(akash,nita).
motherof(isha,nita).
motherof(anant,nita).
motherof(jaianmol,tina).
motherof(jaianshul,tina).
motherof(pruthvi,shloka).

wifeof(dhirubhai,kokilaben).
wifeof(mukesh,nita).
wifeof(anil,tina).
wifeof(kothari,nina).
wifeof(dattaraj,deepti).
wifeof(akash,shloka).
```

wifeof(anand,isha).

Source code:

```
grandfather(X,Y):-
        fatherof(X,Z),
        fatherof(Z,Y),
        male(Y).

grandmother(X,Y):-
        fatherof(X,Z),
        motherof(Z,Y),
        female(Y).

sister(X,Y):-
        fatherof(X,Z),
        fatherof(Y,Z),
        female(Y).

sibling(X,Y):-
        fatherof(X,A),
        fatherof(Y,A),
        X \= Y.

cousin(X,Y):-
        fatherof(X,Z),
        fatherof(Y,Q),
        Z \= Q,
        fatherof(Z,P),
        fatherof(Q,P).

brother(X,Y):-
        fatherof(X,Z),
        fatherof(Y,Z),
        male(Y).
```

Output :

1.Grandfather

```
compiling D:/AI_1211/ambani.pl for byte code...
D:/AI_1211/ambani.pl compiled, 79 lines read - 6661 bytes written, 11 ms
| ?- grandfather(akash,dhirubhai).

yes
| ?- grandfather(dhirubhai,akash).

no
| ?-
```

2.Grandmother

```
| ?- grandmother(akash,kokilaben).

yes
| ?- grandmother(dhirubhai,kokilaben).

no
| ?-
```

3.Sister

```
| ?- sister(nina,deepti).

yes
| ?- sister(nina,mukesh).

no
| ?-
```

4.Sibling

```
| ?- sibling(mukesh,anil).

yes
| ?- sibling(mukesh,dhirubhai).

no
| ?-
```

5. Cousin

```
| ?- cousin(akash,jaianmol).

yes
| ?- cousin(akash,jaianshul).

yes
| ?- cousin(akash,nita).

no
| ?-
```

6.Brother

```
| ?- brother(mukesh,anil).

yes
| ?- brother(mukesh,dhirubhai).

no
| ?- |
```

Q. 2) Write a PROLOG program to implement Simple calculator.
Source code:

```
calc(X,Y):-
        A is X+Y,
        write('Addition is : '),write(A),nl,
        B is X-Y,
        write('Subtraction is : '),write(B),nl,
        C is X*Y,
        write('Multiplication is : '),write(C),nl,
        D is X/Y,
        write('Division is : '),write(D),nl,
        E is X**Y,
        write('Power is : '),write(E),nl,
        F is X//Y,
        write('Integer Division is : '),write(F),nl,
        G is X mod Y,
        write('Modulus is : '),write(G).
```

Output :

```
yes
| ?- [calc].
compiling D:/AI_1211/calc.pl for byte code...
D:/AI_1211/calc.pl compiled, 15 lines read - 2488 bytes written, 21 ms

yes
| ?- calc(15,32).
Addition is : 47
Subtraction is : -17
Multiplication is : 480
Division is : 0.46875
Power is : 4.3143988327398921e+37
Integer Division is : 0
Modulus is : 15

yes
| ?-
```

Q. 3) Write a PROLOG program to display numbers from 1 to N.
Source code:

```
count(N,N):-
        write(N).

count(X,N):-
        X < N,
        write(X),
        X1 is X + 1,
        count(X1,N).
```

Output :

```
| ?- [count].
compiling D:/AI_1211/count.pl for byte code...
D:/AI_1211/count.pl compiled, 7 lines read - 736 bytes written, 9 ms

(16 ms) yes
| ?- count(1,5).
12345

true ? |
```

Q. 4) Write a PROLOG program to find sum of numbers in the list.
Source code:

```
sumlist([],0).
sumlist([Head|Tail],S):-
        sumlist(Tail,Z),
        S is Head + Z.
```

Output :

```
| ?- sumlist([1,2,3,4,5,6,7,8,9,0],S).

