Course Announcements

- Stars was graded, you should have grade reports (just now).
- Autocorrect is due in 8 days.
  - Keep a steady pace
  - Trie is probably simpler than KdTree.
  - But the Javascript part is new and strange.
- No class next Tuesday,
  - Please don’t take off from now until next Wednesday.
Code Review: Stars

John Jannotti

CSCI032 (cs0320)

Feb 15, 2018
Everyone’s code is horrible

The only valid measurement of code quality: WTFs/minute
Except for today’s volunteer

If my code is garbage and I’m deluding myself into thinking I write ok code, please redact my name. – Sam Oliphant

- I included his name, so he’s in the clear.
- It passes tests. It’s written well.
- We’ll still find ways to improve it.
- Make it more flexible, more testable, maybe find some bugs.
- I had more volunteers than I can cover. I’ll put you at the front of the line for Autocorrect (if you still want it!)
Focus on what matters

- **Look for**
  - potential bugs.
  - inaccuracies (comments, specs).
  - inconsistency (internally, or with project)
  - unneeded reimplementations.
  - possible simplifications

- **Try to avoid “bikeshedding”**
  - Sometimes there are two ways to do things.
  - Try to focus on differences that matter.
  - and project consistency.

- **Care most about things that are hard to change later.**
  - You will have “users” who count on things not changing.
“Maslow’s code review”

http://blog.d3in.org/post/111338685456/maslows-pyramid-of-code-review
Some statistics on Stars

- 107 Handins
- Median: 84
- Average: 79
Nice things we looked for

- KDTree is generic
- KDTree implements Collection, extends AbstractCollection
- KDTree is n-dimensional
- Handle commands well (switch, data dispatch)
- Operates on squared distances
- Naïve implementation for testing
- Provide iterator() on KdTree, implement Iterable
- Leaf Nodes contain points that can be searched linearly
- Closes the file (+1 more for doing it right)
- Uses a Map to catalog star names for easy lookup
- Line numbers in error messages for bad star files
- Uses @Override for methods
- Uses assert/some other equivalent in their source code (not tests)
- Report multiple bad lines in one file
Points of emphasis

- KdTree: Collection (only), useful methods, useful “subabstractions”
- CSV: handling errors, potentially large input
- Repl: remove System.in/out, create abstraction for commands
- UI: Use what .ftl offers, bean/list handling, don’t abuse String