Shofiqur Rahman [shofi384@gmail.com](mailto:shofi384@gmail.com)

Bashir Ahamed

Avi Areman

Professor Stephen Lucci

CSC.30400 – Artificial Intelligence

May 24, 2018

Program # 3: UFO Puzzle - Solved with SWI-Prolog

Problem Statement:

UFO Invasions or not? Last week, four UFO enthusiasts made sightings of unidentified flying objects in their neighborhood. Each of the four reported his or her sightings on a different day, and soon the neighborhood was abuzz with rumors of little green men. By the weekend, though, the government stepped in and was able to give each person a different, plausible explanation of what he or she had "really" seen. Can you determine the day (Tuesday through Friday) each person sighted a UFO, as well as the object that it turned out to be?

1. Ms. Langone made her sighting at some point earlier in the week than the one who saw the balloon, but at some point later in the week than the one who spotted the Frisbee (who isn't Mr. Hugh).
2. Friday's sighting was made by either Mr. Barnes or the one who saw a clothesline (or both).
3. Ms. Demetri did not make her sighting on Tuesday.
4. Ms. Langone isn't the one whose object turned out to be a water tower.

Represent this puzzle in Prolog, SWI Prolog is recommended. State the day of each sighting, the person doing the sighting, and the object allegedly sighted and what that object actually was (or at least what the government claimed it was).

Analysis of the Given Statement:

As Given:

The four UFO enthusiasts are Barnes, Hugh, Langone, and Demetri.

The four objects are Frisbee, balloon, water tower, and clothesline.

The four days are Tuesday, Wednesday, Thursday, and Friday.

Also, the provided information includes:

1. Langone's day was earlier than balloon's day.
2. Langone's day was later than Frisbee’s day.
3. Hugh didn't spot the Frisbee.
4. Friday was either Barnes's day or clothesline day (or both).
5. Demetri's day wasn't Tuesday.
6. Langone didn't spot the water tower.

Strategy:

From the given info we can see that

1. Langone didn't spot the balloon(a) or the Frisbee(b) or water tower(f).

Then, Langone must have spotted the clothesline.

1. Since Langone's day can't be Friday (a = earlier than Friday), then Friday must have been Barnes's day(d).
2. The Frisbee wasn't spotted by Langone(b) or Hugh(c). Nor was it spotted by Barnes since Barnes's day was Friday(B) and the Frisbee wouldn't have been spotted on Friday(b). Therefore, Demetri spotted the Frisbee.
3. The Frisbee's day is at least two days before balloon's day (a + b). So, the Frisbee’s day must have been either Tuesday or Wednesday. Again, Demetri spotted the Frisbee(C) and Demetri's day was not Tuesday(e). Therefore, the Frisbee’s as well as Demetri's day was Wednesday.
4. Since Langone's day is after Wednesday(b+D) and not Friday(B), then it must be on Thursday.
5. It follows from E that the balloon's day was Friday(E + a).
6. The balloon was spotted by Barnes(B + F).
7. From A, F, and G, we can conclude that Hugh spotted a water tower on Tuesday.

Solution:

The above inferences can be summarized as follows

* Hugh spotted a water tower on Tuesday.
* Demetri spotted a frisbee on Wednesday.
* Langone spotted a clothesline on Thursday.
* Barnes spotted a balloon on Friday.

Instructions to Run the Program:

Considering SWI-Prolog is already installed in the host machine, run these from the command line to see the output:

* swipl // to invoke SWI-Prolog
* ['ufoPuzzle.pl']. || swipl -f filename.pl // to load program; do not forget the dot at the end of the command.
* puzzle. // to run the program to see the output.
* halt. // to exit SWI-Prolog.

The Program:

The above explained facts, logic, and inferences are represented in SWI-Prolog as below:

SWI-Prolog representation of the UFO Puzzle

remainder(X, [X | Ds], Ds). % X is the remainder of Ds such that X is the first element of a list

/\* prolog clause with '\_' or "don't-care" variable, which will match anything.

It is basically defining the 'remainder' rule to be used for 'before' rule. It says X is the the remainder of Rs such that X is the first element of a list Rs, and the rest is Ds if remainder. . . \*/

remainder(X, [\_ | Ds], Rs) :-

remainder(X, Ds, Rs).

/\* Checks that `X` comes before `Y` in the list `Ds`b.

It is basically defining the 'before' rule with arbitrary variable list Ds, Rs where Rs maps to 'Days' data and Ds and X maps to each element in 'Days'\*/

before(X, Y, Ds) :-

remainder(X, Ds, Rs),

member(Y, Rs). %This says if Y is a member of any list Rs(the second argument)?

members([], \_).

members([X | Xs], Ds) :-

member(X, Ds),

members(Xs, Ds).

puzzle :- % specifying the puzzle in a sequence of steps that need to be done in sequence.

/\*Defining Days data structure that will hold all available days of the week at position 0, the person at positon 1, and the item seen at position at 2\*/

Days = [[onTuesday, \_, \_], [onWednesday, \_, \_], [onThursday, \_, \_], [onFriday, \_, \_]],

before([\_, ms\_Langone, \_], [\_, \_, balloon], Days), % the fact as given; in the days data structue Langone appears before ballon

before([\_, \_, frisbee], [\_, ms\_Langone, \_], Days), % the fact as given

(member([onFriday, mr\_barnes, \_], Days); % more steps of sequence to fill up Days

member([onFriday, \_, clothesline], Days); % these facts are deduced from the given info

member([onFriday, mr\_barnes, clothesline], Days)),

members([[\_, ms\_Langone, \_],[\_, mr\_barnes, \_],[\_, mr\_hugh, \_],[\_, ms\_Demetri, \_]], Days), % Deduced from the analysis

members([[\_, \_, balloon], [\_, \_, frisbee], [\_, \_, clothesline], [\_, \_, water\_tower]], Days),

member([\_, NOT\_mr\_hugh, frisbee], Days), NOT\_mr\_hugh \= mr\_hugh,

member([onTuesday, NOT\_ms\_Demetri, \_], Days), NOT\_ms\_Demetri \= ms\_Demetri,

member([\_, NOT\_ms\_Langone, water\_tower], Days), NOT\_ms\_Langone \= ms\_Langone,

write(Days), nl. % prints out Days