

In-Person

*Foundations in
Genomic Analyses*

Filetypes and Quality Control

Northwestern | INFORMATION TECHNOLOGY
RESEARCH COMPUTING AND DATA SERVICES





COMPUTING AND SOFTWARE

Access to high performance computing, research software, and global networks for conducting computationally intense research.

DATA MANAGEMENT AND SHARING

Learn about data management planning and options for storing, securing, transferring, and sharing data.

DATA SCIENCE, STATISTICS, AND VISUALIZATION

Support for collecting, analyzing, visualizing, and programming with research data.

TRAINING AND CONSULTATION

Identify events, resources, and people to help you learn computational and data skills for your research.

RESEARCH COMPUTING AND DATA SERVICES

We're here to help after the workshop!

quest-help@northwestern.edu

bit.ly/rcdsconsult

<https://sites.northwestern.edu/researchcomputing/>

What do data look like when they come of a sequencer?

How do you interact with these filetypes?

How do you know the quality of your sequence data?



What do data look like when they come of a sequencer?

How do you interact with these filetypes?

How do you know the quality of your sequence data?



How do I organize files on Quest?

How do I look at files from the
command line?

How do I use Quest's software?

SETUP...

1. log onto Quest

```
ssh <netid>@quest.northwestern.edu # enter your netid password
```

2. move to our classroom folder

```
cd /projects/e32680/03_sequencingfiles
```

WHAT'S IN THIS FOLDER?

Take note of the different file types and naming conventions.

```
cd ./files
```

```
ls
```

FASTA FILES

- a text-based format for representing either nucleotide sequences or peptide sequences with single-letter codes
- header line starts with >
- followed by lines of sequence data

```
>NZ_CP084109.1 Micrococcus luteus strain anQ-m1 chromosome, complete genome
ATGCTCGCGTCCACTATGCAGTTCTCAAACAACAAACCCACACCCCCATCCACCAACCCTCAA
GGCCAGCTTCACAAGGTAC
GGACACCAGAAACAACAACCCCAACCCTCACAGATCAGGGTGCTGCTGTTCCAGGAC
CCAACAGTGTGCCAACACCCAGC
TCGCAGCCACCAACACCACCCATTCCAACCACACCCTCACAGGCATGACGTACTCGAGCA
CTGCCGGCACCAACCCACTGGC
ACCATTCGTTGATGTTCCACCCATGAGCAACCCGCCATCACACACGCGGTGATGCAACG
GGCACTAATGCTCCTTAGAAA
```

FASTA FILES

- format originated from FASTA software (similarity search tool, like BLAST)
- ‘FASTA’ stands for Fast-All vs Fast-N and Fast-P
- some make distinction between single and multi-fasta files
- multi-fasta files have multiple sequences, each preceded by a header line
- you can concatenate fasta files together to achieve this

FASTQ FILES

- text-based format for sequence data with single letter codes
- defacto file type for NGS sequencing reads data
- contains sequence data + quality information
- header line starts with @ followed by sequence data, and then quality data

Line	Description
1	Always begins with '@' and then information about the read
2	The actual DNA sequence
3	Always begins with a '+' and sometimes the same info in line 1
4	Has a string of characters which represent the quality scores; must have same number of characters as line 2

FASTQ FILES

@HWI-ST330:304:H045HADXX:1:1101:1111:61397

CACTTGTAAGGGCAGGCCCTTCACCCCTCCGCTCCTGGGGGANNNNNNNNNNNC
GAGGCCCTGGGGTAGAGGGNNNNNNNNNNNGATCTTGG

+

@?@DDDDDDHHH?GH:?FCBGGB@C?DBEGIIIAEF;FCGGI#####
#####

Quality encoding: !"#\$%&'()*+,./0123456789:;<=>?@ABCDEFGHI

Quality score:

0

10

20

30

40

FASTA.GZ OR FASTQ.GZ

- gz is a compressed file format
- gzip to zip files to gz
- gunzip to unzip file from gz
- gzip is a GNU utility, and installed at a system level on Quest
- BUT most software handles zipped files without the need to gunzip, which save you lots of space

READING THESE FILES

- because they are text-based, you can use linux commands for text files to interact with them

`cat` prints entire file contents to the terminal

`head` prints first few lines of a file to the terminal

`less` prints portions of file contents that you can scroll through

- these files can be very large!
- so, avoid holding them all in memory, or printing the entire contents to your terminal



LET'S TRY IT OUT! USE THE COMMANDS BELOW TO INVESTIGATE THE EXAMPLE FILES AND ANSWER THE FOLLOWING QUESTIONS.

