Name: Satyam Jaiswal **UID:** 2021600028 **Batch**: B **Date:** 30/10/2023

EXP 7: Prolog Programming Knowledge and reasoning.

Problem: To solve problems using Prolog Programming

Problem statements:

1) Create a family tree using PROLOG.

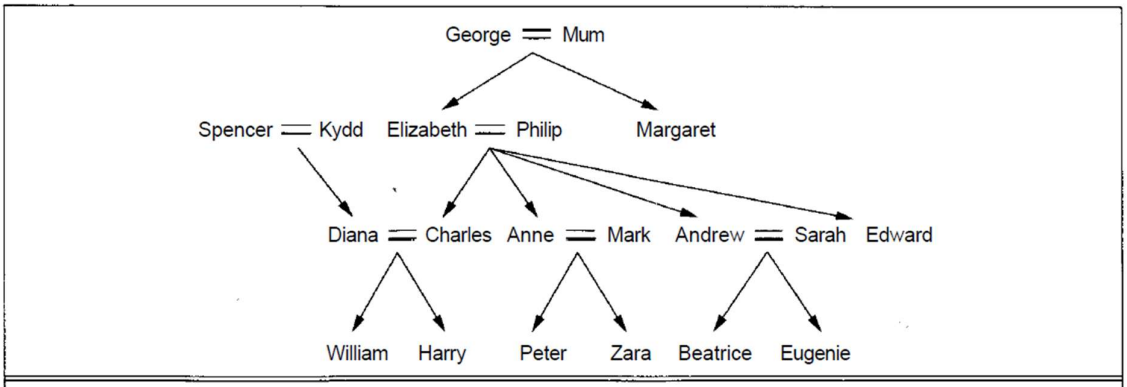


Figure 7.4 A typical family tree.

Write the predicates Grandchild GreatGrandparent, Brother, Sister, Daughter, Son, Aunt, Uncle, BrotherlnLaw, SisterlnLaw, and FirstCousin.

Write down the basic facts depicted in the family tree in Figure 7.4. Using the logical reasoning system in the code repository, TELL it all the sentences you have written down, and ASK it who are Elizabeth's grandchildren, Diana's brothers-in-law, and Zara's great-grandparents.

2) Given a list [a,a,a,a,b,b,b,c,c] write a function that does the following rle([a,a,a,a,b,b,c,c],X) X: [a,b,c]

3) Given a list [a,b,c,d,e,f,g] write a function that does the following slice([a,b,c,d,e,f,g],[2,5],X) X: [c,d,e,f]

4) Group list into sublists according to the distribution given For example subsets([a,b,c,d,e,f,g],[2,2,3],X,[]) should return X = [[a,b][c,d][e,f,g]]

Program:

1)

```
male(charles).

male(william).

male(harry).

male(peter).

male(andrew).
```

```
male(edward).
male(james).
male(george).
male(archie).
male(mark).
female(diana).
female(elizabeth).
female(anne).
female(zara).
female(beatrice).
female(eugenie).
female(louise).
female(charlotte).
parent(george, elizabeth).
parent(george, margaret).
parent(mum, elizabeth).
parent(mum, margaret).
parent(spencer, diana).
parent(kydd, diana).
parent(charles, william).
parent(charles, harry).
parent(diana, william).
parent(diana, harry).
parent(elizabeth, charles).
parent(elizabeth, anne).
parent(elizabeth, andrew).
parent(elizabeth, edward).
parent(philip, charles).
parent(philip, anne).
```

```
parent(philip, andrew).
parent(philip, edward).
parent(anne, peter).
parent(anne, zara).
parent(mark, peter).
parent(mark, zara).
parent(andrew, beatrice).
parent(andrew, eugenie).
parent(sarah, beatrice).
parent(sarah, eugenie).
sibling(charles, anne).
sibling(charles, andrew).
sibling(charles, edward).
sibling(diana, anne).
sibling(diana, andrew).
sibling(diana, edward).
sibling(anne, andrew).
sibling(anne, edward).
sibling(andrew, edward).
sibling(peter, zara).
sibling(william, harry).
sibling(beatrice, eugenie).
spouse(charles, diana).
spouse(diana, charles).
spouse(elizabeth, philip).
spouse(anne, mark).
spouse(andrew, sarah).
father(X, Y) :- parent(X, Y), male(X).
mother(X, Y) :- parent(X, Y), female(X).
```

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

grandchild(X, Y) :- grandparent(Y, X).

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

grandparent(Z, Y).

greatgrandchild(X, Y) :- greatgrandparent(Y, X).

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

male(X), X \= Y.

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

female(X), X \= Y.

