Deliverable 2A: Working with sequencing reads **All Unit 2 work due by Mar 10**

☑ **PREP**: Set up a directory for Unit 2 work, adding any subdirectories you want.

deliv_2A_1of3 and deliv_2A_2of3 terminal sessions

☑ Download and uncompress the files for Week 4, as instructed in section 5.1 of the Marine Genomics tutorial, making sure they are in the appropriate directory.

☑ Check to see which required modules are available on Discovery:

	available?	version
samtools:	yes	1.10, 1.18, 1.9
bowtie2:	yes	1.3.0, 2.3.5.1, 2.5.2 ← use version 2.3.5.1
cutadapt:	no	n/a
fastqc:	yes	0.11.8, 0.11.9

☑ Unzip each file ending in .fastq.gz extension.

☑ Use 'head' to examine the contents. What are these files from?
The files are genome reads. Each file contains nucleotide sequences and their corresponding quality strings.

☑ Determine how many sequences are in each of the six .fastq files.
1. 1000
2. 1000
3. 1000
4. 1000
5. 1000
6. 1000

See Reference #1 for website consulted.

☑ Unzip the file ending in .fna.gz. Use `head` to check out the contents. What sort of sequences does this file contain?
This file contains sea cucumber scaffolds and contigs from whole genome shotgun sequences.

☑ Determine how many sequences are in this file.
2000

☑ **QUALITY CONTROL**: Now perform quality control on the reads according to the instructions in section 5.2.

To display the resulting .html file, log onto the OOD for Discovery and navigate to the directory that contains the .html file.

Open Finder (if on a Mac) and drag the file from the Discovery OOD to your Desktop folder. In Finder, right-click and open with chrome (might also work with other browsers). The result should be some graphically pleasing output.

deliv_2A_3of3_MarineGenomics_20240309_short.txt terminal session

TRIMMING: Remember that next-gen sequencing involves attachment of adapters to the fragments before sequencing. The adapters have sequences that serve as primer-binding sites as well as sequences that allow them to base-pair with oligos on the flow-cell, and short sequences that serve as unique identifiers of different samples.

A small stretch of nucleotides from the adapter are found at the start of each read, and these need to be "trimmed" or cut out since they are not actually part of the read.

☑ Use `head` to examine several reads from a file and see if you can identify the adapter segment that needs trimming: ___TGCAG_____

Refer to section 5.3 for instructions on how to trim the adapter sequence from the reads. This task requires a package called "cutadapt". I have created an environment and path for all the files involved in this program in our shared folder:

```
/courses/BIOL3411.202430/shared/cutadapt_env
```

☑ To use the package, first request access to a computing node:

```
srun --pty /bin/bash
```

☑ Load the anaconda module:

```
module load anaconda3/2022.05
```

☑ Activate the cutadapt environment:

```
source activate /courses/BIOL3411.202430/shared/cutadapt_env
```

☑ Now try out the first command in section 5.3, following the example in the second command. However, place ./ in front of cutadapt since you will be running this command from a different location.

☑ **SHELL SCRIPTS**: Simple scripts (small pieces of code) can be created in nano and run on the shell. Work through sections 3.25 - 3.27, then stare at the creature in 3.28.

☑ Return to section 5.3 and create and run script for trimming the reads of all the files.

INDEXING A GENOME: Each read will need to be aligned to the genome, which involves searching for the highest-scoring alignment. "Indexing" the genome involves breaking it up into a more searchable format.

☑ Be sure you're still on a computing node (look at the info just before the command line cursor), and that bowtie/2.3.5.1 is loaded. Then follow the instructions in section 5.4, checking that the set of indexed files is now in your directory.

☑ What did the second argument in the bowtie-build command do?
The second argument in the bowtie-build command indicates the portion of the filename for each of the index files before their respective file extensions.

☑ What do you see if you use `head` to look at the contents of one of the new files?
I see random characters that do not make sense. This likely indicates that the index files are all compressed.

MAPPING READS TO GENOME: Now that the genome has been made into a more searchable format (indexed), you can ask the program to find the place in the genome where each read aligns. To do this, you'll write another shell script.

☑ Use the code shown in section 5.5.

☑ What else do you need to do to execute this script?
You need to add the code in section 5.5 to a shell script and then change the permissions of the script: `chmod 760 map-reads-script.sh` (alternatively, could use `chmod +x map-reads-script.sh`).
Then, run the script using the command: `./map-reads-script.sh`

☑ Use `head` to look at the contents of one of the new files. What do you see?
There is one main header and then @SQ entries that store info about the reference sequence. Each @SQ entry has the sequence name and the sequence length [2].

CONVERT SAM TO BAM FILES: Bam files are a compressed version of the sam files. We need the compressed versions as input for programs that will analyze the reads in terms of variants or read counts, for example.

☑ Create another bash script containing the code given in section 5.6.

ESTIMATE GENOTYPES: known as genotype "calling". Follow the instructions in section 5.7, except that the code you run will look like this:

First you'll need to activate the environment I set up for the course:

```
source activate /courses/BIOL3411.202430/shared/angsd_env
```

☑ Then you'll run this command rather than the one on the tutorial site:

```
/courses/BIOL3411.202430/shared/angsd_env/angsd/angsd -bam
bam.filelist -GL 1 -out genotype_likelihoods -doMaf 2 -SNP_pval 1e-2 -
doMajorMinor 1
```

Pro-tip extra:
Install tree (on your machine; it's already available on cluster)

All Unit 2 work is due by Mar 10.
Each deliverable is complete when you have:

- answered each question
- saved a terminal session or screenshots demonstrating your performance of the commands (on Discovery or posted on GitHub)
- indicated in a "TOC" (table of contents) file where your work is found (GitHub repo or specific path/name_of_file on the cluster).

References

1. Second method to count # sequences in a fastq file- count number of lines and divide by 4: https://biohpc.cornell.edu/doc/RNA-Seq-2019-exercise1.pdf
2. Format of SAM file contents: https://bioboot.github.io/bimm143_F21/class-material/sam_format/#:~:text=The%20required%20key%2Dvalues%20are,(e.g.%2C%2015%2C072%2C434%20bases).