bDATA: 31/03/2022 EXP No: 07	Title of the Lab Unification and resolution in Prolog	Name:Avinash reddy Vasipalli Registration Number: RA1911027010007 Section: N1 Lab Batch: 1 Day Order: 2

AIM: To determine Unification and Resolution in Prolog

Description of concept or problem:

Unification is a process of combining 2 different logical atomic expressions using a substitution.

Resolution is a technique in which we prove theorems by building refutations proofs. We get conclusion here.

Manual solution:

In Unification we declare knowledge base and we ask queries in Prolog and logical answers are obtained

Where as in resolution we declare a knowledge base but we have a program to check the condition and if they are proven by resolution method then we have successful programs.

Algorithm for Prolog:

- Conversation of facts into first-order logic
- Convert FOL statements into CNF
- Negate the statement which needs to prove
- Draw resolution graph(unification)

$$\begin{aligned} a \to b \wedge (true \to d) &\implies a \to b \wedge d \\ a \to (true \to e) &\implies a \to e \\ \neg true &\implies false \end{aligned}$$

Program Implementation [Coding]:

Unification

father(adam, bob).

father(bob, charlie).

father(charlie, dave).

father(dave, edward).

grandfather(X,Z):- father(X,Y), father(Y,Z).

Week 07.pl

File Edit Browse Compile Prolog Pce Help

Week 07.pl

```
father(adam, bob).
father(bob, charlie).
father(charlie, dave).
father(dave, edward).

grandfather(X,Z):- father(X,Y), father(Y,Z).
```

Resolution:

person(ali,20).
person(bob,20).
person(cal,25).

hobby(ali,skiing).
hobby(bob,skiing).
hobby(cal,skiing).

friends(P1,P2):-
hobby(P1,H),
hobby(P2,H),
P1\=P2,
person(P1,A1),
person(P2,A2),
AD is abs(A2-A1),
AD=<3.

avi.pl

File Edit Browse Compile Prolog Pce Help

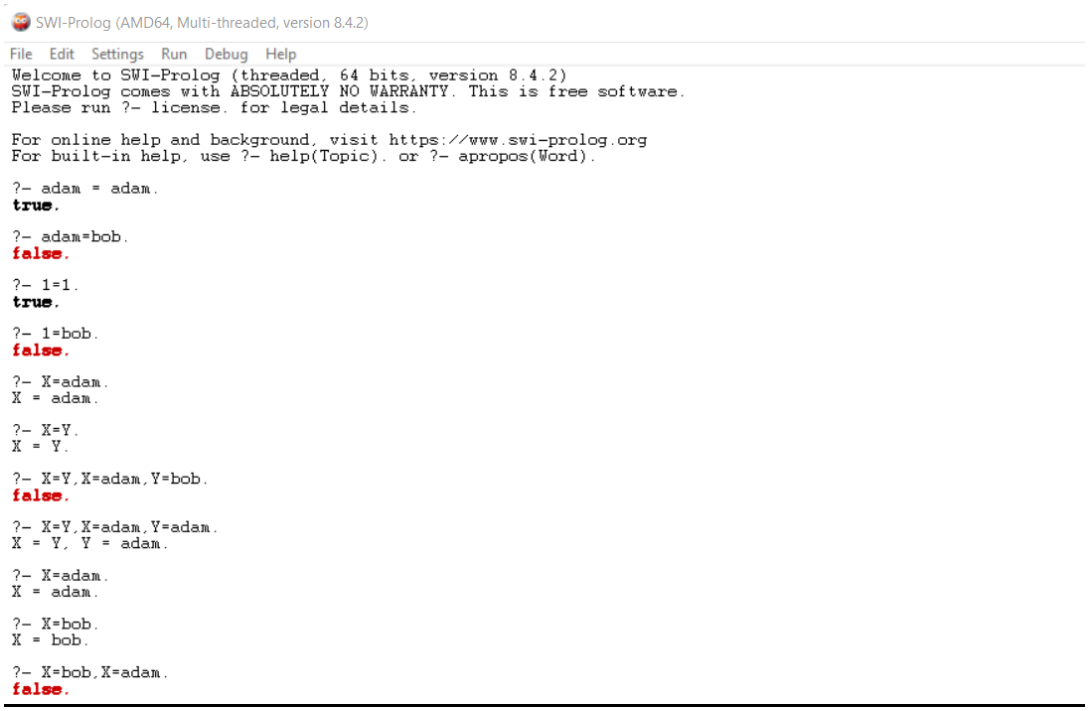
avi.pl

```
person(ali,20).
person(bob,20).
person(cal,25).

hobby(ali,skiing).
hobby(bob,skiing).
hobby(cal,skiing).

friends(P1,P2):-
    hobby(P1,H),
    hobby(P2,H),
    P1\=P2,
    person(P1,A1),
    person(P2,A2),
    AD is abs(A2-A1),
    AD=<3.
```

