BE 434/535 Biosystems Analytics

The field of "bioinformatics" is biology plus data science. This course assumes you have some understanding of the Unix command line and some programming experience. At the conclusion of this course, you should be able to:

- Write, test, and document programs in bash and Python
- Use the source code management system Git to version, share, and distribute code
- Use parallelization techniques and hardware (HPC) to run programs faster
- Package and distribute software to create reproducible workflows

Ocelote (UA HPC)

Given that our class will have students on a variety of operating systems (Windows, OSX, Linux), we will use the HPC (high performance computing) cluster at the University of Arizona for our work.

To gain access to Ocelote, students must:

1) Have a terminal application ("Terminal" or "iTerm2" on OSX, Gitbash on Windows)
2) Sign up for UA's NetID+ (https://netid-plus.arizona.edu/)
3) Be sponsored (https://portal.hpc.arizona.edu/portal/)

Once you have these, open a terminal and type:

```
ssh <NetID>@hpc.arizona.edu
```

You will be prompted for additional authentication for NetID+. If all goes well, you should see something like this:

```
Last login: Sat Jan  5 07:53:30 2019 from ip72-200-123-88.tc.ph.cox.net
This is a bastion host used to access the rest of the environment.

Shortcut commands to access each resource
-----------------------------------------
Ocelote:                El Gato:
$ ocelote               $ elgato
```

Or you may see this (e.g., if you have enabled the menu with the `menuon` command):

```
===============
HPC.ARIZONA.EDU
===============
Please select a target system to connect to:
```

```
(1) Ocelote
(2) El Gato
(Q) Quit
(D) Disable menu
```

Either way, proceed to log in to Ocelote for your work. If you are uncertain which machine you are on, use `hostname`. If you are on the bastion host, it will be "gatekeeper.hpc.arizona.edu." Once you are on Ocelote, the hostname should be something like "login1" or "login2."

SSH Keys

If you would like to avoid the 2-factor authentication, then copy your SSH private key to the target system like so:

1) On your local machine and `cd ~/.ssh`. If you do not have one, then run `ssh-keygen`.
2) Copy the contents of the `id_rsa.pub` file (the "public" part of your key). If you do not have one, then run `ssh-keygen`.
3) Login to the target system and `cd ~/.ssh`. If you do not have one, then run `ssh-keygen`.
4) Edit the `authorized_keys` file (e.g., with `nano`) and paste in the public key. Save and exit your editor.
5) Set the permissions with `chmod 600 authorized_keys`
6) Test your login from your local machine.

Git(hub)

You will use the `git` source code management program to gain access to the course materials as well as turn in your assignments. Once you are on the Ocelote (or a similar Unix platform), it's quite likely that Git is already installed. Check the version like so:

```
[hpc:login3@~]$ git --version
git version 2.2.2
```

Github.com is a commercial company that hosts Git repositories. It is free to create accounts and host small, public repositories. You will need to create an account and then share your username with us. I suggest you add your public SSH key (see "SSH Keys" above) into your Github "Settings/SSH and GPG Keys" so that you can more easily push and pull into your repositories. You'll need to add a key from each machine you intend to work from, i.e., both your laptop and Ocelote.

Once you have an account, visit the course repo in a web browser and "fork" our repository into your own account by clicking the "Fork" button in the upper-right corner. This will create a copy of our repository into your own account.

```
https://github.com/hurwitzlab/biosys-analytics
```

Now you can copy the course repo to your machine(s) like so:

```
[hpc:login3@~]$ git clone git@github.com:yourusername/biosys-analytics.git
Cloning into 'biosys-analytics'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 16 (delta 1), reused 12 (delta 0), pack-reused 0
Receiving objects: 100% (16/16), 104.71 KiB | 0 bytes/s, done.
Resolving deltas: 100% (1/1), done.
Checking connectivity... done.
```

To stay together, you will add the class Github repo as an "upstream" repo:

```
git remote add upstream https://github.com/hurwitzlab/biosys-analytics.git
```

When you need to get new content, "pull" from the upstream repo's "master" branch:

```
git pull upstream master
```

About The Author

> "Computer programming has always been a self-taught, maverick occupation." - Ellen Ullman

My undergraduate degree was in English and music. I learned to program on my own and on the job starting in 1995, so I started relatively late and wasn't formally trained in programming until much later in my MS program. I say this so you know that everyone starts somewhere, some later than others, but it's like the joke "When is the best time to plant a tree? Thirty years ago. When is the second best time? Now." Now is a great time to become a programmer. I look forward to teaching you what I've learned in course of 20+ years of programming in industry and bioinformatics.