Name: Vatsa Nagaria
UID: 2019130041
TE COMPS
Batch: C
Date: 22/11/21

EXPERIMENT 6

(Prolog)

Aim: To perform given problems using prolog language.

Code:

1. Family tree -

```
parent(ratan, vinod).
parent(ratan, jyoti).
parent(ratan, lata).
parent(mulji, vinod).
parent(mulji, jyoti).
parent(mulji, lata).
parent(vinod, vatsa).
parent(sejal, vatsa).
parent(vinod, rishi).
parent(sejal, rishi).

female(ratan).
female(jyoti).
female(sejal).
female(lata).
male(vatsa).
male(vinod).
male(mulji).
male(rishi).

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

son(X, Y):- parent(Y, X), male(X).
daughter(X, Y):- parent(Y, X), female(X).

grandfather(X, Y):- parent(X, A), parent(A, Y), male(X).
grandmother(X, Y):- parent(X, A), parent(A, Y), female(X).
```

```
sister(X, Y):- parent(A, X), parent(A, Y), female(X), X \= Y.
brother(X, Y):- parent(A, X), parent(A, Y), male(X), X \= Y.

aunt(X, Y):- sister(X, Z), parent(Z, Y).
uncle(X, Y):- brother(X, Z), parent(Z, Y).

predecessor(X, Y) :- parent(X, Y).
predecessor(X, Y) :- parent(X, A),predecessor(A, Y).

successor(X, Y):- son(Y, X).
successor(X, Y):- daughter(Y, X).
successor(X, Y):- son(A, X),successor(A, Y).
successor(X, Y):- daughter(A, X),successor(A, Y).
```

2. Removing duplicates -

```
rle([],[]).
rle([X],[X]).
rle([X, X|REMAINING],OUTPUT) :- rle([X|REMAINING],OUTPUT).
rle([X, Y|REMAINING], [X|OUTPUT_TAIL]) :- X \= Y, rle([Y|REMAINING],OUTPUT_TAIL).
```

3. Slice -

```
slice([X|_], 1, 1, [X]).
slice([X|TAIL], 1, CURRENT_INDEX, [X|REM_TAIL]) :- CURRENT_INDEX >1,NEXT_INDEX is
CURRENT_INDEX - 1, slice(TAIL, 1, NEXT_INDEX,REM_TAIL).
slice([_|TAIL], I, CURRENT_INDEX, OUTPUT) :- I > 1,I1 is I - 1, NEXT_INDEX is
CURRENT_INDEX - 1, slice(TAIL, I1,NEXT_INDEX, OUTPUT).
```

4. Subsets -

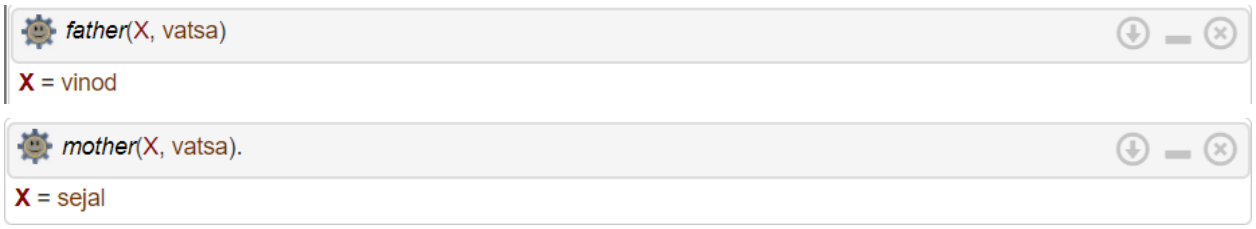
```
el(X,[X|L],L).
el(X,[_|L],R) :- el(X,L,R).
selectN(0,_,[]) :- !.
selectN(N,L,[X|S]) :- N > 0,el(X,L,R),N1 is N-1,selectN(N1,R,S).
subsets([],[],[],[]).
subsets(G,[N1|Ns],[G1|Gs],[]) :-selectN(N1,G,G1),subtract(G,G1,R),subsets(R,Ns,Gs,[]).
```

5. Huffman coding -

```
huffman(Fs,Cs) :-
  initialize(Fs,Ns),
  create_tree(Ns,T),
```

```
   traverse_tree(T,Cs).

initialize(Fs,Ns) :-
    init(Fs,NsU),
    sort(NsU,Ns).

init([],[]).
init([fr(S,F)|Fs],[n(F,S)|Ns]) :- init(Fs,Ns).

create_tree([T],T).
create_tree([n(F1,X1),n(F2,X2)|Ns],T) :-
   F is F1+F2,
   insert(n(F,s(n(F1,X1),n(F2,X2))),Ns,NsR),
   create_tree(NsR,T).

insert(N,[],[N]) :- !.
insert(n(F,X),[n(F0,Y)|Ns],[n(F,X),n(F0,Y)|Ns]) :- F < F0, !.
insert(n(F,X),[n(F0,Y)|Ns],[n(F0,Y)|Ns1]) :-
    F >= F0,
    insert(n(F,X),Ns,Ns1).

traverse_tree(T,Cs) :-
    traverse_tree(T,'',Cs1-[]),
    sort(Cs1,Cs),
    write(Cs).

traverse_tree(n(_,A),Code,[hc(A,Code)|Cs]-Cs) :-
    atom(A).

traverse_tree(n(_,s(Left,Right)),Code,Cs1-Cs3) :-
   atom_concat(Code,'0',CodeLeft),
   atom_concat(Code,'1',CodeRight),
   traverse_tree(Left,CodeLeft,Cs1-Cs2),
   traverse_tree(Right,CodeRight,Cs2-Cs3).
```

Output:



brother(X, vatsa).

X = rishi

aunt(X, vatsa).

X = jyoti
X = lata

grandfather(X, vatsa).

X = mulji

grandmother(X, vatsa).

X = ratan

predecessor(X, vatsa).

X = vinod
X = sejal
X = ratan
X = mulji

successor(vinod, X).

X = vatsa
X = rishi

rle([a,a,a,a,b,b,c,c], X).

X = [a, b, c]

Next | 10 | 100 | 1,000 | Stop

slice([a,b,c,d,e,f,g], 2, 5, X).

X = [b, c, d, e]

Next | 10 | 100 | 1,000 | Stop

subsets([a,b,c,d,e,f,g],[2,2,3], X, []).

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

Next | 10 | 100 | 1,000 | Stop

huffman([*fr*(a,45),*fr*(b,13),*fr*(c,12),*fr*(d,16),*fr*(e,9),*fr*(f,5)],_).

[*hc*(a, 0), *hc*(b, 101), *hc*(c, 100), *hc*(d, 111), *hc*(e, 1101), *hc*(f, 1100)]

true

1

Next | 10 | 100 | 1,000 | Stop

GitHub: https://github.com/vatsa1101/AIML