S = 45

yes
| ?- |
```

Q. 5) Write a PROLOG program to find maximum and minimum number from the given list.

Source code:

```
maxlist([],0).
maxlist([Head|Tail],Max) :-
   maxlist(Tail,TailMax),
   Head > TailMax,
   Max is Head.

maxlist([Head|Tail],Max) :-
   maxlist(Tail,TailMax),
   Head =< TailMax,
   Max is TailMax.


minlist([Min],Min).
minlist([Head,X|Tail],M) :-
   Head =< X,
   minlist([Head|Tail],M).

minlist([Head,X|Tail],M) :-
   Head > X,
   minlist([X|Tail],M).
```

Output :

```
| ?- maxlist([5,3,4,8,6],X).

X = 8 ?

yes
| ?- maxlist([55,32,40,18,62],X).

X = 62 ?

yes
| ?- minlist([12,15,45,32,65],X).

X = 12 ?

yes
| ?- minlist([125,165,450,23,515],X).

X = 23 ?

yes
| ?- |
```

Q.6) Write a PROLOG program to reverse the given list.

Source code:

```
reversel([],[]).
reversel([X|Y],Z):-
        reversel(Y,W),
        appendl(W,[X],Z).

appendl([],X,X).
appendl([X|Y],Z,[X|W]):-
        appendl(Y,Z,W).
```

Output :

```
| ?- reversel([a,b,c,d,e],X).

X = [e,d,c,b,a]

yes
| ?- reversel([1,2,3,4,5],X).

X = [5,4,3,2,1]

yes
| ?- appendl([a,b,c,d,e],X,Q).

Q = [a,b,c,d,e|X]

yes
| ?- appendl([a,b,c,d,e],f,Q).

Q = [a,b,c,d,e|f]

yes
| ?- appendl([a,b,c,d,e],fg,Q).

Q = [a,b,c,d,e|fg]

yes
| ?-
```

Q.7) Write a PROLOG program to reverse the given list.

Source code:

```
reversel([],[]).
reversel([X|Y],Z):-
        reversel(Y,W),
        appendl(W,[X],Z).

appendl([],X,X).
appendl([X|Y],Z,[X|W]):-
        appendl(Y,Z,W).
```

Output :

```
| ?- reversel([a,b,c,d,e],X).

X = [e,d,c,b,a]

yes
| ?- reversel([1,2,3,4,5],X).

X = [5,4,3,2,1]

yes
| ?- appendl([a,b,c,d,e],X,Q).

Q = [a,b,c,d,e|X]

yes
| ?- appendl([a,b,c,d,e],f,Q).

Q = [a,b,c,d,e|f]

yes
| ?- appendl([a,b,c,d,e],fg,Q).

Q = [a,b,c,d,e|fg]

yes
| ?-
```

Q.8) Write a PROLOG program to solve the Monkey Banana problem.
Imagine a room containing a monkey, chair and some bananas. Bananas have been hanged from the center of ceiling. If the monkey is clever enough he can reach the bananas by placing the chair directly below the bananas and climb on the chair. The problem is to prove the monkey can reach the bananas.

Source code:

```
in_room(banana).
in_room(chair).
in_room(monkey).
clever(monkey).
can_climb(monkey,chair).
tall(chair).
can_move(monkey,chair,banana).

can_reach(X, Y):-
        clever(X),
        near(X,Y).

get_on(X,Y):-
        can_climb(X,Y).

under(Y,Z):-
        in_room(X),in_room(Y),
        in_room(Z),can_move(X,Y,Z).

near(X,Z):-
        get_on(X,Y),
        under(Y,Z);
        tall(Y).
```

Output:

```
| ?- can_reach(monkey,chair).

