Programming for Scientists: A Review

Luís Pedro Coelho

Programming for Scientists

April 14, 2009

University of Pittsburgh

Carnegie Mellon

Course Structure

- Module I: Introduction to Programming
- Module II: Software Carpentry
- Module III: Scientific Programming
- Module IV: Advanced Topics

Introduction to Programming

Topics

- Basic programming
- Object-oriented programming

Technologies

- Python

Software Carpentry

Topics

- Source control
- Software testing
- Defensive programming
- Debugging & Profiling

Technologies

- Subversion
- nosetests
- assertions
- Python debugger & profiler
- Shell

Scientific Programming: Numerical Representations

Topics

- How to represent integers
- Integers as bit strings
- How to represent floating point numbers
- Why you should never compare two floats for equality.

Scientific Programming: Array Programming

Technologies

- Numpy
- Scipy
- Matplotlib
- ...

Scientific Programming: Numerical Optimisation

Topics

- Functional minimisation
- Newton's Method
- Gradient Descent
- Random Hill-Climbing
- Local & global minima

Technologies

- OpenOpt
- scipy.optimize

Random Processes

Topics

- Pseudo random numbers
- Reproduceability
- Metropolis-Hastings

How do we get reproduceable random processes?

Open Source

Topics

- Licenses
- Distribution (a little bit)
- ...

File Parsing

Topics

- Regular expressions
- File encodings

Technologies

- Python `re` module
- Unicode & UTF-8
- Python pickle
- comma separated value files
- JSON
- INI files

Software Engineering

Topics

- Waterfall model
- Agile Development

Technologies

- Unified Modeling Language (UML)

Scientific Programming

1. Representation of numbers
2. Numerical minimisation
3. Open source
4. File parsing
5. Random processes

Advanced Topics I: Database Programming

Topics

- Relational Databases

Technologies

- SQL
- SQLite

Advanced Topics II: Graphical User Interfaces

Topics

- Event-based programming
- Signals & Slots
- Widgets as basis of interaction

Technologies

- Qt (PyQt)

Not Over Yet

- Next week, Django
- In two weeks, interfacing with other languages
- The project is due by the end of the semester.