CSCI460 – Introduction to Artificial Intelligence
Instructor: Dr. K. Narayanaswamy

Assignment 4 – Prolog
Due: 04/6/2012 11:59:59pm

The purpose of this assignment is to give you experience in dealing with GNU Prolog/gprolog systems and working with practical systems that support Horn knowledge bases and inference by using unification and modus ponens/backward chaining.

Problem: During the Gold Rush days, many people from all over North America moved out west for their chance to strike it rich. The following clues below provide some information about 6 traveling gold miners who settled in San Francisco during the 1860's.

- Each gold miner settled in a different type of dwelling: tent, shack, hotel, cave, cabin or house.
- Each dwelling has exactly one person living in it.
- Each gold miner was born in a different month. They each play a certain kind of instrument and also brought a unique animal along with them. No gold miner has the same animal, plays the same kind of instrument nor was born in the same month.

Clues:

1. The Canadian lives in the house.
2. The Coloradan brought some chickens with him.
3. The hotel is to the left of the cave.
4. The gold miner that owns the goat lives next to the one that owns the chickens.
5. The gold miner who lives in the hotel was born in September.
6. The Mexican plays the guitar.
7. The gold miner that plays the banjo was born in April.
8. The gold miner who plays the flute brought along a horse.
9. The Alaskan was born in August.
10. The gold miner who lives in the tent plays the violin.
11. The gold miner that lives in the third dwelling was born in May.
12. The Texan lives in the first dwelling.
13. The gold miner that plays the harmonica lives next to the one that has a donkey.

14. The gold miner that owns the cow lives next to the one that plays the violin.
15. The gold miner that was born in January plays the accordion.
16. The Nebraskan lives in a cabin.
17. The Texan lives next to the shack.
18. The gold miner that plays the harmonica has a neighbor that was born in March.

Questions:

a) According to Prolog, how many solutions are consistent with the rules?
b) Who could possibly own the dog according to the answers provided by Prolog?

Programming:

1. Please construct a prolog file goldminer.pl to represent the knowledge base.
2. Please use GNU prolog or gprolog to solve the above riddle and to find the answers to both questions above. Write out your query sentences and their outputs in the Readme.txt file.

This assignment will give you experience in dealing with the GNU Prolog / gprolog systems and give you experience in working with a practical system that supports Horn knowledge bases, and supports inference by using unification and modus ponens/backward chaining.

Hints:

There are many ways to solve this problem. One of the elegant solutions is to formulate the problem using the Prolog list primitives. This involves the ability to combine elements into Lists and to pull them apart using Prolog's built-in | operator and "member" function.

The class notes and tutorials on the web site show how to use these operators. We have described the relevance of Prolog list utility to solving this problem more easily.

Lists are useful in this assignment for two reasons:

a) The configuration of gold miners in the problem is best represented as a List because the problem mentions sequences or positions of the gold miners.

b) We also prefer to think of each gold miner as a list:
[origin/background, animal, instrument, month, dwelling type] for example.

So, the list of the gold miners can be represented as a list of lists, containing 6 elements.
Where you know the facts, you can fill them in. For example, if the Alaskan owned the donkey, you can represent that fact as:
[Alaskan, donkey, _, _, _]
where "_" character is used as a wild card that matches any value in the slot where it appears, rather than introducing a variable that you do not use. Or if the Nebraskan lives in the house, you can represent that as:
[Nebraskan, _, _, _, house]
and so on.

Once you fill out all the information using the above representation, Prolog will help you answer the other questions through backward chaining! So, basically Prolog will do the heavy lifting, if you represent the problem correctly.

You might find the following helper concepts to be useful as well. They are used to define "next_to" and "left" and "right" -- which are used in the problem statement. You can use "iright" for that.

```
next_to(X, Y, List) :- iright(X, Y, List).
next_to(X, Y, List) :- iright(Y, X, List).
iright(L, R, [L | [R | _]]).
iright(L, R, [_ | Rest]) :- iright(L, R, Rest).
```

Grading Policy: (Total Points: 100)

Program Correctness (95 pts):
1. Correct representation of each clue: 3 pts*18 = 54 pts
2. Correct query sentence and output for Question A: 26 pts
3. Correct query sentence and output for Question B: 15

We will test your program by loading the file you provide into gprolog on aludra.usc.edu. You must make sure that the file you submit works on that interpreter. Otherwise, the maximum credit you get will be 30% for program correctness.

Readme.txt (5 pts):

1. A brief description of the program structure, and any other issues you think that will be helpful for the grader to understand your program.
2. Please include your name, student ID and email address on the top.
3. Your query sentences and solutions for Questions A and B.
4. You must submit a program in order to get any credit for the Readme.txt. In short, if you submit ONLY a Readme.txt file you will get 0.

Submission: Follow the same submission procedures as you did for other homework assignments.

For policies on late submissions, please see the Syllabus from the course home page. These policies will be enforced with no exceptions.