Data-Centric Introduction to Programming

Kathi Fisler

Brown University

What does an incoming data-science MS student need to know about CS?

decompose high-level questions into data queries

write queries as programs

clean data programmatically

how to test scripts

standard CS data structures (lists, hashtables, graphs) to prep for sorting and searching

a mainstream language

Course Structure

- Intro to expressions and tables (Pyret)
  - project: contrast student demographics and statewide test scores
  - data had some inconsistencies to resolve (detected via testing)
  - emphasis not on stats, but on CS practices around handling data

- Extract columns to lists, program over lists
  - Functions like map, filter, aggregate list data

testing throughout

- Tables as lists of structs

- Designing data organization (split into structs, tables, lists, etc)
  - hwk: to-do lists with tagged items; circulation and holdings for a library

Course Structure

- Structs for trees; lists and structs for graphs (still Pyret)
- What if want to update data contents? (migrate to Python)
  - example: to-do list manager, change due dates
  - mutating tuple contents; introduce loops
- How data is laid-out in memory (the heap)
  - combine with exercises on tracing program execution
- Hashtables
- User interaction, including Exceptions

Workshop Structure

- Intro to expressions and tables (Pyret)
- Defining functions
- Extract columns to lists, program over lists
- Tables as lists of structs
- Designing data organization (split into structs, tables, lists, etc)
- Structs for trees; lists and structs for graphs
- What if want to update data contents? (migrate to Python)
- How data is laid-out in memory (the heap)
- Hashtables
- User interaction, including Exceptions