

Class 09: Prolog (2)

Held: Friday, 11 February 2011

Summary: We consider Prolog's lists and mathematical operations.

Related Pages:

- EBoard.
- Lab: Prolog (2).
- Reading: Tate 4.3.

Notes:

- HW4 to be posted some time tonight. (I had started to design one assignment, but discarded it after spending too much time trying to solve it myself.)
- EC for CS Table today. Reading: Wikipedia on Nearest Neighbor.

Overview:

- Math in Prolog.
- Lists in Prolog.
- Concatenating Lists.
- Lab.

Math in Prolog

- While we could define all of the mathematical operations ourselves (hmmm ... there's an interesting homework problem), it is more efficient to have them built in to Prolog.
- The basic math model in Prolog is that you can write *thing = expression*.
- Prolog evaluates the expression.
- If the thing is a constant, it compares the two results and returns true or false depending on the result of the comparison.
- If the thing is a variable, it updates the table of variable values and returns true.

Lists in Prolog

- Prolog has lists built from cons cells, plus lots of syntactic sugar.
- `[]` is the empty list.
- `[H|T]` is what you might think of as 'cons'ing H and T.
- `[X,Y,Z]` is a three-element list.
- `[X,Y|T]` is a list with at least two elements (X and Y) and remainder T.

Concatenating Lists

- Two approaches (one obvious, one the funky Tate approach).
- We'll look at both.

Lab

- Do the lab.
- Be prepared to reflect.