CSC161 2011S *Imperative Problem Solving*

Class 40: Sorting

Held: Friday, 15 April 2011

Summary: We explore the problem of sorting and how we think about it in C.

Related Pages:

- EBoard.

Notes:

- We may have prospies today.
- Reading for Monday: K&R 5.10-5.11.
- I will reserve time at the start of class for questions on on assignment 7.
- EC for today's CS Table [academic]
- EC for baseball (2 games Saturday, 2 games Sunday) [peer]
- EC for Bob's Benefit Bash Friday [peer]

Overview:

- Sorting: The Six P's.
- Sorting in C.
- Unit tests (starting).
- Switching parameter typtes.
- Generalizing.

Sorting

- The general idea: Put an array (or other collection) in order, typically from smallest to largest.
- Your first goal: Write the specification for something that sorts an array of integers.
  - `int-sort.h`

Testing Sorting

- Now, let's think about testing.
- I've written a UI that sorts integers from the command line.
  - `int-sort-argv.c`
- I've copied my merge sort from the past exam.
  - `int-merge-sort.c`
- Your second goal: Write a *simple* set of unit tests.
  - `int-sort-utest.c`
  - You'll be writing a more complex set in the homework.

Sorting Other Types

- What if we want to write code to sort arrays of floats?
- Too much code to link (and not yet available). Find it in `Examples/Sorting`.

Generalizing

- We've just coded by copy-paste-change. When you code that way, you should generally find a way to factor out the common code (that is, to generalize)>
- What can we generalize?
- How can we generalize?
- Too much code to link (and not yet available). Find it in `Examples/Sorting/General`.