- Which files are human-readable?
- Are any small enough to use cat with?
 - Use ls -lh to check file size

`cat` prints entire file contents to the terminal

`head` prints first few lines of a file to the terminal

`less` prints portions of file contents that you can scroll through



USING CAT TO COMBINE FILES

Because these file types are a series of lines with distinct headers, you can cat to put them together into one file.

- combining different lanes of sequencing for the same sample
- similarity searching
- this is different from *merging* reads

USING CAT, GREP, AND WC

Because every header starts with > or @, you can search for and then count the occurrence of these characters to know how many sequences you have.

```
cat GCF_020217385.1_ASM2021738v1_genomic.fasta | grep ">" | wc -l
```

QUALITY CONTROL...

- starts with understanding the quality of the reads
- the raw contents of these types of files aren't particularly informative
- you probably want to call some sorts of ~stats~ on them

FASTQC

- GUI-based or command line software for assessing quality of sequence data
- available on Quest's module system
- recommended to either run batch job or use the GUI through **Quest OnDemand**

FASTQC ON QUEST ONDEMAND

1. navigate to qondemand.ci.northwestern.edu in a browser
2. choose “GNOME Desktop” from the interactive apps dropdown
3. fill out the job script

short partition

account e32680

1 hour

5 GB of memory

FASTQC ON QUEST ONDEMAND

1. click “Activites” in the top left hand corner
2. open a terminal from the left side dock
3. module load fastqc/0.12.0
4. fastqc

FASTQC ON QUEST ONDEMAND

1. click “File” -> “Open”
2. navigate to our allocation folder
(/projects/e32680/03_sequencingfiles/files)
3. Load Old_C1_100k.fastq

FASTQC ON QUEST ONDEMAND

1. click “File” -> “Save Report”
2. navigate to your folder within the allocation folder
3. choose either html or all as the filetype to save

