Project 2
Using Prolog as a Knowledge-Based System
CSE 4/563, Knowledge Representation
Due Monday, April 6, 2009

Professor Shapiro

March 16, 2009

1 Project Statement

This project is adapted from Russell & Norvig's AI texts [1,2]. You are to specify and formalize a domain sufficient to answer a series of questions about the information in the sentence (which I will call sentence $\mathcal{S}$):

> *Yesterday, John went to the Wegmans Alberta Drive supermarket and bought a pound of tomatoes and two pounds of ground beef.*

You are then to load the knowledge base (KB) and your formalization of $\mathcal{S}$ into SICStus Prolog [3], and have it answer the following questions. Each question is followed by the correct answer in square brackets.

1. *Is John a child or an adult? [Adult]*

2. *Does John now have at least two tomatoes? [Yes]*

3. *Did John buy any meat? [Yes]*

4. *If Mary was buying tomatoes at the same time as John, did he see her? [Yes]*

5. *Are the tomatoes made in the supermarket? [No]*

6. *What is John going to do with the tomatoes? [Eat them]*

7. *Does Wegmans sell deodorant? [Yes]*

8. *Did John bring any money to the supermarket? [Yes]*

9. *Does John have less money after going to the supermarket? [Yes]*   [1,2]

2 Discussion

You need to formulate your KB so that the questions can be answered correctly, but you must abide by the following rules:

- Your KB must not contain any fact that is just an instance of one of the questions.

- Each question must require at least two rule firings (resolution steps).

- Background knowledge must be in terms of general categories which contain the individuals mentioned in $\mathcal{S}$.

- Your background knowledge need not be strictly true, but generally reasonable and appropriate for answering the questions.

For example, do not put into your KB the fact that John is an adult. Instead think about why you might conclude from S that John is an adult, and put that information in your KB. Similarly, don't put into your KB the fact that Wegmans sells deodorant, but categorize deodorant into some class of goods that supermarkets sell. Also, don't change any question to make it closer to S. For example, question 7 asks about "Wegmans", not about "the Wegmans Alberta Drive supermarket". Think about the relation between "the Wegmans Alberta Drive supermarket" and "Wegmans", and the relation between "the Wegmans Alberta Drive supermarket" and supermarkets.

For another example, don't put into your KB the fact that tomatoes are not made in the supermarket. Rather you should include some categories of goods that are made in supermarkets, as well as some that are not. Even though "pure" Prolog consists only of Horn clauses, there is a way to make use of a kind of negation, called "negation by failure", in most actual Prolog systems, including SICStus Prolog. Here is an example in which canaries and penguins are the only kinds of birds:

```
| ?- [user].
% consulting user...

| % Opus is a bird.
    bird(opus).

| % Canaries are birds.
    bird(X) :- canary(X).

| % Penguins are birds.
    bird(X) :- penguin(X).

| % Tweety is a canary.
    canary(tweety).

| % A bird that is not a canary is a penguin.
    penguin(X) :- bird(X), !, \+canary(X).

| % If something is a canary, it is not a penguin.
    penguin(X) :- canary(X), !, fail.
|
% consulted user in module user, 0 msec 1016 bytes
yes

| ?- % Is Tweety a penguin?
    %   Since Tweety is a canary, she is not a penguin.
    penguin(tweety).
no

| ?- % Is Opus a penguin?
    %   Since Opus is a bird, but not a canary, he is a penguin.
    penguin(opus).
yes
```

The zero-argument predicate `fail` always fails. The negation operator, `\+`, (called **not** in [4]) is a second-order predicate that succeeds if its argument fails. For a description of the cut operator, `!`, see [4]. (Also see [3] for all of these operators.)

Representing the KB, $\mathcal{S}$, and the questions in Prolog presents some difficulties of its own, since Prolog does not include full First-Order Logic. You will design your KB with that in mind, but you might find Prolog constraining when translating $\mathcal{S}$ and the questions into Prolog. Do your best. For example, there is no way in Prolog to ask the literal question, "Is John a child or an adult?" You should not do this in two questions, "Is John a child?" and "Is John an adult?". Instead, you might ask what age bracket John is in.

Notice that time is involved in this project, for example,

> *"Yesterday..."* [$\mathcal{S}$]
> *"Mary was buying ... at the same time as ..."* [question 4]
> *"What is John going to do ..."* [question 6]
> *"Did John bring any money ..."* [question 8]
> *"... have less money after going ..."* [question 9]

You should take time seriously in every statement, question, and KB rule for which it's at all relevant.

