Experiment - 10

10. Implementation of Expert System with backward chaining using RVD/PROLOG

```
/* Facts */
male(jack).
male(oliver).
male(ali).
male(james).
male(simon).
male(harry).
female(helen).
female(sophie).
female(jess).
female(lily).

parent_of(jack,jess).
parent_of(jack,lily).
parent_of(helen, jess).
parent_of(helen, lily).
parent_of(oliver,james).
parent_of(sophie, james).
parent_of(jess, simon).
parent_of(ali, simon).
parent_of(lily, harry).
parent_of(james, harry).

/* Rules */
father_of(X,Y):- male(X),
   parent_of(X,Y).

mother_of(X,Y):- female(X),
   parent_of(X,Y).

grandfather_of(X,Y):- male(X),
   parent_of(X,Z),
   parent_of(Z,Y).

grandmother_of(X,Y):- female(X),
   parent_of(X,Z),
   parent_of(Z,Y).
```

```
sister_of(X,Y):- %(X,Y or Y,X)%
   female(X),
   father_of(F, Y), father_of(F,X),X \= Y.

sister_of(X,Y):- female(X),
   mother_of(M, Y), mother_of(M,X),X \= Y.

aunt_of(X,Y):- female(X),
   parent_of(Z,Y), sister_of(Z,X),!.

brother_of(X,Y):- %(X,Y or Y,X)%
   male(X),
   father_of(F, Y), father_of(F,X),X \= Y.

brother_of(X,Y):- male(X),
   mother_of(M, Y), mother_of(M,X),X \= Y.

uncle_of(X,Y):-
   parent_of(Z,Y), brother_of(Z,X).

ancestor_of(X,Y):- parent_of(X,Y).
ancestor_of(X,Y):- parent_of(X,Z),
   ancestor_of(Z,Y).
```

Output:

SWI-Prolog (AMD64, Multi-threaded, version 9.0.3)

File Edit Settings Run Debug Help

Welcome to SWI-Prolog (threaded, 64 bits, version 9.0.3)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/admin/Desktop/A Star Programs/bc.pl compiled 0.00 sec, 32 clauses
?- mother_of(jess,helen).
false.

?- brother_of(james,simon).
false.

?- ancestor_of(jack,simon).
true .

?- mother_of(X,jess).
X = helen .

?- parent_of(X,simon).
X = jess .

?- sister_of(X,lily).
X = jess .

?- ancestor_of(X,lily).
X = jack .

?-