Next Generation Sequencing and Annotation of Mitogenome

Introduction

DNA Sequencing is a method of determining the exact order of nucleotides in a stretch of DNA. Multiple methods can be used to find this, but the latest method is Next Generation Sequencing(NGS). NGS is a fast and cost-effective, large scale sequencing method that allows us to sequence 1000s of DNA molecules at one time. These high-throughput sequencers can generate a large number of DNA sequences. However, these sequences are raw, and will have a lot of contamination.

FastQC can be used to do quality control checks of this raw data, and can be used to find the per base quality, per sequence quality as well as adapter content, among other things. Raw data will typically have a lot of adapter content and many low quality reads. In order to draw meaningful conclusions from the raw data, it needs to be cleaned first.

Adapter sequences are oligonucleotides attached to the ends of the DNA sequences. While they were useful during PCR amplification for helping primers to attach to them, and in some cases may also contain unique barcoding sequences, they need to be removed as they contaminate the data. These need to be removed from the original genome, along with any low quality reads.

Once the data is cleaned, genome assembly can be done. NOVOPlasty is used to perform *de novo* assembly(from scratch) of organelle genomes using multiple sequences.

The genome, once assembled, can be annotated using softwares like MITOS. MITOS is a web server used to perform a structural assembly of mitochondrial genomes. It helps identify the coding sequences, as well as the regions which code for rRNA and tRNA.

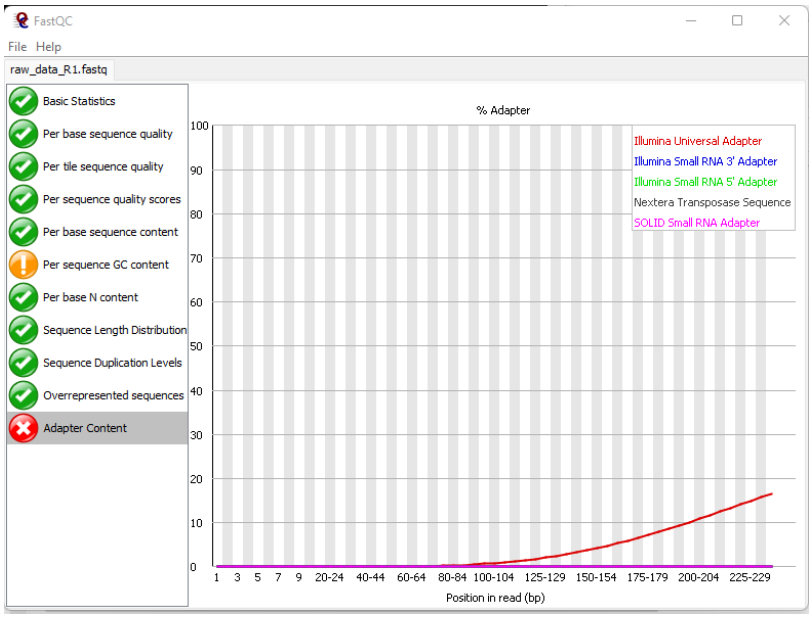
Methods and Results

i) Step 1: Obtain Raw data and check its quality using FastQC:

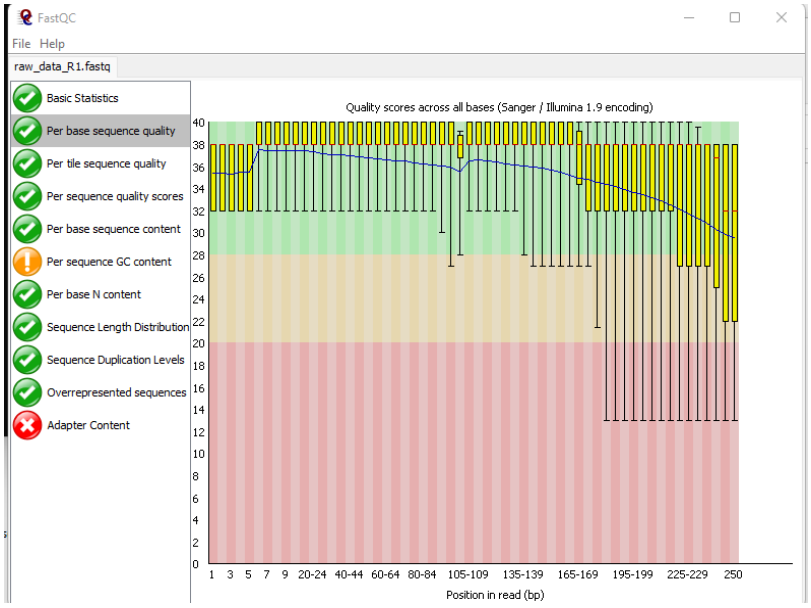
> Here the raw data was untrimmed paired mitochondrial genome sequences. FastQC was used to analyze adapter Content, Per Base Quality and Per Sequence Quality Scores.