Notice that to answer question 2, you will probably need to weigh a sample tomato, and have Prolog do some numerical calculation. Here's an example of Prolog doing arithmetic:

```
| ?- [user].
% consulting user...
| times(X, Y, Z) :- Z is X * Y.
|
% consulted user in module user, 0 msec 232 bytes
yes
| ?- times(2,3,X).
X = 6 ?
yes
```

See the SICStus Manual [3] for more information on doing arithmetic in Prolog.

3 Deliverables

As it says on the CSE 4/563 web page, you are to

> "hand in a paper, produced using a document formatting program such as LaTeX or Microsoft Word, and printed on 8.5 by 11 inch paper, stapled in the upper left-hand corner, with a title, your name, and other identifying information at the top of the first page (Do not use the header page automatically produced by the printer), plus a well-documented listing and run of your program. (Do not enclose your paper in a folder or cover.)" [5]

The paper is due at the start of class on the date shown at the beginning of this document.

In addition to the paper, you are to submit (using submit_cse463 or submit_cse563) your program, so that it can be run and checked if the instructors choose. Name your file proj2KB.prolog. You are to submit your KB file by one half-hour before the start of class on the date shown at the beginning of this document, so that you can get to class on time to hand in the paper.

Your KB file is to include English and Prolog versions of the formalization of your domain, and a commented sequence of Prolog queries. Each query should be preceded by a comment including an English version of the query, and an English summary of the reasoning your program goes through to answer the query. One example of this was the canary *vs.* penguin example shown above. For another, here is a listing of a properly formatted file of the "fish" example from lecture:

```
/* fish.prolog
 * Adapted from Rich & Knight, Artificial Intelligence, 2nd Edition,
 *              McGraw-Hill, New York, 1991, p. 192.
 *    by Stuart C. Shapiro
 */
% Cats like to eat fish.
likesToEat(X,Y) :- cat(X), fish(Y).

% Calicoes are cats.
cat(X) :- calico(X).

% Tuna are fish.
fish(X) :- tuna(X).

% Charlie is a tuna.
tuna(charlie).

% Herb is a tuna.
tuna(herb).

% Puss is a calico.
calico(puss).

/* Questions
 * =========
 *
 * What does Puss like to eat?
 *    Puss is a calico and calicoes are cats.
 *    Cats like to eat fish, tuna are fish, and Charlie and Herb are tuna.
 *    Therefore Puss likes to eat Charlie and Herb.
 likesToEat(puss, X).
 */
```

3.1 The Paper

Your paper should have the following parts:

1. Descriptive title

2. Author identification

3. Introduction: general description of the project

4. Domain and Knowledge Base

   (a) Syntax and semantics of atomic symbols
   (b) English and formal presentation of information in $\mathcal{S}$ and in the KB.

5. Demonstration run, properly commented and formatted

6. Acknowledgments as needed

7. References as needed

4 Grading

Project grading will be according to the following table.

	CSE 463	CSE 563
KB		
The portion of the KB used for each question follows the guidelines of Sect. 2 (9×4)	36	36
Statement $\mathcal{S}$ is reasonably formalized	6	6
KB Subtotal	**42**	**42**
Questions		
Each question is reasonably formalized (9×3)	27	27
Each question is correctly answered ($9 \times (2$ or $1)$)	18	9
Questions Subtotal	**45**	**36**
Paper		
Paper format	1	2
General project description	1	2
English description of KB	2	3
Presentation of syntax and semantics of atomic symbols	3	5
Formatting & presentation of demonstration run	2	3
Correct use of quotation and citations	2	4
Acknowledgments and References	2	3
Paper Subtotal	**13**	**22**
Project Total	**100**	**100**

References

1. Stuart Russell & Peter Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, Upper Saddle River, NJ, 1995, pp 263–264.

2. Stuart Russell & Peter Norvig, *Artificial Intelligence: A Modern Approach, Second Edition*, Prentice Hall, Upper Saddle River, NJ, 2003, pp 372–373.

3. SICStus, SICStus Prolog Manual,
   `http://www.sics.se/sicstus/docs/latest/html/sicstus.html/`.

4. Ronald J. Brachman & Hector J. Levesque, *Knowledge Representation and Reasoning*, Elsevier Science, 2004, pp 104–109.

5. Stuart C. Shapiro, "UB CSE2/563," `http://www.cse.buffalo.edu/~shapiro/Courses/CSE563/2009/`.