Class 24: Revisiting Lists

Held: Wednesday, 7 October 2009

Summary: We explore further details of Scheme's lists. Lists can be a key mechanism for structuring data. We also consider how lists can be used to represent drawings.

Related Pages:

* EBoard.
* Lab: Exploring Lists.
* Reading: Building Data Structures with Heterogeneous Lists.
* Due: Assignment 5: A Miscellany of Image Models.

Notes:

* A few problems on the homework were a bit more difficult than we anticipated. I apologize. (However, I think you're probably now a bit better with building anonymous procedures, and I think that's a good thing.)
* Friday's quiz will include conditionals, `image-compute`, and the new list operations, like `cons`, `car`, and `cdr`.
* Assignment 6 is now ready, and is due Wednesday of next week.
* Looking ahead: No large assignment will be due the week after break.
* Looking further ahead: GIMP (the other one) presents *Rent* (Nov. 13-15).

Overview:

* Lists, Revisited
* New List Operations
* Drawings as Values

Lists, Revisited

* Original notion of lists:
    * Collection of values
    * Homogeneous: All the values in a list are the same type
    * Used primarily with `map` and `for-each`
* Extending that notion
    * Can also be *heterogeneous*: Values in the list may have multiple types
    * Can examine individual elements with `car`, `cdr`, and `list-ref`.
    * Can be extended (creating new lists) with `cons`.
* Why is the new view important?
    * It will soon give us the opportunity to manipulate lists in new and interesting ways.

Implementing Drawings

- We can (and do) use lists to implement drawings.
- The reading considers a simplified representation, in which the width and height of a drawing are the same.
- But the general process is the same
  - ○ Group the values in a list
  - ○ When we need to render or manipulate a drawing, use the list operations to extract parts.