Software to install on your computer before the REU program begins

- *Python*: Install the latest version of Anaconda Python version 3.x . Please verify that you can run Python version 3, and that you can open a Jupyter notebook with Python 3. One way to do this is to use the Navigator that is installed with Anaconda (the icon looks like a green ring), and then click on the Jupyter notebook box within the navigator. If you already have Python 3 installed and working, with Jupyter notebooks available, you can probably skip this step. (Though I strongly recommend Anaconda Python over whatever default comes installed on your machine; see below for one important reason.) If you are installing Anaconda on Windows, I recommend checking the option to add Anaconda to your Windows PATH – this will allow you to easily run Python through your Bash terminal just like on Mac or Linux.
- *Bash* : Be sure that you have access to bash command line. For Mac and Linux users, this is through your terminal application, which should already be installed. For Windows users, you will need to install your own application. A simple and light-weight option is to install git bash. For a more robust development environment, you may prefer to install Windows Subsystem for Linux (WSL). Note that if you install WSL, you should probably install Anaconda on the WSL side, rather than the Windows side. (I have used both git bash and WSL and prefer WSL, but if this is your first time using bash, you may simply want to start with git bash.)
- *Git* : Mac and Linux users should already have this installed. Windows users, git will be installed with the "git bash" or WSL link I included above. To check, open a (bash) terminal and type `git --version`. You should see the version number. If you don't, then you need to install git. For Mac users, if you do not have git installed or it is before version 2.17, please follow these instructions to install git via Homebrew, which is a program to manage software installations like this on a Mac. You may need to install homebrew first (instructions for that are provided). If you're using Linux (or WSL) and need to install git follow these instructions for your Linux distribution. After you've installed git, try `git --version` in the Terminal again to make sure it is installed.
- *GitHub*: Please create an account for yourself on GitHub (if you don't have one already).
- *Powerpoint* (or similar) : Optional, though you will probably want to use this to create your posters.
- *Browser* : I'm sure you all have a web browser installed on your computer. We will be using many browser-based applications (and creating our own websites), so you will need one. I recommend Google Chrome or Firefox.
- *Text editor:* You will need a text editor that is NOT Microsoft Word. I recommend using VS Code or Sublime Text (both for free), but you may have your own preferences. VS Code is a powerful IDE that can also integrate with git. Sublime is a very good text editor that is simple to use. I've used both and prefer VS Code.

Setting up a new Python environment in Anaconda

One of the major advantages to using Anaconda Python is that it allows you to create multiple Python "environments" on your computer. For instance, if you want to install some package but are worried it might corrupt your Python install, or if you usually use Python 3.10, but now you need Python 2.7, environments are the solution you've been looking for. Here's some documentation that may be useful, and below I will write the specific commands that you should type in your bash terminal to create your environment for this REU program. (If you can't get these to work before the summer, that's OK. I will help during our first Python workshop.)

- *To create your REU2023 Python environment (command should be all one line):*

```
conda create -n REU2023 -c conda-forge python=3.10 jupyter numpy scipy matplotlib pandas astropy astroquery bokeh emcee corner
```

- *To activate this environment (do this before running any Python commands):*

```
conda activate REU2023
```

- *To deactivate this environment (and return to your default Python):*

```
conda deactivate
```