GOOD FASTQ EXAMPLE

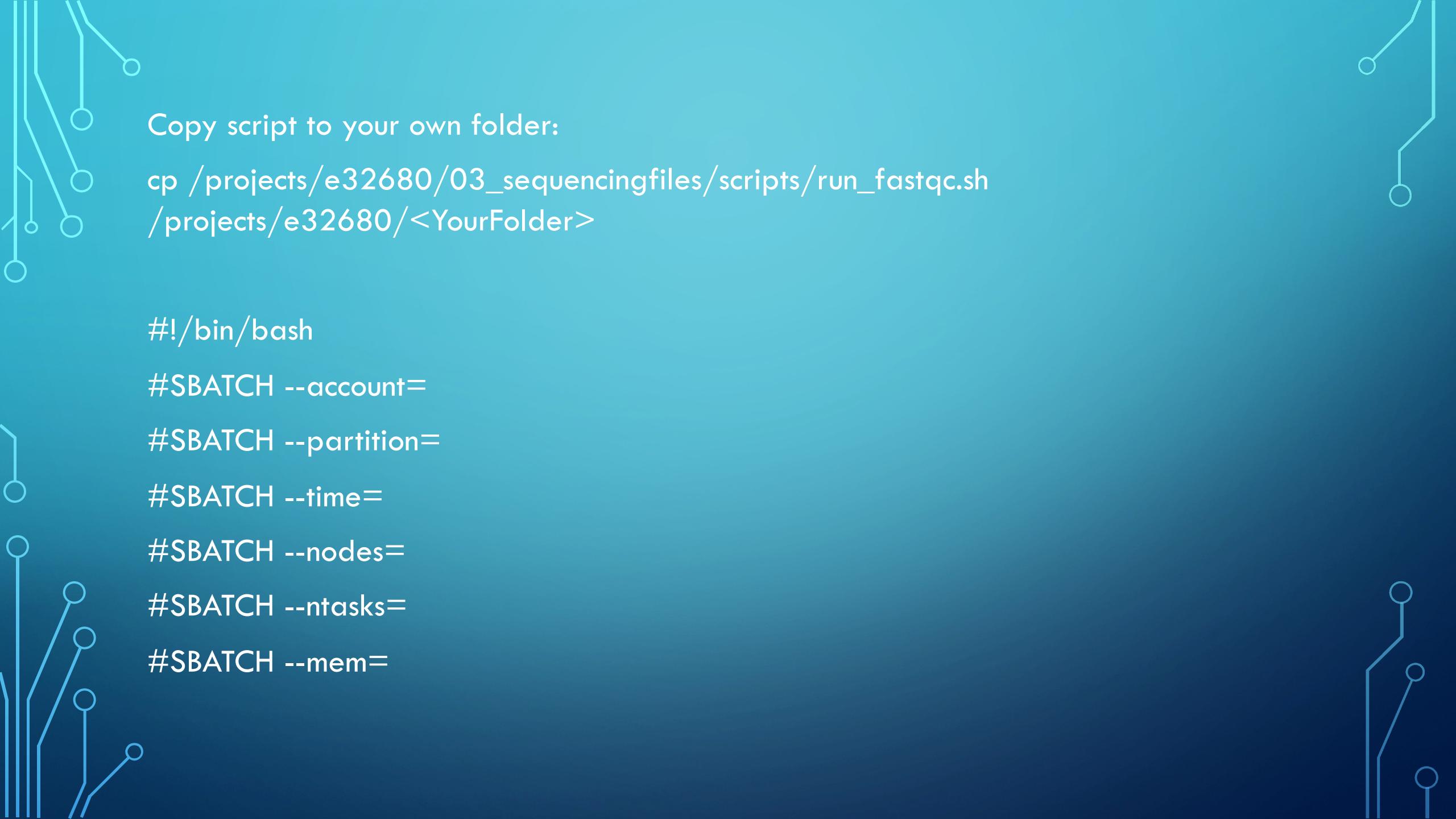
- https://www.bioinformatics.babraham.ac.uk/projects/fastqc/good_sequence_short_fastqc.html#M5

FASTQC COMMAND LINE

- GUI-based or command line software for accessing quality of sequence data
- available on Quest's module system
- recommended to either run **batch job** or use the GUI through Quest OnDemand

RUNNING A BATCH JOB

- requires a job script
 - sbatch parameter to request resources from SLURM
 - load modules needed to set up software environment
 - command to run fastqc
- launch with sbatch jobsript.sh
- check on it with slurm commands: squeue, sacct, seff



Copy script to your own folder:

```
cp /projects/e32680/03_sequencingfiles/scripts/run_fastqc.sh  
/projects/e32680/<YourFolder>
```

```
#!/bin/bash  
  
#SBATCH --account=  
#SBATCH --partition=  
#SBATCH --time=  
#SBATCH --nodes=  
#SBATCH --ntasks=  
#SBATCH --mem=
```

```
#!/bin/bash

#SBATCH --account=e32680

#SBATCH --partition=short

#SBATCH --time=00:40:00

#SBATCH --nodes=1

#SBATCH --ntasks=1 #scale up if running multiple samples

#SBATCH --mem=5G
```

```
#!/bin/bash

#SBATCH --account=e32680
#SBATCH --partition=short
#SBATCH --time=00:40:00
#SBATCH --nodes=1
#SBATCH --ntasks=1 #scale up if running multiple samples
#SBATCH --mem=5G

module purge
module load fastqc/0.12.0
```

