Software to install on your computer before the REU program begins

- *Python*: If you already have Python 3 installed and working (and it is not the basic version of Python automatically included on your computer), and you can definitely use Jupyter notebooks (test by running `jupyter notebook` in a terminal; does a tab open with a coding interface somewhere?) then you can probably skip this step. Otherwise, we recommend installing MiniForge:
  - "MiniForge is a community led and supported environment, unlike Anaconda, which requires paid licensing for use… MiniForge is 100% free and the version being deployed contains many of the most common packages used." - From the High Computing Center of the DoD
  - Also, if you use a Mac that has an M1 processor, Anaconda famously has not played nicely with it. M1 users report much higher success with MiniForge for programming needs.
  - Basically, this ensures you will not encounter licensing issues, and you will absolutely still be able to do all the coding you need to do, as all packages that we will use are open-source.
  - To install:
    - Go to https://conda-forge.org/download/ and choose the appropriate installer
    - If on Unix/Mac, run `bash Miniforge3-$(uname)-$(uname -m).sh` in a terminal
    - If on Windows, simply execute (double-click, run) the installer you downloaded
  - If having troubles, check the MiniForge Github for more assistance
- *Bash*: Be sure that you have access to a 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 MiniForge on the WSL side, rather than the Windows side. (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 the WSL described 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.) 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).
- *Text editor:* You will need a text editor that is NOT Microsoft Word. I recommend using VS Code, Emacs, or Vim (all are free), but you may have your own preferences. VS Code can also integrate with git. Vim and Emacs are simple to use.

Setting up a new Python environment in MiniForge

One of the major advantages to using "Conda"-based Python (which includes MiniForge!) is that you can 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.

- *To create your CIERAREU Python environment (command should be all one line):*
  ```
  conda create -n CIERAREU -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 CIERAREU
  ```
- *To deactivate this environment (and return to your default Python):*
  ```
  conda deactivate
  ```