

# Recursion and Repetition

9-23-2011

# Opening Discussion

- Minute essay comments:
  - How close is Scala to Java?
  - Math on tests relative to homework.
  - Making negatives of variables, `-b` should work. Space between `=` and `-`.

# Motivation

- We have the ability to do things once and to control whether or not certain things happen that once.
- Computers are really great for doing things multiple times.
- Reading a whole file or doing something until the user tells us to stop.

# Mathematical Recursion

- The idea of recursion comes from mathematics.
- A function is recursive if it is defined in terms of itself.
- All recursive functions will have at least two cases.
  - One where the function refers to itself.
  - A base case where it doesn't refer to itself.
- Let's look at some examples of this.

# Programmatic Recursion

- Now I want us to write some Scala functions that are recursive.
- They will look much like the math functions.
- We have to provide a return type.
- One argument changes to tell us when to stop.

# Scripts and Redirection

- One way to process data from a file is to write a script and use redirection.
  - `scala script.scala < input.txt`
- This way you don't have to enter the input over and over. Also handy if the input is really large.
- We'll learn other ways to deal with files later.

# Averaging Numbers

- I have some large files of numbers in the in-class code directory.
- I want to use recursion and redirection to find the average of these.
- If time allows let's think of some other things we might want to do with them.

# Minute Essay

- What questions do you have about this topic?
- We will do IcP #3 on Monday.