The Little Schemer Simplified v2

Deepak Venkatesh

October 9, 2025

Contents

Note:

These are my personal notes created to deepen my understanding of Lisp and programming in general. ‘The Little Schemer’ (4th edition) by Daniel Friedman and Matthias Felleisen is a remarkable book that teaches programming concepts in a unique and playful way. It builds from first principles using only a small set of primitives, showing how powerful ideas—such as recursion, functional programming, lambda functions, and interpreters—can be expressed using just those few building blocks.
While the book uses Scheme, I prefer to work in Common Lisp and have adapted the examples accordingly. Despite its lighthearted tone, the book is far from an easy read—it demands close attention and careful thought.
All mistakes in these notes, whether typographical or conceptual, are entirely my own. Any misinterpretations are as well. I am still new to both Lisp and programming.

1 Foreword

By Gerald Sussman of MIT, co-author of the book SICP (the wizard book).

Key Takeaway: *In order to be creative one must first gain control of the medium.*

- Core skills are the first set of things required to master any pursuit.
- Deep understanding is required to visualize beforehand the program which will be written.
- Lisp provides freedom and flexibility (this is something which will only come in due course of time, as we keep learning more about programming).
- Lisp was initially conceived as a theoretical vehicle for recursion and symbolic algebra (this is the algebra we have been taught in school $(a + b)^2 = a^2 + b^2 + 2ab$).
- In Lisp procedures are first class. Procedures are essentially a varaint of functions. A mathematical function maps a given input to an output (domain - range/co-domain) but a procedure is a process to arrive at the result via computation.
- First Class basically means that procedure itself can be passed around as arguments to other procedures. Procedures can be return values. They can also be stored in data structures. A similar corollary (though not exact) is function of a function in pre-calculus.
- Lisp programs can manipulate representations of Lisp programs - this likely refers to macros and how in Lisp code can be treated as data.

2 Preface

Key Takeaway: *The goal of the book is to teach the reader to think recursively.*

- Programs take data, apply a process on that data, and then produce some data.
- Recursion is the act of defining an object or solving a problem in terms of itself.
-

3 Toys

4 Do It, Do It Again, and Again, and Again ...

5 Cons the Magnificient

6 Numbers Games

7 * Oh My Gawd *: It's Full of Stars

8 Shadows

9 Friends and Relations

10 Lambda the Ultimate

11 ... and Again, and Again, and Again, ...

12 What Is the Value of All of This?

13 Intermission

14 The Ten Commandments

15 The Five Rules