

E05 Family Problem (Prolog)

18364066 Yanzuo Lu

October 6, 2020

Contents

1	About Cousin and Removed	2
2	Problem Description	3
3	Tasks	3
4	Codes	4
5	Results	6

1 About Cousin and Removed

What Is a First Cousin, Twice Removed?

If someone walked up to you and said, “Howdy, I’m your third cousin, twice removed,” would you have any idea what they meant? Most people have a good understanding of basic relationship words such as “mother,” “father,” “aunt,” “uncle,” “brother,” and “sister.” But what about the relationship terms that we don’t use in everyday speech? Terms like “second cousin” and “first cousin, once removed”? We don’t tend to speak about our relationships in such exact terms (“cousin” seems good enough when you are introducing one person to another), so most of us aren’t familiar with what these words mean.

Relationship Terms

Sometimes, especially when working on your family history, it’s handy to know how to describe your family relationships more exactly. The definitions below should help you out.

Cousin (a.k.a “first cousin”)

Your first cousins are the people in your family who have two of the same grandparents as you. In other words, they are the children of your aunts and uncles.

Second Cousin

Your second cousins are the people in your family who have the same great-grandparents as you., but not the same grandparents.

Third, Fourth, and Fifth Cousins

Your third cousins have the same great great grandparents, fourth cousins have the same great-great-great-grandparents, and so on.

Removed

When the word “removed” is used to describe a relationship, it indicates that the two people are from different generations. You and your first cousins are in the same generation (two generations younger than your grandparents), so the word “removed” is not used to describe your relationship.

The words “**once removed**” mean that there is a difference of one generation. For example, your mother’s first cousin is your first cousin, once removed. This is because your mother’s first cousin is one generation younger than your grandparents and you are two generations younger than your grandparents. This one-generation difference equals “once removed.”

Twice removed means that there is a two-generation difference. You are two generations younger than a first cousin of your grandmother, so you and your grandmother’s first cousin are first cousins, twice removed.

2 Problem Description

Please fulfill the following tasks by using Prolog:

1. Using the predicates **male**, **female**, **child**, and **spouse**, write facts and rules describing the family tree in Figure 2, and add the fact that William has a daughter Charlotte. Please do not write redundant facts that can be defined with rules.
2. Write rules describing the predicates **Grandchild**, **Greatgrandparent**, **Ancestor**, **Sibling**, **Brother**, **Sister**, **Daughter**, **Son**, **FirstCousin**, **BrotherInLaw**, **SisterInLaw**, **Aunt**, and **Uncle**.
3. Find out the proper definition of **mth cousin n times removed**, and write rules to define the predicate `mthCousinNremoved(X,Y,M,N)`. *Hint: You'd better define a helper predicate `distance(X,Y,N)` meaning that there are N generations between X and Y by recursion (please refer to `hanoi.pl`).*
4. ASK who are **Elizabeth**' s grandchildren, **Diana**' s brothers-in-law, **Zara**' s great-grandparents, **Eugenie**' s ancestors, and **Charlotte**' s first cousin once removed.

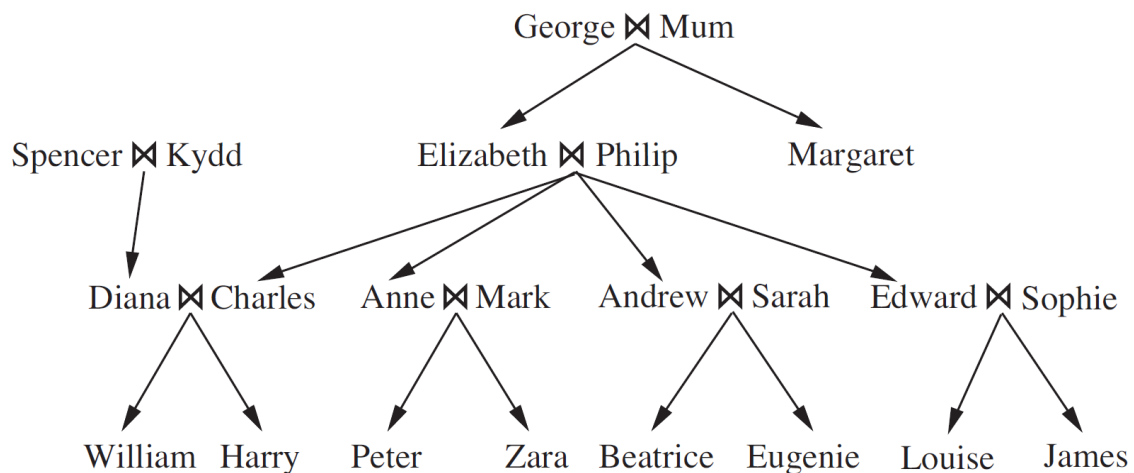


Figure 1: A typical family tree. The symbol \bowtie connects spouses and arrows point to children.

