CSC 425/525 Programming Assignment #3
Due Date: Monday, April 3

This assignment is loosely defined and it will be up to you to select a problem, domain and implementation method. In this assignment you are to implement some form of reasoning system that uses uncertainty handling. Your options for uncertainty handling are:

- a justification truth maintenance system based on a series of rules some of which include unless clauses where your system represents the rules in a graph form and implements the three operations of inspection, modification and updating.
- A rule-based system using MYCIN-like certainty factors
- A fuzzy logic system which built-in membership functions to map input values to real numbered values, a small set of rules, a combining operation and a translation operation to convert the combined results into actions.
- A Demster Shaefer diagnostic reasoned.
- A Bayesian-based rule-based system that computes probabilities.
- A reasoned that uses a classification hierarchy where you might employ Bayesian probabilities, rules or feature-based pattern matching to determine whether a given node is relevant or not.
- Some form of set covering reasoned which generates all possible explanations for a given set of data and then uses some method to evaluate and select the most likely overall explanation.
- A Bayesian network reasoned that applies the chain rule.

NOTE: most of the problems that will require any of the above uncertainty handling approaches will be interpretation/classification/diagnosis/recognition type tasks but you might be able to come up with a different type of task that can use one of these approaches.

You can implement your system in Clips or Jess, or directly in a high level language of your choice. If your rule-based program (assignment #2) is amenable to adding certainty factors, you can build on top of that system but that is not required. Pick a domain that you know reasonably well to have somewhat correct knowledge (it doesn't have to be perfect, complete or even real accurate as long as it makes some sense).

Hand in:

1) the listing of your program
2) several run traces (at least 3) each of which proves or performs a different problem within the domain
3) a report describing your domain and knowledge, why you selected the method of uncertainty that you did, difficulties you had in implementation, enhancements that you feel should be made if you were going to spend more time on it, and whether the uncertainty approach selected fits the problem you implemented or whether you think one of the other approaches would be better, and why.

Undergraduates can work in groups of up to 3, graduate students must work alone.