1) Table from ABySS Output (unitigs, contigs, scaffolds)

	n — Number of Each	N50 of Each	sum — Predicted Genome Length
Unitigs	5770	5876	3922377 bp
Contigs	5351	7113	3946520 bp
Scaffolds	5256	7846	3950660 bp

2) Function of each ABySS command included in my code:

- **abyss-pe** → This is the main command that allows the abyss assembly to start running.
- **name=assembly** → This allows all output files to have the prefix “assembly” in front of it.
- **k** → This sets the k-mer length that is used in the de Bruijn graph construction.
- **B** → This controls memory usage.
- **in=** → This defines the input files.

3) Can you identify how you could modify the code you used to do a hybrid assembly with nanopore reads? Please explain what a hybrid assembly is and why someone might want to do that.

A hybrid assembly combines long-read and short-read sequencing data. Someone might want to do it this way because both long-read and short-read sequencing data have their own pros and cons, and combining them can improve the overall quality of the genome assembly. To modify the code for a hybrid assembly with nanopore reads, you can use the `--nanopore` code.

4) Quast Report Screenshots for:

A. Spades Assembly

Report

	scaffolds
# contigs (>= 0 bp)	221
# contigs (>= 1000 bp)	51
# contigs (>= 5000 bp)	28
# contigs (>= 10000 bp)	22
# contigs (>= 25000 bp)	19
# contigs (>= 50000 bp)	18
Total length (>= 0 bp)	4106590
Total length (>= 1000 bp)	4022576
Total length (>= 5000 bp)	3968146
Total length (>= 10000 bp)	3917295
Total length (>= 25000 bp)	3862015
Total length (>= 50000 bp)	3825088
# contigs	130
Largest contig	475358
Total length	4071672
GC (%)	47.41
N50	344343
N90	57327
auN	291329.0
L50	5
L90	16
# N's per 100 kbp	2.46

All statistics are based on contigs of size >= 500 bp, unless otherwise noted (e.g., "# contigs (>= 0 bp)" and "Total length (>= 0 bp)" include all contigs).

B. Abyss Assembly

Report

	assembly-scaffolds
# contigs (>= 0 bp)	5256
# contigs (>= 1000 bp)	668
# contigs (>= 5000 bp)	271
# contigs (>= 10000 bp)	104
# contigs (>= 25000 bp)	4
# contigs (>= 50000 bp)	0
Total length (>= 0 bp)	4737249
Total length (>= 1000 bp)	3877640
Total length (>= 5000 bp)	2739061
Total length (>= 10000 bp)	1578836
Total length (>= 25000 bp)	137855
Total length (>= 50000 bp)	0
# contigs	776
Largest contig	47530
Total length	3955540
GC (%)	47.47
N50	7846
N90	2539
auN	9901.0
L50	149
L90	495
# N's per 100 kbp	123.37

All statistics are based on contigs of size >= 500 bp, unless otherwise noted (e.g., "# contigs (>= 0 bp)" and "Total length (>= 0 bp)" include all contigs).

5) Based on the statistics from your genome, which assembly do you think is best? Why?

