Function Literals and Higher Order Functions

9-15-2010

Opening Discussion

- Let's look at solutions to the interclass problem.
- Minute Essay Comments:
  - What would be the purpose of having a function output a different type than the input type?
  - Is there a generally preferred way to do input?
  - Classes and methods vs. functions.
  - Functions and tuples.
  - Doing operations with more than one function: f(g(x))
  - High level math and computers.

Why Functions?

- Functions are used in programs for a number of reasons.
  - Reduce code duplication. You can call the same function multiple times and only write it once.
  - Improve readability and maintainability. Good function names make it easier to read. Small functions are easier to test and debug.
  - Break problems down/problem decomposition.

Problem Decomposition

- Never solve a hard problem. If a problem is hard, break it into smaller problems that are easier. Repeat until you are only solving trivial problems.
- Top-down
  - This is the “normal” approach where you start with the full problem and break it into pieces.
- Bottom-up
  - Sometimes you realize that different trivial pieces will be useful and build up from those.

Function Literals

- Just like 5 is a literal for an Int and “hi” is a literal for a string, you can write literals of functions.
- The full syntax is an argument list followed by an equals arrow followed by the function expression.
  - (a:Int,b:Int) => 3*a+2*b
- Types don't have to be specified in many situations, only if Scala can't figure it out.

Higher Order Functions

- This is the reason function literals exist. These are functions that take other functions as arguments or return other functions.
- The compose method is higher order.
- We could write our own compose function that is a higher order function.
- We will see a lot more higher order functions in chapter 7.

Minute Essay

- Write a function to evaluate a polynomial given the coefficients a, b, and c.
- Quiz #2 is next class.
- Interclass problem:
  - Write a Cartesian to polar coordinate converter. (Two arguments in and a tuple with two values out.)
  - Or
  - Write a function that takes a tax rate and a desired income and returns the required hourly wage.