Ch 16

Recursive Datatype

- defined in terms of itself
  - Eg: immutable lists

Datatype Definitions

- ADT vs Recursive

Functions over Recursive Datatypes

- declared in the specification for the type, and implemented with one case per Concrete variant

Writing a Program

And let's broaden it further to a recipe for **Writing a program** (consisting of ADTs and procedures):

1. **Choose datatypes.** Decide which ones will be mutable and which immutable.
2. **Choose procedures.** Write your top-level procedure and break it down into smaller steps.
3. **Spec.** Spec out the ADTs and procedures. Keep the ADT operations simple and few at first. Only add complex operations as you need them.
4. **Test.** Write test cases for each unit (ADT or procedure).
5. **Implement simply first.** Choose simple, brute-force representations. The point here is to put pressure on the specs and the tests, and try to pull your whole program together as soon as possible. Make the whole program work correctly first. Skip the advanced features for now. Skip performance optimization. Skip corner cases. Keep a to-do list of what you have to revisit.
6. **Reimplement and iterate and optimize.** Now that it's all working, make it work better.