3 Tasks

1. Please complete the Prolog codes. There are several tutorials in the folder and I will explain the usage of Prolog in class.

2. Write the related codes and take a screenshot of the running results in the file named E05_YourNumber.pdf, and send it to ai_2020@foxmail.com.

4 Codes

```
male(george).  
male(philip).  
male(spencer).  
male(charles).  
male(mark).  
male(andrew).  
male(edward).  
male(william).  
male(harry).  
male(peter).  
male(james).
```

```
female(mum).  
female(kydd).  
female(elizabeth).  
female(margaret).  
female(diana).  
female(anne).  
female(sarah).  
female(sophie).  
female(zara).  
female(beatrice).  
female(eugenie).  
female(louise).  
female(charlotte).
```

```
child(elizabeth, george).  
child(elizabeth, mum).  
child(diana, spencer).  
child(diana, kydd).  
child(charles, elizabeth).  
child(charles, philip).  
child(anne, elizabeth).  
child(anne, philip).  
child(andrew, elizabeth).
```

```

child(andrew , philip).
child(edward , elizabeth).
child(edward , philip).
child(william , diana).
child(william , charles).
child(harry , diana).
child(harry , charles).
child(peter , anne).
child(peter , mark).
child(zara , anne).
child(zara , mark).
child(beatrice , andrew).
child(beatrice , sarah).
child(eugenie , andrew).
child(eugenie , sarah).
child(louise , edward).
child(louise , sophie).
child(james , edward).
child(james , sophie).
child(charlotte , william).

```

```

spouse(george , mum).
spouse(spencer , kydd).
spouse(elizabeth , philip).
spouse(diana , charles).
spouse(anne , mark).
spouse(andrew , sarah).
spouse(edward , sophie).
spouse(mum , george).
spouse(kydd , spencer).
spouse(philip , elizabeth).
spouse(charles , diana).
spouse(mark , anne).
spouse(sarah , andrew).
spouse(sophie , edward).

```

```

grandchild(X, Y) :- child(X, Z) , child(Z, Y) , male(Z).
greatgrandparent(X, Y) :- grandchild(Y, Z) , child(Z, X) , male(Z).
ancestor(X, Y) :- child(Y, X); grandchild(Y, X); greatgrandparent(X, Y).
sibling(X, Y) :- male(Z) , child(X, Z) , child(Y, Z) , \+ (X = Y).
brother(X, Y) :- sibling(X, Y) , male(X).

```

```

sister(X, Y) :- sibling(X, Y), female(X).
daughter(X, Y) :- child(X, Y), female(X).
son(X, Y) :- child(X, Y), male(X).
firstCousin(X, Y) :- grandchild(X, Z), grandchild(Y, Z), male(Z), \+ (X
    = Y).
brotherInLaw(X, Y) :- spouse(Y, Z), brother(X, Z).
sisterInLaw(X, Y) :- spouse(Y, Z), sister(X, Z).

mthCousinNremoved(X, Y, 0, 0) :- sibling(X, Y).
mthCousinNremoved(X, Y, M, 0) :- M1 is M-1, child(X, U), child(Y, V),
    mthCousinNremoved(U, V, M1, 0).
mthCousinNremoved(X, Y, M, N) :- N1 is N-1, child(Y, Z),
    mthCousinNremoved(X, Z, M, N1).

aunt(X, Y) :- female(X), mthCousinNremoved(X, Y, 0, 1).
uncle(X, Y) :- male(X), mthCousinNremoved(X, Y, 0, 1).

```

5 Results

本次实验主要考察的内容是对于 Prolog 语言的理解和应用，在英国王室成员关系图的基础上构建若干亲属关系的逻辑是难点，尤其是 mthCousinNremoved 逻辑的构建需要应用到递归的思想，此处我没有利用 hint 中的指示使用 distance 逻辑，原因是 cousin 要求共同祖先而 distance 计算两个人之间的代数差不会考虑这一点。

实验过程中让我比较疑惑的一点是 \= 符号的使用，尽管和 \+ 大致相同都是表达不满足条件，但是 \= 在嵌套使用中交换逗号前后语句会导致结果的不同，\+ 则不会有这个问题，希望后续能够在课上解决这个问题。下面给出 pdf 中要求查询若干语句的结果。

```
?- [main].  
true.
```

```
?- grandchild(X, elizabeth), write(X), nl, fail.  
william  
harry  
peter  
zara  
beatrice  
eugenie  
louise  
james  
false.
```

```
?- brotherInLaw(X, diana), write(X), nl, fail.  
andrew  
edward  
false.
```

```
?- greatgrandparent(X, zara), write(X), nl, fail.  
george  
mum  
false.
```

```
?- ancestor(X, eugenie), write(X), nl, fail.  
andrew  
sarah  
elizabeth  
philip  
george  
mum  
false.
```

```
?- nthCousinNremoved(X, charlotte, 1, 1), write(X), nl, fail.  
peter  
zara  
beatrice  
eugenie  
louise  
james  
false.
```