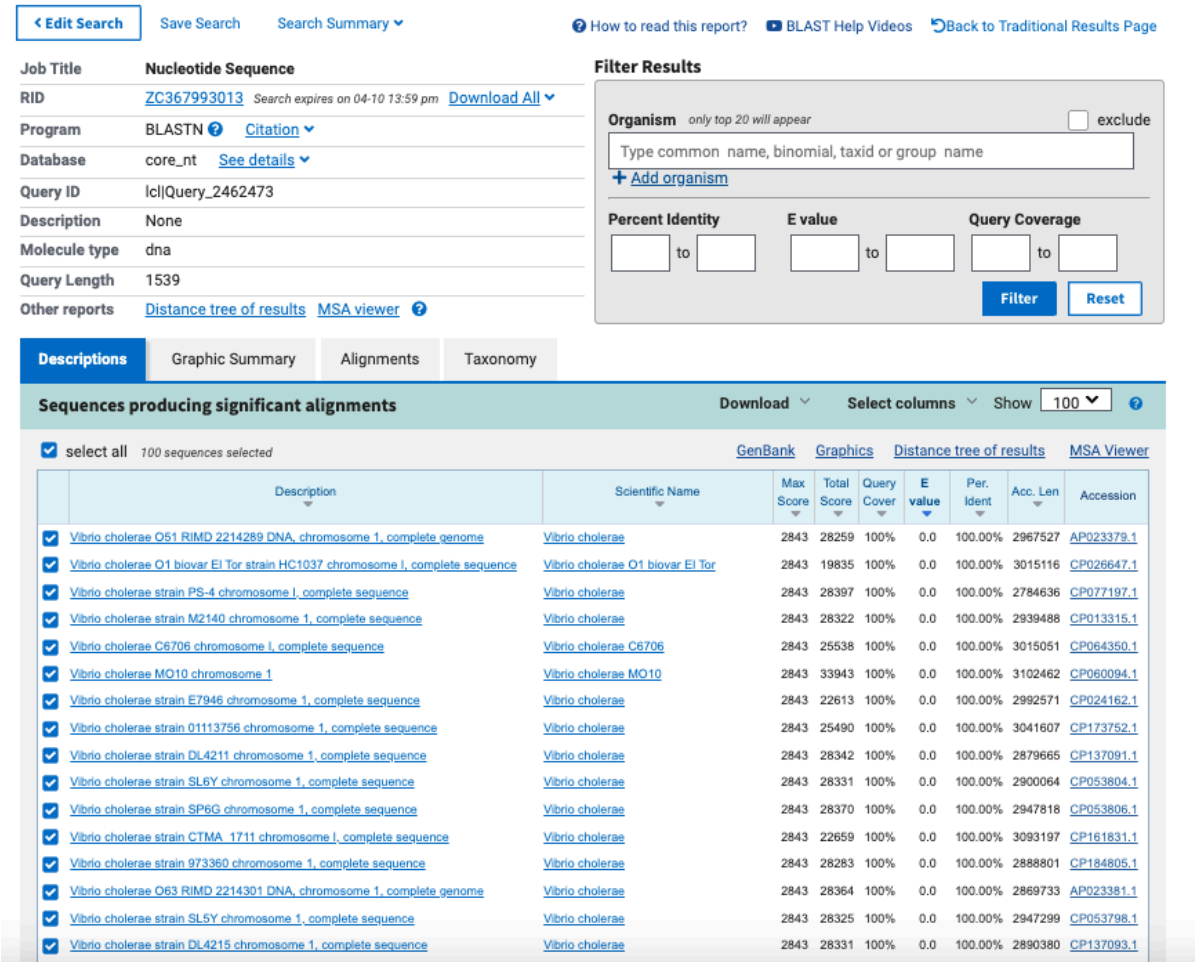
From looking at both Quast results, I determined that the Spades assembly is better than the Abyss assembly. This is because it has a higher N50 value (344,343), which means that it is more contiguous and has a higher quality. It also has fewer contigs (130), which suggests that it has better scaffolding.

6) How can we use barrnap to identify which species we have? Why is using the 16S rRNA sequence a good, but imperfect tool for figuring out species identity?

We can use barrnap to identify our species because it is a bacterial ribosomal RNA predictor, which means that it can identify ribosomal RNA genes in a given genome assembly. Using the 16S rRNA sequence is good because it is not as large as the 23S rRNA, and it is still long enough to narrow the search down to a specific organism. It is an imperfect tool because many species have a nearly identical 16S, which doesn't help clarify which is which. It also cannot detect horizontal gene transfer.

7) What species do you have? Include a screenshot of your top NCBI results.

My NCBI results showed that I had *Vibrio cholerae*.



8) What is genome annotation? Why is it important to do that?

Genome annotation allows you to identify functional elements in a given DNA sequence. It's an important step because it allows people to understand different gene functions and to do comparative genomic analyses.

9) Perform a genome annotation using two different programs. Find 3 of the 5 genes/features in your results file and create a table of those results: recA, gyrA, 16S rRNA, rpsB, dnaA. What is the location of the genes you chose? What does each program tell you about the gene? How are the outputs different between the two programs?

Gene Name	**Gene Location**	**What does DFAST tell me about this gene?**	**What does RAST tell me about this gene?**
recA	NODE_5_length_344343_cov_63.802830_186319_185255	For each gene, DFAST shows the gene, locus tag,	For each gene, RAST shows the contig_id, feature_id, type,

gyrA	NODE_4_length_403094_cov_55.389295_243426_246110	amino acid translation sequence, and codon start.	location, function, nucleotide sequence, and amino acid sequence.
dnaA	NODE_14_length_75234_cov_60.806571_46022_44619		

<u>How are the outputs different between the two programs?</u>
Between the two programs, the outputs are different in the way they operate and present information. DFAST operates entirely through the terminal and it saves my output folders directly into my genome assembly folder on my desktop. On the other hand, RAST operated on a webpage and I had to manually download the files I wanted and organize them in my genome assembly folder.

10) Download two related genomes from NCBI and run fastANI for determining Average Nucleotide Identity. Create a table for your ANI results. How do you interpret these results? What do each of the columns represent?

Species	Similarity	Mappings	Query Fragments
Vibrio vulnificus	79.1288	401	1317
Vibrio harveyi	78.581	342	1317

How do you interpret these results?
These results tell me that *Vibrio vulnificus* is slightly more closely related to *Vibrio cholerae* (79.1288%) than *Vibrio harveyi* is to *Vibrio cholerae (*78.581%).

What do each of the columns represent?
The similarity column tells me how closely related a given species is to my original species; it's the average nucleotide identity. The mappings column tells me the number of high-similarity DNA fragments that were used to calculate the ANI. The query fragments tell me the number of fragments generated from the query genome during analysis.