This is your introduction to the Unix environment. Several of these exercises are pointless and really do nothing. Their purpose is to familiarize yourself with some of the basic commands/file manipulations you will use throughout the course. Keep a cheat sheet for yourself that you can quickly refer to in the future. You can find these on google, but some of frequent commands you will use (ex. qlogin) are more specific to our system.

#---#

LOGGING ON TO quanah or hrothgar

#---#

There are various ways to log on to our HPCC. You can do this through the terminal (for Mac or *nix systems), a Unix emulator (CYGWIN) or you can use an SSH client. For simplicity we will use the BitVise SSH Client that has a bundled FTP client. For Mac people you can use your terminal and Filezilla.

1] Download and install the appropriate SSH/FTP combo
Windows: https://www.bitvise.com/

Mac: http://sourceforge.net/projects/filezilla/files/FileZilla_Client/3.14.1/FileZilla_3.14.1_macosx-x86.app.tar.bz2/download?nowrap

2] Set up and save your profile on the SSH client Host Name. There are two choices of clusters to use:

Quanah is newer, faster, and has more processors but it's also used more and has a limited wall time, which limits how long you can run any given job. It's also usually busier, limiting whether or not you can get processors to use.

Hrothgar is older, slower, and has fewer processors but I have my own queues that do not have wall time limits.

If you can, and there are processors available, choose Quanah. If not, go with Hrothgar and use any of my queues, Chewie, Yoda, and R2D2.

Specific instructions are available at:
http://www.depts.ttu.edu/hpcc/userguides/index.php#quanah

http://www.depts.ttu.edu/hpcc/userguides/general_guides/login_general.php

Use these instructions but substitute 'hrothgar' for 'quanah' if you're using Hrothgar.

Save your profile and connect.

The same information can be used to save a profile on your FTP client (Filezilla)

For Macs, in your terminal:

```
$ssh <eraider user name>@quanah.hpcc.ttu.edu
```

or

```
$ssh <eraider user name>@hrothgar.hpcc.ttu.edu
```

This guide will refer to quanah for the rest of the text but you can generally just substitute hrothgar for quanah if you need to.

#--#

MOVING FROM THE HEAD NODE

#--#

Now that we are logged on, you should notice that we are on the head node.

```
quanah:$
```

Do not perform any analyses on the head node. It is a crime punishable by death. It slows everyone down on the entire system. ONE of the ways to move off of the head node is to request an interactive session (via qlogin),

```
$qlogin –q omni –P quanah –pe sm 1
```

Notice the change in your command prompt. This tells you that you're are working from a compute node. The change will be something like this:

```
quanah:$  to   compute-20-10:$
```

If no nodes are available to use for qlogin, you will get a message saying your qlogin job failed. In this case, you can try doing a qlogin on Hrothgar, or a different Quanah queue.

```
$ssh hrothgar
$qlogin –q Chewie –P communitycluster –pe sm 1
```

You can also check the status of your job request via:

```
$qstat
```

#--#

MANIPULATING FILES AND DIRECTORIES

#--#

Now that you are logged on, there are several "basics" that we need to cover. These include - Directories - Paths (relative and absolute) - Commands * pwd * mkdir * cd * ls * cp * mv * rm (-r) - Tab completion There is a lot of information/utility in these

commands. We will discuss them here but it is your responsibility to take as many or as few notes as you need. For now, download the following file.

Often, you will need to create or transfer a file to your directory. The easiest way to transfer a files is through the GUI FTP. But there are plenty of other ways.

Read through this document (https://kb.iu.edu/d/afsk) and then:

1) Transfer the following files to a new folder in your home directory
   /lustre/work/daray/heliconiine/all_TEs_analysis/sine_tails/hMel_recent_metulj_100 _9_muscle.fa
   /lustre/scratch/daray/myoLuc/myoLuc2.fa
2) Display the first 10 lines of each file on your screen
3) Display the entire contents to your screen

If you want to make minor edits to your file without transferring each time you can use one of the many text editors that are available. Nano/Pico and VIM/EMACs are the most commonly used. VIM/EMACs are more powerful than Nano/Pico, but more complicated. For now, stick with nano.

