CS 61A Week 14

Topic: lazy evaluator, nondeterministic evaluator

Reading: Abelson & Sussman, Section 4.2, 4.3

A version of the lazy evaluator is online in `~cs61a/lib/lazy.scm`

A version of the nondeterministic evaluator is online in `~cs61a/lib/ambeval.scm`

A modified version of the nondeterministic evaluator, based on the vanilla metacircular evaluator rather than on the analyzing evaluator, is online in `~cs61a/lib/vambeval.scm`

Homework:

There are a lot of exercises this week, because we are covering a lot of material. These are all great exercises, from which you'll learn a lot, but for those who find the homework time-consuming, we're dividing them into two categories, crucial and less-crucial. Again, I recommend all of them!

crucial: 4.25, 4.26, 4.28, 4.42, 4.45, 4.49, 4.50, 4.52

less crucial: 4.30, 4.32, 4.33, 4.36, 4.47, 4.48

Note: Part II of the fourth programming project is also due next week.

Extra for experts:

Exercise 4.31 in Abelson and Sussman. This exercise doesn't require great brilliance, but it's a lot of work and involves a lot of debugging of details. On the other hand, completing this exercise will teach you a lot about the evaluator.

Fix the `handle-infix` procedure from project 4 to handle infix precedence for arithmetic operators properly. That is, multiplications and divisions should be done before additions and subtractions. Comparison operators like `=` come last of all.

Unix feature of the week: `!`, `history`

Emacs feature of the week: `C-x (`, `C-x )`, `C-x e` (keyboard macros)