Raw Data 1:

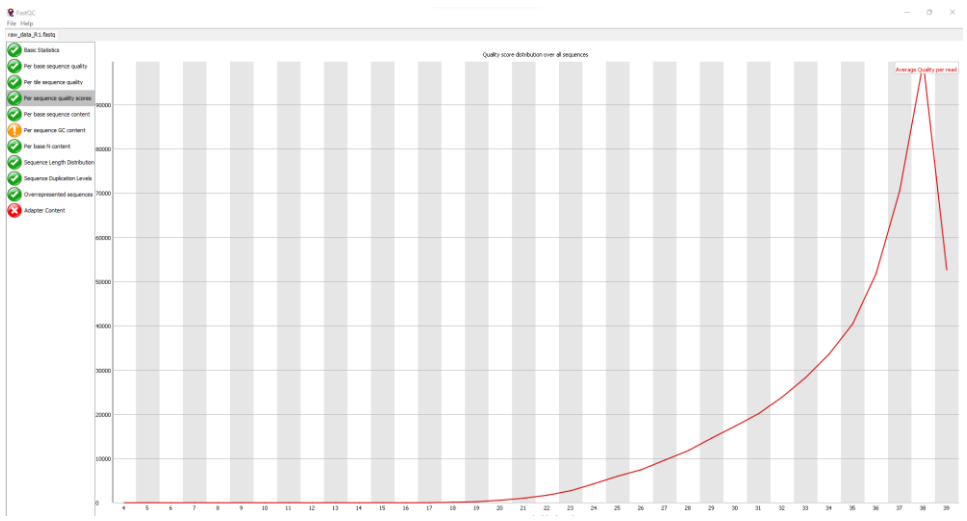
a. Adapter Content:



b. Per Base Sequence Quality

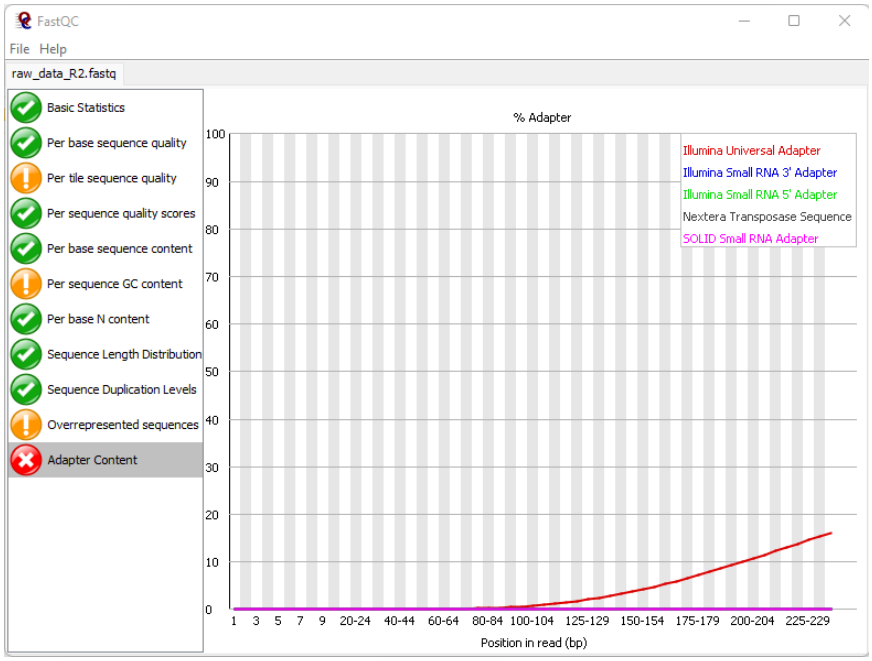


c. Per Sequence Quality Score

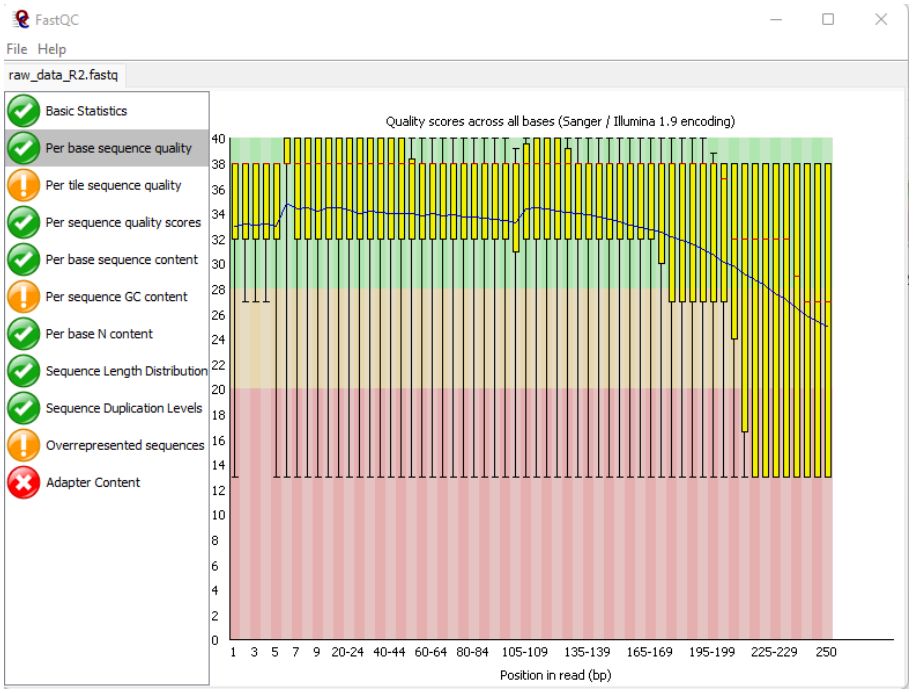


Raw Data 2:

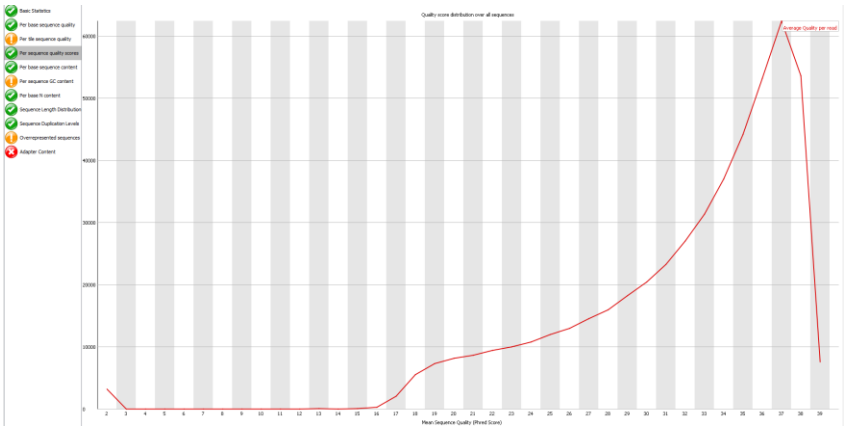
a. Adapter Content



b. Per Base Sequence Quality



c. Per Sequence Quality Score



We observed that there is some adapter contamination in both Raw Data 1 and 2. The red line, curving upwards, showed that there is some amount of the Illumina Universal adapter in both sequences.

We also saw that there are a lot of low quality Base scores in both sequences. Many bases had box plots that extended towards the red region in the graph, showing that, for these bases, they had a lot of low quality reads.

