Assignment 3 - Twitbook

Philip Lassen (vgh804), Nicolas Ringsmose Larsen (vgn209)

October 2018

1 Introduction

This was our first time working with Prolog for both Philip and Nicolas. Philip had actually worked with Datalog which is syntactically a subset of Prolog. It is used as a query language for deductive databases.

2 Implementation

In order to complete the assignment a few of the built in functions were re-implemented (as specified by the spec). The member function was something that was used frequently. In our implementation this was called *mem*. We also needed to use = quite frequently. Luckily the implementation of this predicate was trivial, and called *same* in our implementation. One built in predicate that would have been extremely useful was the not function.

Our implementation of Mem is given below

```
mem(Y, [Y| Tail]).
mem(Y, [Head | Tail]) :- mem(Y, Tail).
likes([person(X, Friends)|Tail], X, Y) :- mem(Y, Friends).
likes([Head | Tail], X, Y) :- likes(Tail, X, Y).
```

This code became substantially longer as we had to redefine a built in predicate as discussed earlier. The hardest part of the implementation came when negation would have been extremely helpful in defining a fact.

Our code was written far more verbosely as a result. As previously stated we were unable to come up with the not function, which would have allowed negation. We are still curious if there is a simple implementation of this predicate.

3 Testing

In order to test the code we tested the code against the Graph given as an example in the homework specification. We made sure to test everything when the names did not appear in the Graph, as well as a variety of other edge cases for the small example Graph provided.

4 Assessment

As we were unable to come up with our own implementation for the predicate *not*, This made a lot of the code very clunky. Many things would have become one liners with this simple predicate. We also were working on the resubmit of Homework 1. Thus we were unable to finish level 2, or start level 3. This means we are still missing a fair portion of the desired implementations.