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

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

cousin(X, Y) :- parent(Z, X), parent(W, Y),

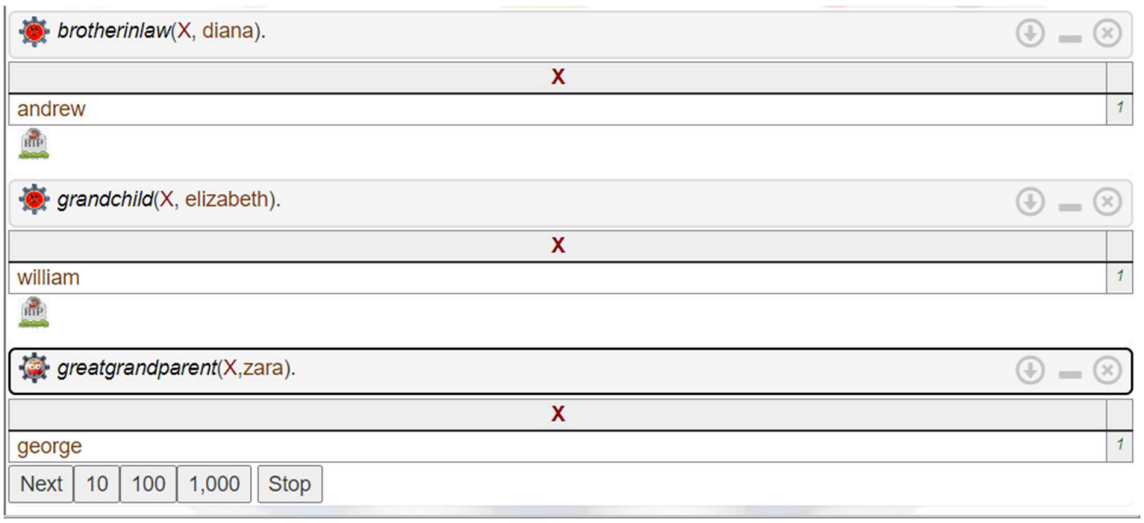
sibling(Z, W).

brotherinlaw(X, Y) :- (spouse(Y, Z), brother(X, Z)).

sisterinlaw(X, Y) :- (wife(X, Z), sister(Z, Y));

(wife(X, Z), brother(Z, Y)).

Output:



2)

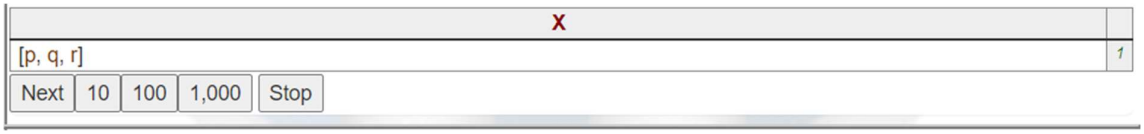
compress([],[]).

compress([X],[X]).

compress([X,X|Xs],Zs) :- compress([X|Xs],Zs).

compress([X,Y|Ys],[X|Zs]) :- X \= Y, compress([Y|Ys],Zs).

Output:



3)

slice(List, [Start, End], Result) :-

slice(List, Start, End, Result).

slice([X|_], 1, 1, [X]).

slice([X|Xs], 1, K, [X|Ys]) :-

K > 1,

K1 is K - 1,

slice(Xs, 1, K1, Ys).

slice([_|Xs], I, K, Ys) :-

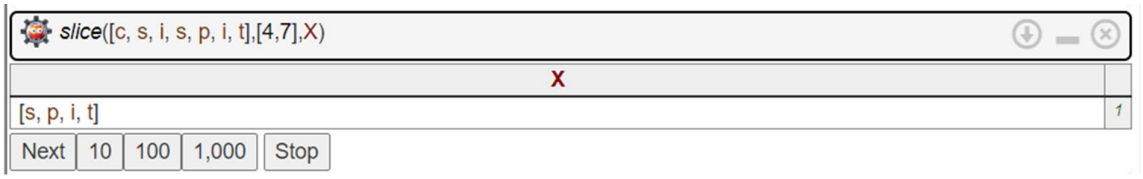
I > 1,

I1 is I - 1,

K1 is K - 1,

slice(Xs, I1, K1, Ys).

Output:



4)

subsets(_, [], [], []).

subsets(List, [N|Ns], [Subset|Rest], Remaining) :-

split(List, N, Subset, NewList),

subsets(NewList, Ns, Rest, Remaining).

split(List, N, Subset, Remaining) :-

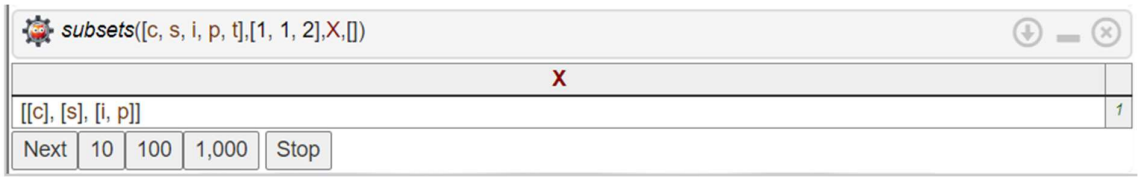
length(Subset, N),

append(Subset, Remaining, List).

remove_element(Element, List, Result) :-

select(Element, List, Result).

Output:



Conclusion:

1) Learnt about prolog and how it is used in programming and querying logical knowledge base.
2) Also, learnt about how to convert family tree into a knowledge base creating slices and subsets.