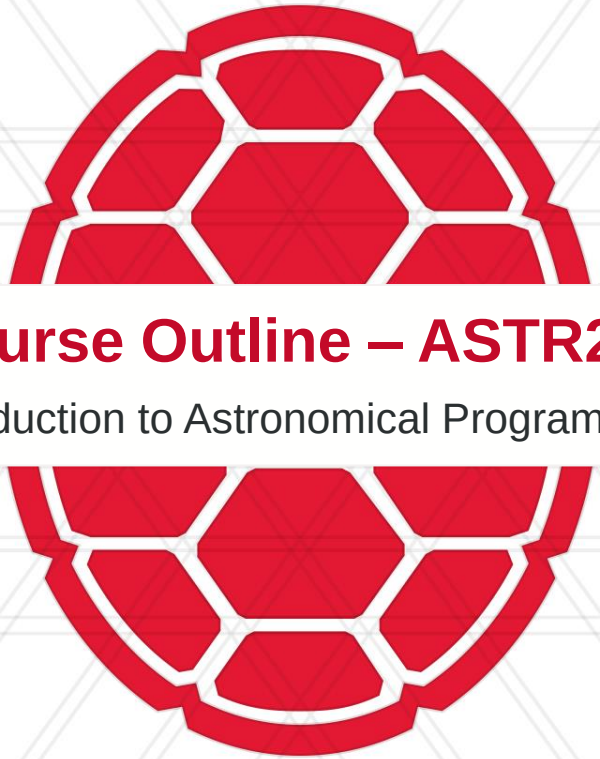


Course Outline – ASTR288

Introduction to Astronomical Programming

Kyle R. Murphy
2018-01-29
UMCP – NASA GSFC

UNIVERSITY OF MARYLAND

About Me

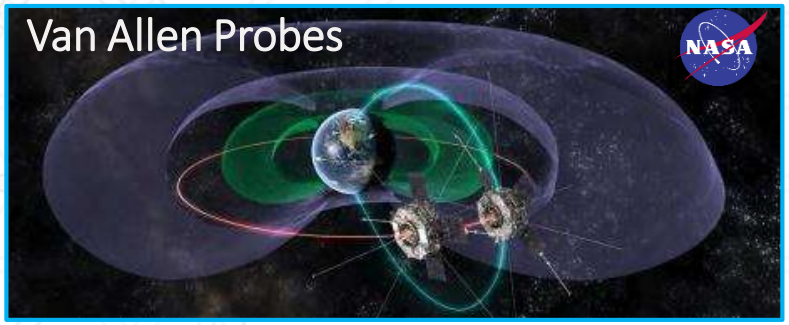
Kyle R. Murphy
krmurphy@umd.edu

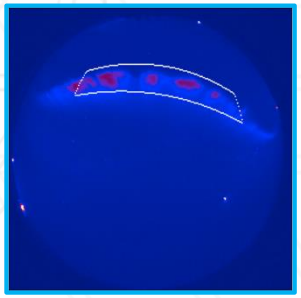
http://www.astro.umd.edu/~krmurphy1/ASTR288

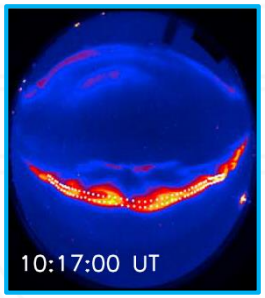
- Office: ATL 0251A
  - Based at NASA Goddard
- Office Hours: Monday 3:00 pm – 4:00 pm (before class)
  - Confirm office hours (often off campus).
  - Additional times can be set up by email if needed.

About Me

- Magnetospheric Physicist
  - Space weather
    - Solar Wind
    - Radiation Belts
    - Aurora







- Programming
  - C++ and Python
  - IDL (scientific analysis, similar to Matlab)

Course Format

Introduce students to Scientific and Astronomical Programming.