Another way to transfer files, when they're coming from outside the system, is through wget. Be aware, you should use a qlogin for this. I don't think it will even work from the head node but it's rude to even try, especially with large files.

```
$wget "ftp://ftp.ncbi.nih.gov/genomes/Asparagus officinalis
/Assembled chromosomes/seq/4686 ref Aspof.V1 chr1.fa.gz"
```

#--#

WORKING WITHIN THE QUEUE

#--#

You can exit your qlogin session by closing your terminal. But that is a jerk thing to do. Your session will stay in the queue forever, locking out others from using that set of processors. You have to specifically tell the system you want to end your session by typing:

```
$exit
```

Now check the queue again,

```
$qstat
```

You should eventually see your job disappear. This brings the interactive session to an end, but what if you don't want an interactive session. You want to submit your job then go home. To do this, we use submission scripts.

Transfer the following script to your home folder.

```
$cd ~

$cp /home/daray/counter.sh
```

Then run the script using:

```
$qsub counter.sh
```

and peek at the queue with

```
$qstat
```

Notice the differences. This job is either running or queued to run, but we aren't doing anything.

Open the file in a text editor,

```
$nano counter.sh
```

and you will see why. Check what files were output, also using nano.

Change your submission script to count to 1000 instead of 100, resubmit, then once it is running look in your home directory. What new files exist?

Now that this program has been altered, it will run for a very long time. Anytime you want to kill a job just type

```
$qdel <the job number>
```

The job number can be found using qstat.

#--#

INSTALLING A LOCAL COPY OF PYTHON

#--#

Python is a useful programming language that I use a lot and will take advantage of in this course. You will need to have python installed to accomplish some of tasks. You may already have python3 installed if you've used HPCC previously but if not, we'll take care of that here. Instructions are derived from http://www.depts.ttu.edu/hpcc/userguides/application_guides/python.local_installation.php. You should read this guide.

1] log in to Quanah and remain in your home directory but grab a node with qlogin

```
$qlogin -q omni -P quanah -pe sm 1
```

2] install miniconda (a version of python with minimal but expandable features)

```
$/lustre/work/examples/InstallPython.sh
```

3] get python into your working environment

```
$. /home/<eraider>/conda/etc/profile.d/conda.sh

$conda activate
```

#---#

CONNECTING FROM HOME OR ELSEWHERE

#---#

Some of the work for this class will be conducted outside of class. You may be off campus when you do it. On campus, you can connect to HPCC directly. Off campus, you need to connect to a secure VPN first. Instructions to do so are available at: https://www.askit.ttu.edu/portal/app/portlets/results/viewsolution.jsp?guest=1&solutionid=140702103827226.

Make sure you do both steps, 1. Download and install the software. 2. Connect to the VPN server when you want to log in to HPCC.

#---#

THE FEW LAST THINGS

#---#

It is helpful to understand how about your .bashrc and path….

You path is the address of where a file is. You've used it extensively already. The text in red below is the path to the file 'hMel_recent_metulj_100_9_muscle.fa'

/lustre/work/daray/heliconiine/all_TEs_analysis/sine_tails/hMel_recent_metulj_100_9_muscle.fa

There are a lot of parameters that can be adjusted surrounding your terminal session. These can be manipulated via the .bashrc file. Each time you log on the .bashrc file is read…so if you want to store a program name and always have it available, then you could include this information in your .bashrc. **BE CAREFUL** when editing your .bashrc file, and **DO NOT** delete the existing lines – they are there for a reason, and deleting them could cause you to be unable to interact with the command terminal properly.

Whenever you are running a program you can use to bring it up if it is in your path. This is a list of directories that the system will search to find programs to run. For example, if you want to run MUSCLE but it's in a different directory you can either include the path to MUSCLE or you can add the directory to your $PATH variable…and have it

automatically found. Either way works and both have their advantages. For now, don't worry about this, but be aware of it.

Other times, you will need to load pre-built modules using 'module load'. For example, we may use samtools. Rather than installing for yourself, you can just load it into your environment by typing

```
$module load samtools.
```

Some modules are oddly named and it may take some effort to find them. For that purpose, use

```
$module spider <name of module you want>
```

This will search the available modules and let you know how to load it if it's available.

For more information about loading modules, look here: http://www.depts.ttu.edu/hpcc/userguides/general_guides/software_environment.php.

More details and examples of basic maneuvering and use of Quanah can be found in the HPCC user guides: http://www.depts.ttu.edu/hpcc/userguides/index.php.