



UNIVERSITY OF ASIA PACIFIC

Department of Computer Science & Engineering

Course Title – Artificial Intelligence and Expert Systems Lab.

Course Code – CSE-404.

Assignment : 1 – Basic family relationship tree structure of family using prolog.

SUBMITTED BY

Shawan Das.

ID – 19101020

Section – A1

SUBMITTED TO

Dr. Nasima Begum

University of Asia Pacific

Date of Submission – 20-08-2022

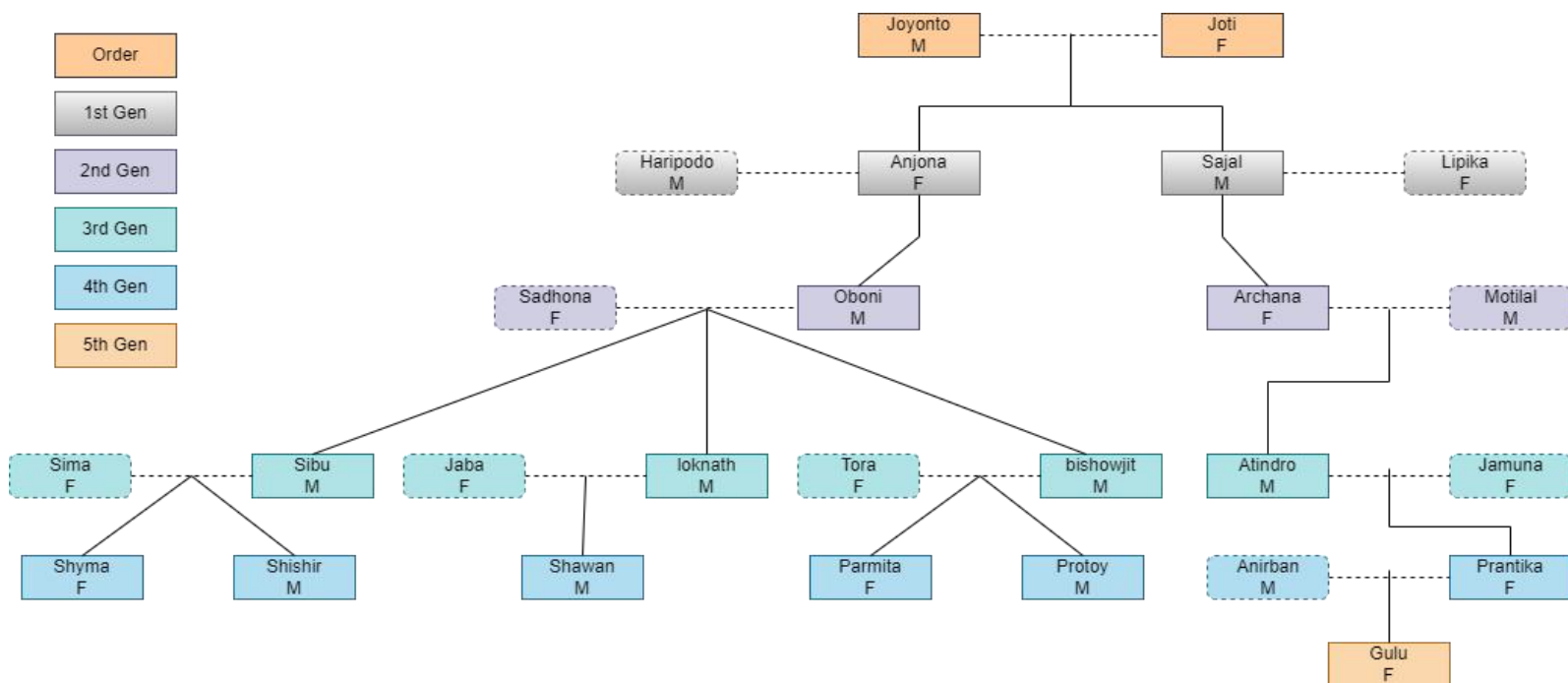
Problem Title: Implement a Basic family Relationship tree structure of my own family using Prolog.