- Lectures will occur during most classes.
  - Class time will also be used to work through in-class problems/examples (learn by doing).
- Homework
  - Will be assigned throughout the course at natural stopping points.
  - 4-5 assignments generally due at the start of class.
  - End-of-term project and presentation in place of a written final exam.

Course Format

Learning Assessments	Category Weight
Participation:	10%
Assignments:	90%

Final Grade Cutoffs									
+	97.00%	+	87.00%	+	77.00%	+	67.00%		
A	94.00%	B	84.00%	C	74.00%	D	64.00%	F	<60.0%
-	90.00%	-	80.00%	-	70.00%	-	60.00%		

Participation:

- Attendance and in class exercises

Homework

- As the class progresses homework will become more difficult. As such there will be a cumulative marking scheme for homework, i.e., total points on all homework.

Resources

Won’t be using a textbook.

For coding the web is an excellent resource for all questions; debugging, installation, code description, manuals, etc.

- Google
- Wikipedia
- http://www.stackoverflow.com, Any Questions
- https://git-scm.com/doc, Git
- http://codecademy.com, Python, Git
- https://www.tutorialspoint.com, C, Python

What we'll cover

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, atom, visual studio, and many others), people have strong opinions but I do not!
- Base commands (cd, mkdir, ls, ssh...)
- Tools
  - Unix Tools (**ls**, **cp**, **mkdir** …)
  - Community Tools (**git**, gcc, g++, **python**, IDL, Matlab, R, …)

Scripting

- Python
- ipython
- **Jupyter**

Numerical Methods and Data Analysis

Open Source Software

- Final Project

What we won’t cover

- Parallel Programing (OpenMP, MPI, CUDA)
- Machine Learning
- Word processing

Schedule

			Topic
Week 1		Jan 21 – Jan 27	No Class
Week 2	Lecture 1	Jan 28 – Feb 3	Introduction and Assessments
Week 3	Lecture 2	Feb 4 – Feb 10	Unix (shells, files and directories), GIT
Week 4	Lecture 3	Feb 11 – Feb 17	More on Unix (file commands, scripting, permissions)
Week 5	Lecture 4	Feb 18 – Feb 24	Unix/Python (remote access, paths, variables, miniconda)
Week 6	Lecture 5	Feb 25 – Mar 3	Python (jupyter, variable types, control flow)
Week 7	Lecture 6	Mar 4 – Mar 10	Python (arrays, plotting) **Guest Lecture**
Week 8	Lecture 7	Mar 11 – Mar 17	Python scientific programming and numerical methods
Week 9	Lecture 8	Mar 18 – Mar 24	**Spring Break, no lecture**
Week 10	Lecture 9	Mar 25 – Mar 31	Python scientific programming and numerical methods
Week 11	Lecture 10	Apr 1 – Apr 7	Python scientific programming and numerical methods
Week 12	Lecture 11	Apr 8 – Apr 14	Python scientific programming and numerical methods
Week 13	Lecture 12	Apr 15 – Apr 21	Python scientific programming and numerical methods
Week 14	Lecture 13	Apr 22 – Apr 28	Python scientific programming and numerical methods
Week 15	Lecture 14	Apr 29 – May 5	Final Presentations
Week 16	Lecture 15	May 6 – May 12	Last class, recapping (continued presentations if required).

Topics we can cover:

- Root finding, curve fitting, numerical integration, numerical differentiation, time-series analysis, image processing, plotting, and multidimensional arrays.

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

Windows

- putty, **Windows10 bash (Ubuntu)**, VNC viewer

Activity - 1

Write on a piece of paper:

1. Your Name
2. Your year
3. Your Major
4. Do you plan to use your own computer? If so what operating system do you have?
5. Why are you taking this course?
   - fun, requirement, etc.
6. What grade are you hoping to get?
7. What did you like about previous classes? What didn't you like?
8. What are your expectations of this class?

UNIVERSITY OF
MARYLAND