deerwalk
DWIT College

Lab Report 2

Submitted by:

Abhishek Kadariya

0501

2019 ‘A’

Submitted to:

Birodh Rijal

(Artificial Intelligence Lecturer)

A. Load "family.pl" and find the answer to the following questions:

(a) Is Abraham a parent of Bart?

No, Abraham is not a parent of Bart.

```
?- consult("F:\\DWIT\\5. Fifth semester\\4. Artificial Intelligence\\lab\\lab 2\\lab.pl").
true.

?- parent(abraham, bart).
false.
```

(b) Is Lisa a child of Mona?

No. Lisa is not child of Mona.

```
?- parent(mona,lisa).
false.
```

(c) Who are Bart's parent?

Bart's parent are Homer and Marge.

```
?- parent(X, bart).
X = homer ;
X = marge.
```

(d) Who are Homer's children?

Bart and Lisa are Homer's children.

```
?- parent(homer, X).
X = bart ;
X = lisa ;
```

A. Add the following facts to the database using only the parent predicate:

(a) Maggie is the daughter of Homer and Marge.

(b) Selma is the parent of Ling

```
?- assert(parent(marge,maggie)).
true.

?- assert(parent(homer,maggie)).
true.

?- assert(parent(selma,ling)).
true.
```

```
?- parent(homer,maggie).
true .

?- parent(marge,maggie).
true.

?- parent(selma,ling).
true.
```

B. Find the answer to the following queries:

(a) Who are the grandchildren of Abraham?

Bart, Lisa and Maggie are grandchildren of Abraham.

```
?- parent(abraham,X),parent(X,Y).
X = homer,
Y = bart ;
X = homer,
Y = lisa ;
X = homer,
Y = maggie.
```

(b) Who are the grandchildren of Clancy who have Marge as a parent?

Bart, Lisa and Maggie are the grandchildren of Clancy who have Marge as a parent.

```
?- parent(clancy,marge),parent(marge,Y).
Y = bart ;
Y = lisa ;
Y = maggie ;
```

C. Augment the database with predicates to distinguish between male and female persons.

male(abraham).
male(homer).
male(jackie).
male(bart).
female(clancy).
female(marge).
female(selma).
female(patty).
female(lisa).
female(mona).
female(maggie).
female(selma).
female(ling).

D. Query the database to find out:

(a) Who are the male children of Marge?

Bart is male child of Marge.

```
?- parent(marge,X),male(X).
X = bart ;
false.
```

(b) Who is Lisa's father?

Homer is Lisa's father.

```
?- parent(X,lisa),male(X).
X = homer ;
false.
```

(c) Who is Bart's grandfather?

Abraham and Jackie are Bart's grandfather.

```
?- parent(X,Y),parent(Y,bart),male(X).
X = abraham,
Y = homer ;
X = jackie,
Y = marge ;
false.
```

E. Augment the database with rules and predicate for the following relations:

(a) Mother

mother(X,Y):-female(X),parent(X,Y).

(b) Father

father(X,Y):-male(X),parent(X,Y).

(c) Grandfather

grandfather(X,Y):-male(X),parent(X,Z),parent(Z,Y).

(d) Grandmother

grandmother(X,Y):-female(X),parent(X,Z),parent(Z,Y).

F. Add the different relation to your database, which is true if its two arguments are not the same, and is defined as follows. Do not worry about the definition for now, it will be eventually taught. different(X,X):-!,fail. different(X,Y).

G. Now, augment the database with rules and predicates for the following relations:

(i) sister: so that sister(X,Y) is true if X is the sister of Y

sister(X,Y):-female(X),parent(Z,Y),parent(Z,X),(X\=Y).

(ii) brother: so that brother(X,Y) is true if X is the brother of Y

brother(X,Y):-male(X),parent(Z,Y),parent(Z,X),(X\=Y).

(iii) aunt: so that aunt(X,Y) is true if X is the aunt of Y

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

(iv) uncle: so that uncle(X,Y) is true if X is the uncle of Y

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

(v) cousin: so that cousin(X,Y) is true if X is the cousin of Y

cousin(X,Y):-parent(Z,X),(uncle(Z,Y);aunt(Z,Y)).

(vi) siblings: so that siblings(X,Y) is true if X is the cousin of Y

sibling(X,Y):-(parent(Z,X),(uncle(Z,Y);aunt(Z,Y)));(parent(Z,Y),parent(Z,X)).

H. Create your own family tree. Only use the parent relation and male/female predicate. Consult your parents if needed.

parent(mohan, abhishek).
parent(mohan,isha).
parent(laxmi,abhishek).
parent(laxmi,isha).
parent(indra,surendra).
parent(indra,narendra).
male(abhishek)
male(indra).
female(isha).
male(surendra).
male(narendra).
female(laxmi).

I. **Extra credit: Implement a rule for ancestor relation which is true if X is the ancestor of Y.**

ancestor(X,Y):-parent(X,Y).
ancestor(X,Y):-parent(X,S),ancestor(S,Y).

```
?- ancestor(X,bart).
X = homer ;
X = marge ;
X = abraham ;
X = mona ;
X = clancy ;
X = jackie ;
false.

?- ancestor(X,sahil).
X = rajendra ;
X = kanta ;
X = krishna ;
false.

?- ancestor(X,homer).
X = abraham ;
X = mona ;
false.

?- ancestor(X,marge).
X = clancy ;
X = jackie ;
false.

?- ancestor(X,patty).
X = jackie ;
false.
```