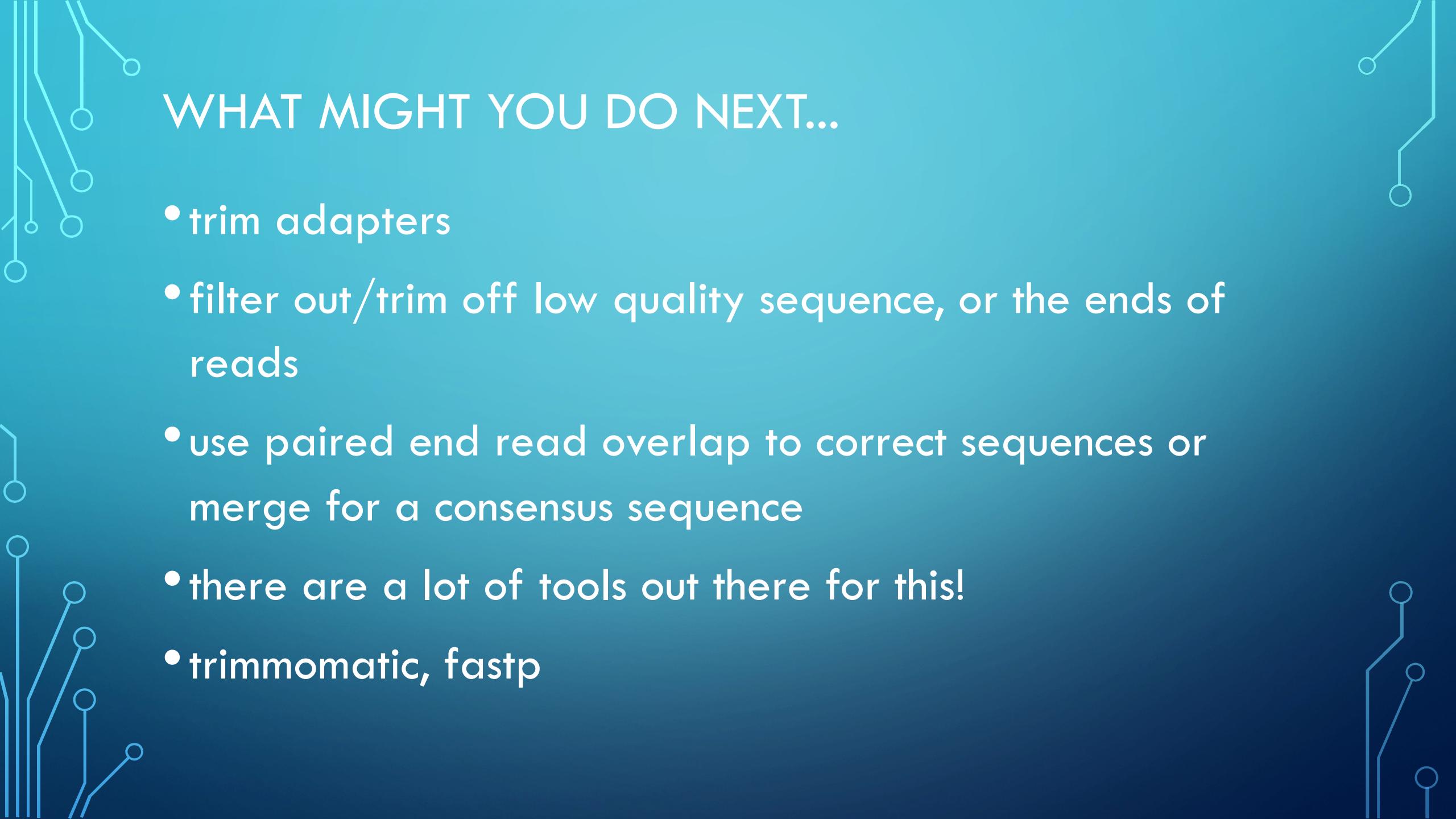
```
#!/bin/bash
#SBATCH --account=e32680
#SBATCH --partition=short
#SBATCH --time=00:40:00
#SBATCH --nodes=1
#SBATCH --ntasks=1 #scale up if running multiple samples
#SBATCH --mem=5G
module purge
module load fastqc/0.12.0
fastqc -t 1 --extract /projects/e32680/03_sequencingfiles/files/Old_C1_100K.fastq
```

THINGS TO NOTE...

- File paths should be relative to the directory you launch the script in, or absolute
- You can use the wildcard * to match all fastq files in a folder, or any naming pattern
- -t indicates the number of threads/CPUs to use, this should be the same as the --ntasks SBATCH parameter, you can use the \$SLURM_NPROCS variable
- --extract gives you the zipped folder as well as the html file

MORE ABOUT FASTQC

- <https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>
- <https://github.com/s-andrews/FastQC>
- https://mugenomicscore.missouri.edu/PDF/FastQC_Manual.pdf



WHAT MIGHT YOU DO NEXT...

