**DEERWALK INSTITUTE OF TECHNOLOGY**

**LAB 2: Introduction to Prolog.**

**(Artificial Intelligence)**

|  |  |  |  |
| --- | --- | --- | --- |
| **SUBMITTED BY:** |  |  | **SUBMITTED TO:** |
| **NAME: Kriston Pal** |  |  |  |
| **ROLL NO.: 0511** |  |  |  |
| **SECTION: A** |  |  |  |
| **DATE: 13th April, 2018** |  |  | **Birodh Rijal** |

**KATHMANDU, NEPAL**

**Edit a ﬁle and enter the following database of facts. Save the ﬁle under the name “family.pl”**

**parent(abraham,homer).**

**parent(mona,homer).**

**parent(clancy,marge).**

**parent(jackie, marge).**

**parent(jackie.selma).**

**parent(jackie,patty).**

**parent(homer,bart).**

**parent(homer,lisa).**

**parent(marge,bart).**

**parent(marge,lisa).**

1. **Load “family.pl” and ﬁnd the answer to the following questions:**
2. **Is Abraham a parent of Bart?**

No, Abraham is not a parent of Bart.

1. **Is Lisa a child of Mona?**

No. Lisa is not child of Mona.

1. **Who are Bart’s parent?**

Bart’s parent are Homer and Marge.

1. **Who are Homer’s children?**

Bart and Lisa are Homer’s children.

A screenshot of a computer

Description generated with very high confidence

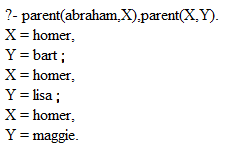
1. **Add the following facts to the database using only the parent predicate:**
2. **Maggie is the daughter of Homer and Marge.**
3. **Selma is the parent of Ling**

A screenshot of a computer

Description generated with very high confidence

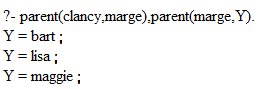
1. **Find the answer to the following queries:**
2. **Who are the grandchildren of Abraham?**

Bart, Lisa and Maggie are grandchildren of Abraham.



1. **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.



1. **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).

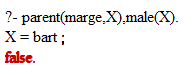
female(maggie).

female(selma).

female(ling).

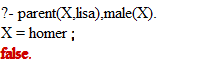
1. **Query the database to ﬁnd out:**
2. **Who are the male children of Marge?**

Bart is male child of Marge.



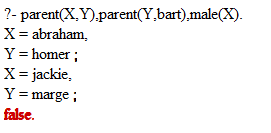
1. **Who is Lisa’s father?**

Homer is Lisa’s father.



1. **Who is Bart’s grandfather?**

Abraham and Jackie are Bart’s grandfather.



1. **Augment the database with rules and predicate for the following relations:**
2. **Mother**

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

1. **Father**

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

1. **Grandfather**

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

1. **Grandmother**

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

1. **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).**

1. **Now, augment the database with rules and predicates for the following relations:**
2. **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).

1. **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).

1. **aunt: so that aunt(X,Y) is true if X is the aunt of Y**

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

1. **uncle: so that uncle(X,Y) is true if X is the uncle of Y**

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

1. **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)).

1. **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)).

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

parent(shiv,kriston).

parent(shiv,radha).

parent(durga,kriston).

parent(durga,radha).

male(kriston).

male(shiv).

female(radha).

female(durga).

1. **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).

