Prolog Programming Assignment

1. How does the query in kb.pl file are executed?

→

Code : % knowledge bases

```
lovers (vincent, mia)
lovers (marcellus, mia)
lovers (pumpkin, honey-bunny)
lovers (honey bunny, pumpkin).

jealous (x,y):-
    lovers (x,z),
    lovers (y,z).
```

Query : lovers (x, mia)
O/P   :   x = vincent
            x = marcellus

Explanation: Here as we know vincent lovers mia aswell as marcellus lovers mia. Thus the kb assumes that x is either vincent or marcellus

Query 2 : ?- jealous (x,y)
            x = y, x = vincent
            x = vincent
            Y = Marcellus

K.G.C.E.
Karjat - Raigad

Page No. :
Date :

KGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCE

X : Marcellus
X = Y , Y = Marcellus
X = Y, Y = Pumpkin
X = Y, Y = Honey - bunny

Explanation :

As there is no fixed parameter in our query. The query will produce output of every jealous (x, y) pair on our prolog code. The jealous () rule follows :

jelous (x,y) : lovers (x, z), lover (y, z).

Initially , x, & y both were associated to vincent i.e, self association it then follows reflexive property for the rest of the prolog code.

2] How does the queries in list . Pt file are executed ?

Code: suffix (xs, ys)
append (- , ys, xs)

Prefix (xs, ys) :-
append (- ys, xs)

sublist (xs, ys) :-
suffix (xs, zs)
preffix (zs, ys)

nrev ([], [ ])
nrev ([H|T∅], L) :-
nrev (T∅, T).
append (T, [H], L].

Query 1 : ? . sublist ([a, b, c, d, e], [c, d]).
output : True.

Explanation : · A sublist procedure looks for a match between the first elements of the sublist & the main-list . Here [c, d] is the sublist of the main list [a, b, c, d, e]. As the main list containts the sublist [c, d] the output is true else, the output would have been false.

K.G.C.E.
Karjat - Raigad

Page No. :

Date :

KGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCE

Query 2 : ? suffix ([a,b,c], zs).

Output : zs [abc]
zs = [b, c]
zs = [c]
zs = [ ]

Explanation : Suffix in general eliminates the front elements from a list, Here, by using suffix procedure, [a,b,c] elements are removed from a & continues until all the elements are removed As there are no more elements in the list, the output will be displayed as false

Q3) Programming create a prolog code to find factorial of a number

→ Code : factorial (0, 1)
factorial (N, F) :
N > 0,
N₁ is N-1
factorial (N, F₁)
N is N * F₁

Query : ? . factorial (3, w)

Output : w = 6

K.G.C.E.
Karjat - Raigad

Page No. :

Date :

KGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCEKGCE

Q4 In example data set movies. Pl write query strings & results of query execution for any 'OR 5 tasks:

a) In which year was the movie american Beauty released ?

Query : ? - movie (American, beauty, y)

output : y = 1999

b) find the movies released in year 2000

Query : ? - movie (M, 2000)
Output : M = down-from-the-mountain
M = O-brother-where-art-thou
M = ghost-world

c) find movies released before 2000

Query : ? - movie (M, y), y < 2000

Output : M = american-beauty
Y = 1999

M = anna
Y = 1987

M = barton-fink
Y = 1991

d) Find the movies released after 1990

Query: ? - movie (M, Y) $Y \le 1990$

output: M = american_beauty.
                Y = 1999

              M = barton-fink.
                Y = 1991

e) find a director of a movie in which Scalett
Johansson appeared.

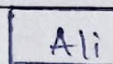Query : ? - actress (M; scarlett-Johansson)-)
                director (M, O)

Output: O - Peter-webber
                M = girl-with-a-pearl-earring.

Q5) Draw a family tree of you / any arbitrary
family, which has the following relations
mother, father, daughter, son, grandson,
grandmother, sibling, uncle, person, male, female
you need to convert it into KB & write
atleast 6 queries & Query result on your
KB

→ Diagram :

Family Tree

Query 1 : ? mother - of (x, jess)

Output : x = helen

Query 2 : ? parent - of (x, simon).

Output : x = jess

Query 3 : ? - sister-of (x, lily)

output : x = jess

Query 4 : ? parent - of (x, harry).

output : x = lily
        x = james.

Query 5 : ? aunt-of (x, simon)

Output : x = lily

Query 6 : ? grandfather-of (x, harry)

Output : x = jack.