- trim adapters
- filter out/trim off low quality sequence, or the ends of reads
- use paired end read overlap to correct sequences or merge for a consensus sequence
- there are a lot of tools out there for this!
- `trimmomatic`, `fastp`

OTHER FILETYPES

.UBAM OR .BAM

- .bam files are generally mapped - binary alignment map
- binary version of SAM files - sequence alignment maps
- BUT some sequencing centers also send unmapped bam files
- GATK software takes and sometimes prefers unmapped bam files
- Like the other file formats, they have header lines followed by sequence information. They also include information about the quality of the alignment.

.VCF FILES AND .GTF FILES

- tab delimited file txt files
- vcf - variant calling format
- gtf - general transfer format (same as gff2)
- gff - general feature format (now at gff3)

EXAMPLE GFF FILE

gencode.v41.annotation.gtf

Annotation file from GENCODE

##description: evidence-based annotation of the human genome (GRCh38), version 41 (Ensembl 107)
##provider: GENCODE
##contact: gencode-help@ebi.ac.uk
##format: gtf
##date: 2022-05-12

```
chr14 HAVANA gene 100725705 100738224 . + . gene_id "ENSG00000185559.16"; gene_type "protein_coding"; gene_name "DLK1"; level 2; hgnc_id "HGNC:2907"; havana_gene "OTTHUMG00000171600.5";  
chr14 HAVANA transcript 100725705 100732070 . + . gene_id "ENSG00000185559.16"; transcript_id "ENST00000392848.9"; gene_type "protein_coding"; gene_name "DLK1"; transcript_type "protein_coding"; transcript_name "DLK1-203"; level 2; protein_id "ENSP00000376589.5"; transcript_support_level "4"; hgnc_id "HGNC:2907"; tag "alternative_5_UTR"; tag "mRNA_end_NF"; tag "cds_end_NF"; havana_gene "OTTHUMG00000171600.5"; havana_transcript "OTTHUMT00000414388.2";  
chr14 HAVANA exon 100725705 100725833 . + . gene_id "ENSG00000185559.16"; transcript_id "ENST00000392848.9"; gene_type "protein_coding"; gene_name "DLK1"; transcript_type "protein_coding"; transcript_name "DLK1-203"; exon_number 1; exon_id "ENSE00002444405.1"; level 2; protein_id "ENSP00000376589.5"; transcript_support_level "4"; hgnc_id "HGNC:2907"; tag "alternative_5_UTR"; tag "mRNA_end_NF"; tag "cds_end_NF"; havana_gene "OTTHUMG00000171600.5"; havana_transcript "OTTHUMT00000414388.2";
```

OTHER TOOLS

- samtools and bcftools <https://www.htslib.org/>
- multiqc
https://www.hadriengourle.com/tutorials/data/fastqc/multiqc_report.html
- bbtools <https://jgi.doe.gov/data-and-tools/software-tools/bbtools/>
- trimmomatic <http://www.usadellab.org/cms/?page=trimmomatic>
- fastp <https://github.com/OpenGene/fastp>

CHECK QUEST SOFTWARE AND APPLICATIONS

- <https://www.it.northwestern.edu/departments/it-services-support/research/computing/quest-software-and-applications.html>

Quest Software and Applications

This page provides a self-service, interactive tool to search for software and applications commonly used on Quest, the University's high-performance computing cluster. Choose to search by name or explore your options using various filtering criteria. Click on any item in the software and applications list to expand a dropdown and view more details.

Please note that this Quest software and applications list is subject to change, and additional software not listed here may be available. Please log into Quest and run the command `module available` for the complete list. Except where noted, the [Modules Software Environment Manager](#) must be used to set up your environment to use the Quest software.

Filters

[Clear All Filters](#)

Search software by keyword
(separate multiple keywords by commas)

Quest Software and Applications

abaqus
abinit
ActiveTcl

CHECK QUEST SOFTWARE AND APPLICATIONS

- <https://www.it.northwestern.edu/departments/it-services-support/research/computing/quest-software-and-applications.html>
- Check software version on QUEST
`module spider <SoftwareName>` (i.e `module spider fastqc`)