Class 08: Procedure Definitions

Held: Wednesday, 6 September 2006

Summary: Today we begin to consider how you can write your own procedures and why you might do so.

Related Pages:

- EBoard.
- Lab: Defining Your Own Procedures.
- Reading: Defining Your Own Procedures.

Notes:

- Reminder: Osgood on Rosenfield convocation. Tomorrow. 11:00 a.m.

Overview:

- Why define your own procedures?
- How to define your own procedures.
- Lab.

User Defined Procedures

- It’s clear that programmers often want to (and need to) define their own procedures.
- User-defined procedures can add clarity to a program.
  - Rather than looking at how code does something, the user of a procedure can focus on what the code does.
- Programmers can avoid repetitive (and, therefore, error prone) code.
  - Rather than retyping the same code again and again, just changing a few values, a programmer can give a name to the same code.
- How do you define your own procedures? Using the following template:

  ```scheme
  (define your-procedure
   (lambda (param_1 ... param_n)
     expression-possibly-involving-parameters))
  ```

- For example,

  ```scheme
  (define square
   (lambda (val)
     (* val val)))
  ```

- You can (and should) document your procedures so that others can understand what they are supposed to do.
Lab on User-Defined Procedures

- Do the lab.
- Be prepared to reflect (e.g., to describe the most important or most confusing thing you dealt with today). (And no, you can’t say "Sam is the most confusing thing I dealt with today.")