Csc466 Assignment: Project Finding Framework

This document is intended to facilitate your efforts to formulate a reasonable research/programming project for this course. It consists of a collection of rather eclectic notes, unified by the idea of assuring that each of you is successful in this course. I strongly encourage you to engage with this document like you mean it, and to consider the contemplation of these notes to be an important matter.

On Tuesday, while you are presenting your two candidate research project topics, I will ask you how you might relate each of your candidates in one way or another to this collection of notes. You are welcome to generate a little document for this assignment which reflects upon these suggestions with respect to your candidate research/programming projects. That said, you are not required to do so.

Think about going with something that is relatively well defined

1. Perhaps a state space problem solver for a problem that presents some interesting modeling considerations.
2. Perhaps a genetic algorithm for a relatively real problem, or just something fun!
3. Perhaps a puzzle or a game? Puzzles and games have the wonderful property, that while they are well defined, so that you can clearly demonstrate what you are doing, they afford any number of interesting approaches! You might even consider exploring the limits of an exhaustive approach to a game with a heuristic approach.

Think about avoiding projects that contain too much domain specific knowledge

Think about avoiding projects that require, for example, knowledge of a body of mathematics that you do not already possess in substantial measure.

You don't have time to learn a body of mathematics and harness its power within a Lisp program within 12 weeks or so. Unless you already know a something about Markov processes, for example, you might not want go there! Same for many other mathematical domains, and non-mathematical domains, as well.

Think about what you are contributing to the code

This is not intended to be a project in which you merely use computational tools that others have written, or merely integrate in a loose fashion large modules of code that others have written. You

are certainly welcome to use libraries of code, but you must be very clear about what you will be contributing in terms of the coding for your project.

AI text book problems

Think about the possibility of doing a modest project that has been detailed at a conceptual level in some AI textbook.

A number of really great text books talk about essential ideas and techniques associated with AI at a conceptual level. The text by Rich and Knight comes to mind, but there are many others as well. You might like to find a topic within a book like this and simply explore it from a computational perspective. Think of scripts, for example. You might craft a project that features scripts. Or frames. Or a semantic network. Or a conceptual-dependency graph. Or you might like to try to implement a rule-based expert system shell. Or you might like to implement a minimax game player with alpha-beta pruning. There are all sorts of possibilities. I think it would be preferable to explore some basic element of AI from a computational perspective, than to try to do something more advanced that even teams of programmers with PhDs in computer science in high-powered companies struggle to make progress on.

AI textbook exercises

Think about the possibility of doing a project of appropriate size and difficulty that has been proposed as an exercise in some AI textbook

There are a range of programming text books within the field of computer science that explore AI topics. Perhaps you could find a case study in one of these books and craft a project that involves translating parts of the study and/or adapting parts of the study in one way or another.

Do something that you can do!

Appreciate that you can't afford to misjudge what you are capable of doing

It is best to pick something that you can make some immediate progress on, some immediate tangible progress! You want to craft a project that can be incrementally scaled up. As you wrap up one phase, you can begin the next. Your goal should probably be to have 10 or 12 phases, each of which you can complete in 6 or 8 hours, max. Think about making discernible weekly progress. If after one week you haven't got your week one goal met, there is real trouble!

Imagine an incremental approach to your work

Think about how you will define an incremental approach to your work. Yes, I know that I mentioned this in the last note, but really significant matters deserved to be emphasized by means of repetition and elaboration.

How will you define the stages of your project? What will be its "Hello World" and the first extensions to its "Hello World"? What are the well defined points interest that you can set as goals to complete as the semester progresses. You need to be able to write these down, and then translate them into well defined tasks that you can demo and discuss in relation to the other points of interest.

A few modest suggestions ...

1. Rush Hour Traffic Jam puzzle (state space problem solver)
2. Bridge bidding program or bridge playing program (or some other card game)
3. A board game (other than Dobo)
4. A question answering system based on a well-defined domain of knowledge (perhaps a system oriented towards aspects of food or art or music or movies)
5. A natural language like interface to a system of some sort (perhaps a game)
6. The generation of some form of restricted texts, whether language texts or pictorial texts or musical texts (perhaps using a genetic algorithm)

Regardless of what problem you choose to focus on, do your best to feature interesting knowledge representations in your work, and elegant means by which express knowledge at multiple levels of abstraction.