

# Project Instructions and Suggestions

## General

- The results of your project are to be captured in a written document of 10 to 20 pages or an oral presentation to the class, with slides.
- You may, if negotiated with me in advance, use your work to extend or complement the work you do in some other context.
- You are welcome to work together in teams, just like in the real world. Separate written work and oral presentation is to be done, however, as everyone is to have that experience.

## Schedule

The schedule is to be determined. In any event, some sort of proposal will be required before spring break.

Probably, students doing projects will come together outside of normal class hours to debug presentations.

## Varieties

Projects can be, but are not limited to, any of the following types:

- **A simulated area exam.** You read several papers and do a scholarly critique. Part of the process is to pick a set of dimensions (ugh) along which to compare and contrast the papers, recalling the spider diagram approach. You conclude with a description of the degree to which you were moved (yippee), and why.

For example, you could contrast the work of Pearl (Bayes nets) with that of Zedah (fuzzy sets), viewed as efforts aimed at explaining commonsense reasoning; or you could compare various papers on commonsense reasoning; or you could examine the contrasting views of various anti and pro AI philosophers, such as John Searle (Berkeley) and Daniel Dennett (Tufts). When you have a rough idea of what you might want to do, I can perhaps recommend particular papers.

The simulated area exam is your safety net; if your other attempts do not work out, or you just cannot identify something you want to do, this is the sure-fire, finite-time, still highly educational, way out.

- **A research proposal.** You propose a mechanism for performing some clearly defined task. You write your idea up in the form of a thesis proposal or funding proposal, with all the usual parts expected of such proposals, which will be discussed in class.

One way to generate ideas is to write all the paper names on slips of paper, put them in a hat, and draw two at a time, asking if they make any sense. A famous mathematician (who?) claims he does new mathematics in a roughly analogous fashion.

For example, you might draw the Rao paper and the Borchardt paper and ask if Borchardt's representation might be useful in describing Rao's routines.

If you get inspired and passionate about something, you might just find yourself with a real thesis proposal when you are done with this. After all, Dave Huffman discovered Huffman coding while doing a term paper in Bob Fano's information theory class.

- **A reimplementation.** You pick one of the papers that describes an implemented idea and you reimplement, reporting on the surprises that invariably emerge when you understand something at the level required to write code. You also report on improvements you make.

For example, you could redo the work reported in the Yip and Sussman paper, the Kirby paper, or the Borchardt paper. Note, however, that of these only the Yip and Sussman paper is sufficiently specific to enable their experiments to be confirmed.

A reimplementation is risky, because everyone seems to overestimate what they can do in two months. On the other hand, if sufficient detail is given, and you are a good coder, this can be a lot of fun. I hacked up a simple version of Yip and Sussman in a weekend for a class demonstration.

- **A complete research project.** You both have idea and do the implementation, and report on your experimental results. This is beyond the call of duty, inasmuch as your project is not all you do in the subject. You might accomplish more by working in teams, but then you are exposed to weak-link problems exacerbated by the part-time, conflicting pull nature of student labor. Caution is advised, along with especially clear and modest goals.