**Eric\_QC for GTS**

**- Requirement**

A) Eric\_QC.py

B) Eric-QC\_metadata.csv

C) Active NIH-HPC (biowulf) account

**- Concept and method of implementation**

I used SWARM system to run several QC metrics, such as fastQC and Kraken based on Eric QC recipe, over each fastq file within a single directory in parallel mode. Python script will parse the input directory and create a hash table based on the metadata CSV file's parameters; Then it will generate a swarm file based on that. This python script will also create a shell script separately for the multiQC step. Since multiQC should execute at the last step, the shell script used --dependency method in SBATCH system- binds the execution of multiQC script to the swarm script completion. All these functions are managed by a general bash script, which will control this structure. This bash script will be the third output of the python script.

So, to be specific the python script will generate the following output based on metadata file:

A) QC swarm file

B) multiQC script

C) execution bash script

**Step-by-Step tutorial**

**- Step1: updating metadata file**

\*. open provided metadata file with Microsoft Excel, and edit the following fields

1) title

give your project a title; this title would be the top directory that locates all outputs.

2) single\_end

This is a boolean value which specifies your fastq files layout to be “paired-end” or “single-end”; 1=single-end, 0=paired-end

3) input

input is the directory path which all your fastq files are located there; the script only scans the first level depth of the given path, so subdirectories will not be included

4) output

Output is the path which will be used as workspace and result path

\*\*.save your metadata CSV file path after editing

**- Step2: load python**

you need python3 to run this script; please load python3 with the following command:

**module** load python

**- Step3: run python script**

please locate the python script called Eric\_QC.py, and your updated metadata file,

the execution of this script is very straight forward:

python Eric\_QC.py ./Eric-QC\_metadata.csv

.\*\*\* The location of your metadata file will be used to generate the required script.

**- Step4: verify generated scripts**

as you check the location of your CSV metadata file, three files have been generated

A)Eric-QC\_metadata\_NGS\_test\_script.swarm

B)Eric-QC\_metadata\_NGS\_test\_multiqc.sh

C)Eric-QC\_metadata\_NGS\_test\_execution.sh

**- Step5: submit to NIH-HPC**

submit the execution script to Biowulf with the following command:

bash Eric-QC\_metadata\_NGS\_test\_execution.sh

**- Step6: check the result**

when the execution is completed, you can open the generated multiQC HTML file with your browser and check the result. It is located on your specified output

A)NGS\_test\_multiqc\_report.html