CS 4100/5100 – Foundations of Artificial Intelligence

Assignment 1: Knowledge Representation and Logical Reasoning[1]

Due: September 27th, 5:59pm

Learning Objectives

- Gain familiarity with logic programming
- Learn how to represent knowledge in formal logic
- Use Prolog's inference capabilities to solve problems

Resources

- SWI-Prolog is available on the Windows CCIS lab machines, under Start > All Programs > Department Applications > CCIS Applications > Course Software > SWI-Prolog (or by typing SWI-Prolog into the search bar on the start menu).
- If you would like to run SWI-Prolog on your own machine, you can download it from here: http://www.swi-prolog.org/
- There is an excellent Prolog tutorial online here: http://www.csupomona.edu/~jrfisher/www/prolog_tutorial/contents.html
- We will be going over how to write programs in Prolog in class on September 20th.

Assignment Description

For this assignment, you will be creating a little adventure game in Prolog. Adventure games are a genre of game that focuses on players moving around a small space and solving puzzles to be able to make progress. Key examples of the genre are the Zelda series and the Monkey Island series.

I have provided sample code that you can work from. This code is by no means complete – you will be doing extensive modifications to it for your assignment. It provides code for moving between two rooms, picking up objects, and printing a list of objects that are in the room.

You are not required to use the starter code.

Grading Criteria

Your adventure game is your own creation, I encourage you to use your imagination and have fun with it! Your assignment will be graded based on the features you have added. Aim for having 10 rooms and 5-10 unique objects that can interact with each other. Undergraduate students are required to reach a total of 80 points; graduate students are required to reach a total of 100 points. A maximum of 10 percentage points extra credit will be rewarded for adding extra features. I reserve the right to assign a small amount of extra credit to assignments that I feel are particularly interesting or creative: yes, this is

[1] This assignment is adapted from David Matuszek's similar assignment at UPenn: http://www.cis.upenn.edu/~matuszek/cis554-2011/Assignments/prolog-02-text-adventure.html

subjective. Points will *not* be deducted for boring games (but please consider we'll be playing 50 of these, so try to make it interesting!). You are required to have a game that is possible to understand and play. Feel free to recreate games or stories you have seen in other media (e.g. a scene from Monkey Island or part of an episode of Star Trek). Points may be deducted for code that is hard to read or commented poorly.

You are *required* to implement the following features:

- **(5 points) Instructions –** the start/0 rule I have provided in the sample code should be modified to also print a list of instructions – the goal of your game and the verbs available for use.
- **(5 points) Room reporting** – when you enter a room, the game should print a description of the room, a list of all objects contained in the room, and a list of all paths out of the room.
- **(10 points) Conditional movement** – you are not allowed to move in a certain direction unless you have satisfied some condition. For example, a door might be locked and require a key for entry, or a passageway might be dark and require a flashlight to move through.
- **(15 points) Main antagonist –** another character in the game than the main character who acts to block the player's progress. Perhaps it is a monster that attacks when you enter a room and you need a special weapon to defeat it, or a little boy who is crying and must be soothed with a toy hidden elsewhere in the world.
- **(15 points) Limited resources –** a collection of resources the player holds (hit points, flashlight batteries, torches, rope for climbing) that you can run out of and are optionally renewable by exploring certain areas in the game. This should require you to use arithmetic in prolog.
- **(20 points) Magic teleportation device** – your game should have at least two disconnected areas – i.e. there is no way to walk normally between them. For example, a spaceship and the planet it is orbiting. The magic teleportation device can transport the player from one area to a named, player-specified location in another area, or vice versa. This device **cannot** transport you between locations that are connected and traversable. For example, the transporter can take you from the spaceship to the planet and back, but cannot transport you between areas on the spaceship or between areas on the planet. If there is a room blocked due to not satisfying the conditions for movement, the magic teleportation device *can* take you to it. You must make sure when the player tries to use the device that it is impossible to walk normally between the start and end locations.

Here are some ideas for other features and their point values:

- **(5 points) Loot container** – a locked container of some sort (treasure chest, cabinet) that contains something of value to the player. It can only be opened with a specific key.
- **(5 points) Skeleton key** – a key or other device that can open any lock in the game.
- **(5 points) Inventory** – a command that lists all of the objects that the player is holding.
- **(Max 15 points) Custom object –** a special object in the game that has a lot of different ways you can use it with different kinds of objects in the environment. For example, a scanner that can look through doorways, find medical problems with characters, and tell whether a box has a trap on it. 5 points per type of object it can interact with.
- **(10 points) Win/loss condition –** you win when you have found all the gold nuggets in the world, or you lose when you accidentally walk out of the airlock. If you have won, you can keep exploring but you need to give special messaging to the player. If you have lost, the player must restart.

- **(Max 15 points) Multi-action sequence –** you are required to perform a sequence of actions to receive a particular reward. For example, you are required to acquire some sleeping pills and grind them with a coffee grinder. Then you must mix the ground up pills into a drink to give your evil stepmother. Then she falls asleep so you can sneak out of the house.

Have an idea for something more complicated and not sure what category it fits into? Suggest it on Piazza to find out if it's feasible and get ideas on how to implement it, and we will try to give it a point value. **Do not ask me via email, the volume of questions becomes overwhelming and I will not be able to answer**.

Submission Instructions

Submit a zip file containing:

- Your *commented* .pl or .pro file containing your game.
- A text or PDF file containing a sample log of your game being played. Copy and paste this from the interpreter window into the document.
- A readme file with your name, how many late days (if any) you wish to use, and a list of any people you received help from.

All materials must be submitted through Blackboard. **Assignments emailed to me will not be accepted.**