New Course: Designing Programs for Data Science

Jennifer Dy, Tina Eliassi-Rad, David Kaeli, Rajmohan Rajaraman, Olga Vitek

Introduction to the fundamentals of programming for data sciences. The course encompasses the entire design cycle, from problem analysis to the development of test suites. The course will also introduce students to programming as a people discipline. Students will work in pairs, present code walks to panels, and learn to cope with an evolving code base. The course will rely on examples from programming languages that are commonly used in data science, such as Python, R, or Matlab.

Required topics include:

- Designing functions for finite data
- Designing functions for arbitrarily large data
- Abstracting with functions
- Designing generative recursive functions
- Designing functions with accumulators
- Design choices
- Designing communicating programs
- Designing classes
- Designing applicative classes
- Designing stateful classes
- Abstracting with classes
- Designing unit and system tests
- Working with files
- Regular expressions
- Introduction to parsing

Optional topics:

- Programming with vectors and matrices
- Data formats
- Data aggregation and group operations
- Time series
- IDEs and code version control

No prior programming experience is required.

Textbooks for the course include

- *How to Design Programs.* M. Flatt, M. Felleisen, R. B. Findler, and S. Krishnamurthi, MIT Press 2001
- *Learning Python.* Mark Lutz. O'Reilly Media, 2013.

Additional resources:

- *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython.* Wes McKinney, O'Reilly Media, 2013.

After completing this course, the student will be able to demonstrate the following competencies:

- Solve basic computational problems required in many areas of data science

- Represent information read from a file as data types
- Perform transformations on data to group and/or summarize, and compute simple statistics
- Write appropriate comments and test cases for maintaining software