yes
| ?- can_reach(monkey,banana).

true ?

yes
| ?- get_on(monkey,banana).

no
| ?- get_on(monkey,chair).

yes
| ?- under(monkey,chair).

no
| ?- near(monkey,chair).

yes
| ?- near(monkey,banana).

true ?

yes
| ?-
```

Q.9) Write a PROLOG program to implement graph.

The graph data is represented as follows

```
edge(a, c).
edge(a, b).
edge(b, d).
edge(c, d).
edge(c, f).
edge(d, e).
edge(f, g).
edge(k, i).
edge(g, h).
edge(i, j).
edge(j, k).
edge(e, c).
```

Write predicate to check two nodes are connected or not?

connected(Node1,Node2) is a predicate which will check two nodes Node1 and Node2 are directly connected or indirectly connected. **path(X,Y)** predicate checks, is there path between node X and node Y. **cycle(X)** predicate check whether cycle exists between node X to node X. e.g. node e to e cycle exist.

edge(a,c) predicate indicates that the edge exist from a to c.

Source code:

```
edge(a, c).
edge(a, b).
edge(b, d).
edge(c, d).
edge(c, f).
edge(d, e).
edge(f, g).
edge(k, i).
edge(g, h).
edge(i, j).
edge(j, k).
edge(e, c).

connected(Node1,Node2):-
        edge(Node1,Node2).
connected(Node1,Node2):-
        edge(Node1,X),
        connected(X,Node2).
```

Output:

```
| ?- connected(a,b).

true ?

yes
| ?- edge(d,a).

no
| ?- connected(c,e).

true ?

yes
| ?- |
```

Q.10) Write a PROLOG program to display travelling distance.

The knowledge base given is as follows

```
road(mumbai,indore,601).
road(indore,gwalior,505).
road(gwalior,agra,121).
road(agra,delhi,222).
road(mumbai,surat,300).
road(surat,vadodara,154).
road(vadodara,jodhpur,554).
road(jodhpur,jaipur,357).
```

Write a predicate route(Town1,Town2,Distance), which will find distance between Town1 and Town2.

The route may be direct or indirect(i.e. visiting different towns and reaching to goal)

If the goal is route(mumbai,indore,Disatnce) then it should display Distance=601.

If the goal is route(indore,delhi,Disatnce) then it should display Distance=848 (i.e. 505+121+222=848)

Source code:

```
road(mumbai,indore,601).
road(indore,gwalior,505).
road(gwalior,agra,121).
road(agra,delhi,222).
road(mumbai,surat,300).
road(surat,vadodara,154).
road(vadodara,jodhpur,554).
road(jodhpur,jaipur,357).



route(X,Y,Distance):-
        road(X,Y,Distance),
        write('Distance Between '),write(X),write(' and '),write(Y),write(' is '),
        write(Distance),write(' km.'),nl.
route(X,Y,Distance):-
        road(X,Z,Distance1),
        route(Z,Y,Distance2),
        Distance is Distance1+Distance2,!,
        write('Cumulative Distance Between '),write(X),write(' and '),write(Y),write(' is '),
        write(Distance),write(' km.'),nl.
```

Output :

```
| ?- route(mumbai,indore,Distance).
Distance Between mumbai and indore is 601 km.

Distance = 601 ? ;

no
| ?-
route(delhi,indore,Distance).

no
| ?- route(indore,delhi,Distance).
Distance Between agra and delhi is 222 km.
Cumulative Distance Between gwalior and delhi is 343 km.
Cumulative Distance Between indore and delhi is 848 km.

Distance = 848

yes
| ?- route(mumbai,indore,Distance).
Distance Between mumbai and indore is 601 km.

Distance = 601 ?

yes
| ?- route(mumbai,gwalior,Distance).
Distance Between indore and gwalior is 505 km.
Cumulative Distance Between mumbai and gwalior is 1106 km.

Distance = 1106

yes
| ?-
```