user:person/2: (loaded) static, 3 clauses, number_of_rules(0),

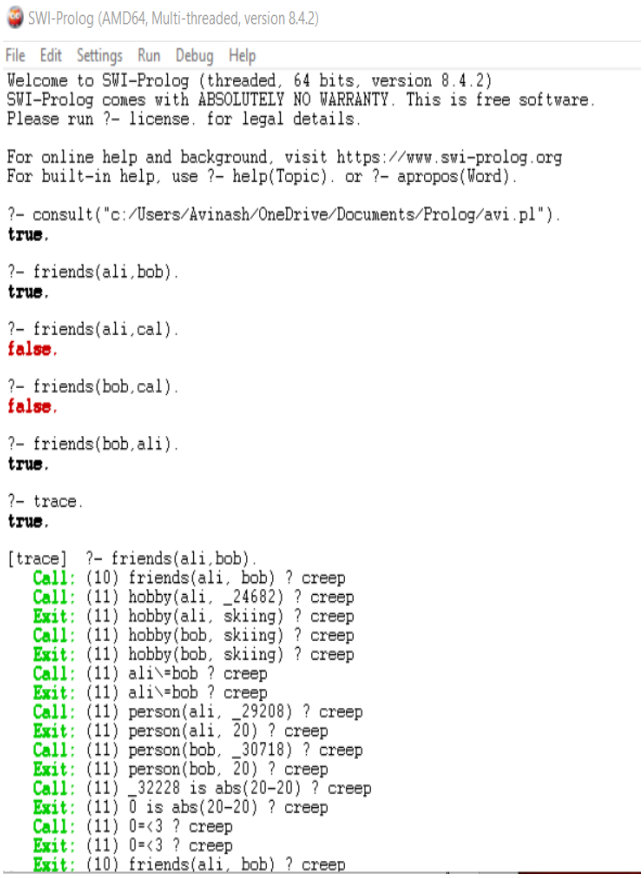
Output:

Unification

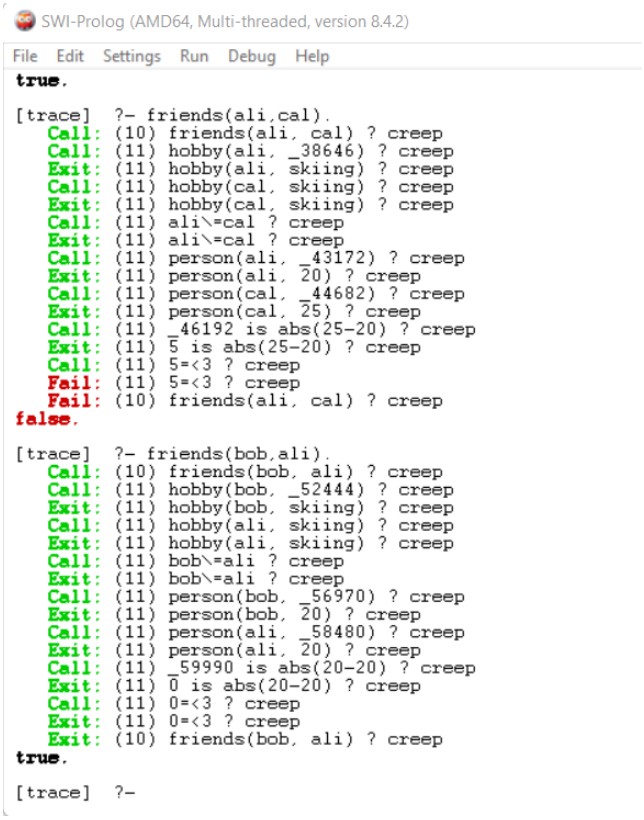


```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File  Edit  Settings  Run  Debug  Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.2)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

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

?- adam = adam.
true.

?- adam=bob.
false.

?- 1=1.
true.

?- 1=bob.
false.

?- X=adam.
X = adam.

?- X=Y.
X = Y.

?- X=Y,X=adam,Y=bob.
false.

?- X=Y,X=adam,Y=adam.
X = Y, Y = adam.

?- X=adam.
X = adam.

?- X=bob.
X = bob.

?- X=bob,X=adam.
false.
```