Problem Description: We need to design the relationship tree structure of my family using “Prolog”. Also write rules against degree and removal for up to 3rd degree and twice removed situations for cousin relationships. We have to use recursion in my rules for different family relations.

Tools & Languages:

- Diagram.net . (Design Tree)
- VS Code/Notepad. (Write rules & facts)
- Swi Prolog.

Diagram:



Necessary Logic:

Relationship between subject and relative given the relationship to their most recent common ancestor

		Relative		
Separation in generations to ancestor R→		2	3	4
S↓	Relationship to ancestor	Grandparent	Great-grandparent	Great-great-grandparent
Subject	2	Grandparent	1st cousin	1st cousin once removed
	3	Great-grandparent	1st cousin once removed	2nd cousin
	4	Great-great-grandparent	1st cousin twice removed	2nd cousin once removed
				3rd cousin

For cousins ($R \geq 2$ and $S \geq 2$): degree = $\min(R, S) - 1$, removal = $|R - S|$

Sample Input/Output:

```
?- sibling(X,shishir).
X = shyma .

?- sibling(X,parmita).
X = protoy .

?- cousin(X,shawan).
X = protoy ;
X = parmita ;
X = shishir ;
X = shyma .

?- sibling(X,sibu).
X = loknath ;
X = bishowjit ;
false.

?- grandparent(X,shawan).
X = oboni ;
X = sadhona ;
false.

?- grandparent(X,gulu).
X = atindro ;
X = jamuna .

?- grandparent(X,prantika).
X = motilal ;
X = archana .
```

```
?- parent(X,shawan).
X = loknath ;
X = jaba .

?- parent(X,shishir).
X = sibu ;
X = sima .

?- parent(X,parmita).
X = bishowjit ;
X = tora .

?- parent(X,gulu).
X = anirban ;
X = prantika .

?- parent(X,prantika).
X = atindro ;
X = jamuna .

?- parent(X,lokmath).
X = oboni ;
X = sadhona .

?- parent(X,atindro).
X = motilal ;
X = archana .
```

```

?- great_grandparent(X,shawan).
X = haripodo ;
X = anjona ,

?- great_great_grandparent(X,shawan).
X = joyonto ;
X = joti ,

?- great_great_grandfather(X,prantika)
X = joyonto ;
false.

?- great_great_grandparent(X,prantika)
X = joyonto ;
X = joti ,

?- thirdCousin(shawan,X).
X = prantika ;
false.

?- tcor(X,shawan).
X = gulu ,

?- secondCousin(X,loknath).
X = atindro ;
false.

?- scor(X,loknath).
X = prantika ;
false.

?- sctr(X,loknath).
X = gulu ;
false.

?- fcoc(X,archana).
Correct to: "fcor(X,archana)"? yes
X = sibu ;
X = loknath ;
X = bishowjit ;
false.

?- fcoc(X,oboni).
Correct to: "fcor(X,oboni)"? yes
X = atindro ;
false.

?- fctc(X,oboni).
Correct to: "fctr(X,oboni)"?
Please answer 'y' or 'n'? yes
X = prantika ;
false.

?- fcor(X,oboni).
X = atindro ;
false.

?- fctr(X,oboni).
X = prantika ;
false.

?- fcthr(X,oboni).
X = gulu ,

?-

```

Source Code: github.com/Shawan-Das

Slide: [Canva/ShawanDas](https://www.canva.com/ShawanDas)

Challenges & Conclusion:

Faced some minor difficulties while writing the code. Name and relations should be correctly placed. Facts have to be as simple and specific. There were some errors in SWI-Prolog but those were fixed successfully after some troubleshooting.

To implement any knowledge-base, creating a decision tree and a structured logics can help a lot while troubleshooting. Naming conventions are case sensitive. Be careful of spellings.