

Class 18: Recursion Basics

Held: Friday, 28 September 2007

Summary: We begin our exploration of *recursion*, the most general form of repetition available in Scheme. You can use recursion to both build and iterate over different kinds of values. Today, we explore how you use recursion to build simple numeric values from lists, and how to remove elements from lists.

Related Pages:

- EBoard.
- Lab: Recursion Basics.
- Reading: Recursion Basics.
- Due: Exam 1: Scheme Basics.

Notes:

- Assignment 8 is now available (and due Tuesday).
- We'll try something different today: We'll start with lab. I will go over the key ideas in Monday's class.
- But yes, there is a reading for Monday. It's about a different kind of recursion, tail recursion.
- Extra Credit for going to the Science Poster Session on Saturday, and seeing Amanda M's poster, or Emily's poster, or anyone else in the class's poster.

Overview:

- The idea of recursion.
- A sample recursive procedure: `sum`.
- Another example: Filtering.

Copyright © 2007 Janet Davis, Matthew Kluber, and Samuel A. Rebelsky. (Selected materials copyright by John David Stone and Henry Walker and used by permission.) This material is based upon work partially supported by the National Science Foundation under Grant No. CCLI-0633090. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.