Resolution



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File  Edit  Settings  Run  Debug  Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.2)
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).

?- consult("c:/Users/Avinash/OneDrive/Documents/Prolog/avi.pl").
true.

?- friends(ali,bob).
true.

?- friends(ali,cal).
false.

?- friends(bob,cal).
false.

?- friends(bob,ali).
true.

?- trace.
true.

[trace]  ?- friends(ali,bob).
   Call: (10) friends(ali, bob) ? creep
   Call: (11) hobby(ali, _24682) ? creep
   Exit: (11) hobby(ali, skiing) ? creep
   Call: (11) hobby(bob, skiing) ? creep
   Exit: (11) hobby(bob, skiing) ? creep
   Call: (11) ali\=bob ? creep
   Exit: (11) ali\=bob ? creep
   Call: (11) person(ali, _29208) ? creep
   Exit: (11) person(ali, 20) ? creep
   Call: (11) person(bob, _30718) ? creep
   Exit: (11) person(bob, 20) ? creep
   Call: (11) _32228 is abs(20-20) ? creep
   Exit: (11) 0 is abs(20-20) ? creep
   Call: (11) 0=<3 ? creep
   Exit: (11) 0=<3 ? creep
   Exit: (10) friends(ali, bob) ? creep
```



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File  Edit  Settings  Run  Debug  Help
true.

[trace]  ?- friends(ali,cal).
   Call: (10) friends(ali, cal) ? creep
   Call: (11) hobby(ali, _38646) ? creep
   Exit: (11) hobby(ali, skiing) ? creep
   Call: (11) hobby(cal, skiing) ? creep
   Exit: (11) hobby(cal, skiing) ? creep
   Call: (11) ali\=cal ? creep
   Exit: (11) ali\=cal ? creep
   Call: (11) person(ali, _43172) ? creep
   Exit: (11) person(ali, 20) ? creep
   Call: (11) person(cal, _44682) ? creep
   Exit: (11) person(cal, 25) ? creep
   Call: (11) _46192 is abs(25-20) ? creep
   Exit: (11) 5 is abs(25-20) ? creep
   Call: (11) 5=<3 ? creep
   Fail: (11) 5=<3 ? creep
   Fail: (10) friends(ali, cal) ? creep
false.

[trace]  ?- friends(bob,ali).
   Call: (10) friends(bob, ali) ? creep
   Call: (11) hobby(bob, _52444) ? creep
   Exit: (11) hobby(bob, skiing) ? creep
   Call: (11) hobby(ali, skiing) ? creep
   Exit: (11) hobby(ali, skiing) ? creep
   Call: (11) bob\=ali ? creep
   Exit: (11) bob\=ali ? creep
   Call: (11) person(bob, _56970) ? creep
   Exit: (11) person(bob, 20) ? creep
   Call: (11) person(ali, _58480) ? creep
   Exit: (11) person(ali, 20) ? creep
   Call: (11) _59990 is abs(20-20) ? creep
   Exit: (11) 0 is abs(20-20) ? creep
   Call: (11) 0=<3 ? creep
   Exit: (11) 0=<3 ? creep
   Exit: (10) friends(bob, ali) ? creep
true.

[trace]  ?-
```

Result:

Successfully implemented the Unification and resolution using SWG Prolog for a taken English grammmer