Astro Data Science Seminar
Lab 1 - 2016 April 12
Tools of the Trade

Part 1
Installing things!

Python - install using Anaconda, works for Linux, Mac, PC
git - can be installed via GitHub app
Github - you'll need an account
atom - good all around text editor
slack - team communication

Part 2
How to download and submit materials for the seminar

The seminar will be a single open source repository, which lives on GitHub at:
https://github.com/jradavenport/WWU-seminar-2016

You will need for "Fork" this repository to your own GitHub account.

Then "Clone" *your* version on to whatever machine you're using. You'll have to configure your local version to point back at the original source:
https://help.github.com/articles/configuring-a-remote-for-a-fork/

Each week a new lab will appear with it's own directory, which will have example code and data files you'll need. You will have to sync your fork with my original version:
https://help.github.com/articles/syncing-a-fork/

Then do the assignment. Commit your changes to your local "repo", push to your fork. Then submit a pull request back to the original (my) version.
https://help.github.com/articles/using-pull-requests/

Make sure *your* work is in either it's own subdirectory, e.g.:
`WWU-seminar-2016/lab1/davenport/FILES`

Part 3

Look at the Notebook (.ipynb) file in the Lab 1 directory. If you view it on GitHub, you'll notice it is rendered nicely in the browser. Sweet! However, this is just a snapshot of the notebook when I synced it, and you cannot edit/run it unless you download it.

So, get the tools installed from Part 1, go through the steps in Part 2, and fire up the IPython Notebook. Usually on most machines you do this from the terminal (or shell) like this:

```
ipython notebook
```

But if you're using the Anaconda "Launcher", or similar, you might have an icon to click on.

The goal of this exercise is:
1) refresh your memory on how to use Python and IPython Notebooks
2) get comfortable with submitting assignments using GitHub
3) compute the radius that encloses half the mass of the Earth (total mass of the Earth is about 5.9e24 kg)

There are *many* ways to solve this computational problem using the data that I've provided. You could do a FOR loop (as I've started to outline), you could make tricky use of the numpy cumulative sum function...