COMP 161 - Lecture Notes - 08 - How to Design Procedures, Round 2

Spring 2014

We then continue our exploration of data structures and procedure design by looking at compound and self-referencing data structures, including the C++ string type.

More Kinds of Data

1. Compound Structures
   Struct types. aka "And" data. Compound data is a singular data type constructed from more than one value. The racket *posn* was probably your first foray into compound data.

2. Self-Referencing Structures
   aka Recursive data. Lists are built up from smaller lists. This is the essence of self-reference. We also sometimes treat numbers in the same fashion: $5 = 1 + 4$, "five is 1 plus another number, 4".

3. Mutually-Referencing Structures Think a list of lists or lists of structs containing lists. Here we have types containing not just a self reference, but a reference to another kind of data which, in turn, refers back to the original data.

The *HtDP* templates capture some basic design guidelines for working with these kinds of structures.

Compound: Select each field, write helper procedures for the fields, and recombine the results of the fields, with another helper, as needed.

Self-Referencing: Use the itemization rule to determine if your dealing with a self-referencing variant. If you are, use the compound rule to separate the self-reference from the data. Recursively call the procedure on the self-reference, write helpers for the data, and recombine all helper results as needed.

Mutual-Reference: A combination of all other rules.

Compound Structures

Self-Referencing Structures

Mutually-Referencing Structures

Multi-Argument Procedures