

EX.NO: 06

PROLOG - FAMILY TREE

AIM :

To develop a family tree program using PROLOG with all possible facts, rules, and queries.

CODE :

The image shows the SWISH Prolog IDE interface. At the top is a menu bar with 'File', 'Edit', 'Examples', and 'Help'. Below the menu bar is a tab labeled 'Program' with a close button and a plus sign to add more programs. The main area contains a list of Prolog facts and rules, numbered 1 through 26. The facts define the gender of individuals (male and female) and the parent-child relationships (parentOf). The rules define family relationships (father, mother, grandfather, grandmother, brother, and sister) based on the facts.

```
1 male(peter).
2 male(john).
3 male(chris).
4 male(kevin).
5
6 female(betty).
7 female(jeny).
8 female(lisa).
9 female(helen).
10
11 parentOf(chris,peter).
12 parentOf(chris,betty).
13 parentOf(helen,peter).
14 parentOf(helen,betty).
15 parentOf(kevin,chris).
16 parentOf(kevin,lisa).
17 parentOf(jeny,john).
18 parentOf(jeny,helen).
19
20 father(X,Y):- male(Y), parentOf(X,Y).
21 mother(X,Y):- female(Y), parentOf(X,Y).
22 grandfather(X,Y):- male(Y),parentOf(X,Z),parentOf(Z,Y).
23 grandmother(X,Y):- female(Y),parentOf(X,Z),parentOf(Z,Y).
24 brother(X,Y):- male(Y), father(X,Z), father(Y,W),Z==W.
25 sister(X,Y):- female(Y), father(X,Z),father(Y,W),Z==W.
26
```

OUTPUT :

GIVING QUERY:

?- sister(**x**,**y**).

Examples▲ History▲ Solutions▲

☐ table results **Run!**

⚙️ *father*(**x**,**y**).

x = chris,

y = peter

Next 10 100 1,000 Stop

⚙️ *mother*(**x**,**y**).

x = chris,

y = betty

Next 10 100 1,000 Stop

⚙️ *sister*(**x**,**y**).

x = **y**, **y** = jeny

Next 10 100 1,000 Stop

RESULT :

Thus, the implementing of family tree program using PROLOG is successfully executed and the output is verified.