Course Outline

ASTR288P

Sean Griffin

2017/09/01

UMCP / NASA GSFC

About Me

- Office: ATL 0251A
- Office hours: Friday 1PM-2PM (before class)
- Always confirm with be before coming for office hours!
- astro email: sgriffin
- I'm actually based at NASA Goddard, so I'm not around on other days.

Course format

- Homework:
  We have them, they will be assigned as the course progresses (”natural stopping points”).
  Generally due at the start of the following class.
  There will be time to work on them in-class.
- End-of-term project and presentation in place of a final written exam.

Resources

Books: Is that like the Internet but made out of a tree?

Online:

- Wikipedia
- `http://www.stackoverflow.com` Any Questions
- `http://www.codecademy.com` Python, Git, ...
- `http://tutorialspoint.com/cprogramming` C language
- `http://projecteuler.net` Challenging Problems
- `http://rosettacode.org/wiki/Averages/Arithmetic_mean`

What we'll cover (more advanced topics)

- UNIX:
  - Shell (we will use **bash** but others exist), shell scripting
  - File system (/, /usr/bin, $HOME, etc.)
  - Window managers, desktop environments
  - Editors (**emacs**, vi, gedit, pico, **sublime**, many others) – people have strong opinions but I do not!
  - Base commands (**cd, mkdir, ls, ssh**...)
  - Tools (**git**, gcc,)
- Scripting
  - Python
  - ipython
  - Jupyter
- Word processing with LaTeX:
  - Not included last time this course was taught but I think it's important to learn early on.
  - This document produced using LaTeX.

What we'll cover

- Some Object Oriented Programming (OOP)
- Compiled code
  - C/C++, Makefiles
  - FORTRAN is a thing that exists but we won't use it.
- Data analysis
  - Compiling and running analysis code
  - Analysis and plotting

What we probably wont cover

- Machine learning
- Multi-threaded / parallel programming (OpenMP, MPI, CUDA)
- ...

Hardware

Lab machines:

- Master: ursa.astro.umd.edu
- Nodes: lab001, lab002, ... lab013
- Printer: labs.astro.umd.edu

Virtual machines:

- **virtualbox**
- vmware

Your own machine:

- Linux (Ubuntu, Redhat, Fedora, Mint, debian, gentoo...) – You can dual-boot if you want.
- Mac OS X
  - You need Xquartz installed so certain features will work.
- Windows – Not recommended for this class
  - putty, Windows10 bash, VNC viewer
  - Probably easier just to run a virtual machine (this is what I do!).

Virtualbox

Virtualbox is a program that allows you to create and run virtual computeres within your host computer.

This is a handy alternative to dual-booting if you want to use both Linux and Windows (e.g. if you're a hardcore gamer but need *nix to do analysis for work).

You are effectively running an entire computer inside your OS, so performance can be an issue (particularly graphical stuff, although things are getting better).

We'll go through the process of creating a virtual machine and installing a Linux distribution on it. This will also fammiliarize you with the different aspects of a Linux installation.

Virtualbox

I personally use Linux Mint 17.2, but you can (and should!) try other distributions.

Lots of different desktop versions: Cinnamon, MATE, GNOME 3, Unity, KDE...