We saw that the Per Sequence Quality Score is good for both sequences. The peak, which depicts the average quality score that most sequences have, was high for both. In Raw Data 1, over 90,000 sequences have an average score of 38 while for Raw Data 2, more than 60,000 sequences have scores of 37.

ii) Step 2: Trimming of Adapter sequences and low quality reads:

The adapter content along with low quality reads were removed using Trimmomatic software. The code to do that is as follows:
java -jar trimmomatic-0.39.jar PE raw_data_R1.fastq raw_data_R2.fastq trimmed_sanch_fp.fastq trimmed_sanch_fup.fastq trimmed_sanch_rp.fastq trimmed_sanch_rup.fastq ILLUMINACLIP:TruSeq3-PE.fa:2:30:10:2:True LEADING:3 TRAILING:3 MINLEN:36

```
sanchitha@AU-CSLAB23:/mnt/c/Users/CSLAB/Desktop/NGS_Practical$ ls
NOVOPlasty4.3.1.pl  code_ngs_practicals.txt  log_Test.txt       raw_data_R2.fastq
sanchitha@AU-CSLAB23:/mnt/c/Users/CSLAB/Desktop/NGS_Practical$ java -jar trimmomatic-0.39.jar PE raw_data_R1.fastq raw_data_R2.fastq trimmed_sanch_fp.fq.gz trimmed_sanch_fup.fq.gz trimmed_sanch_rp.fq.gz trimmed_sanch_rup.fq.gz ILLUMINACL
IP:TruSeq3-PE.fa:2:30:10:2:True LEADING:3 TRAILING:3 MINLEN:36
TrimmomaticPE: Started with arguments:
 raw_data_R1.fastq raw_data_R2.fastq trimmed_sanch_fp.fq.gz trimmed_sanch_fup.fq.gz trimmed_sanch_rp.fq.gz trimmed_sanch_rup.fq.gz ILLUMINACLIP:TruSeq3-PE.fa:2:30:10:2:True LEADING:3 TRAILING:3 MINLEN:36
Using PrefixPair: 'TACACTCTTTCCCTACACGACGCTCTTCCGATCT' and 'GTGACTGGAGTTCAGACGTGTGCTCTTCCGATCT'
ILLUMINACLIP: Using 1 prefix pairs, 0 forward/reverse sequences, 0 forward only sequences, 0 reverse only sequences
Quality encoding detected as phred33
```

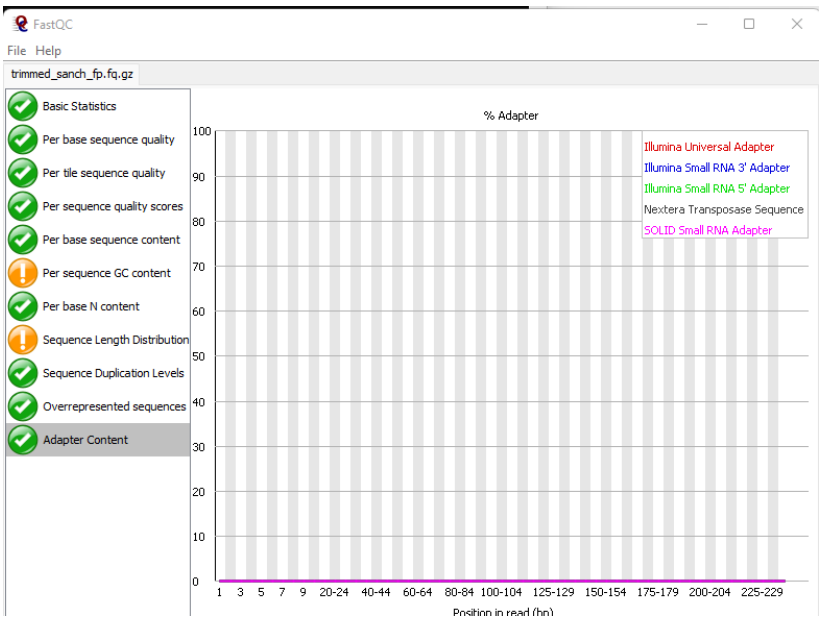
Trimmed Forward and Reverse paired sequence files were created and analyzed using FastQC to check if the adapters and low quality reads had been removed.

iii) Step 3: FastQC to Check Adapter Removal of Trimmed sequences:

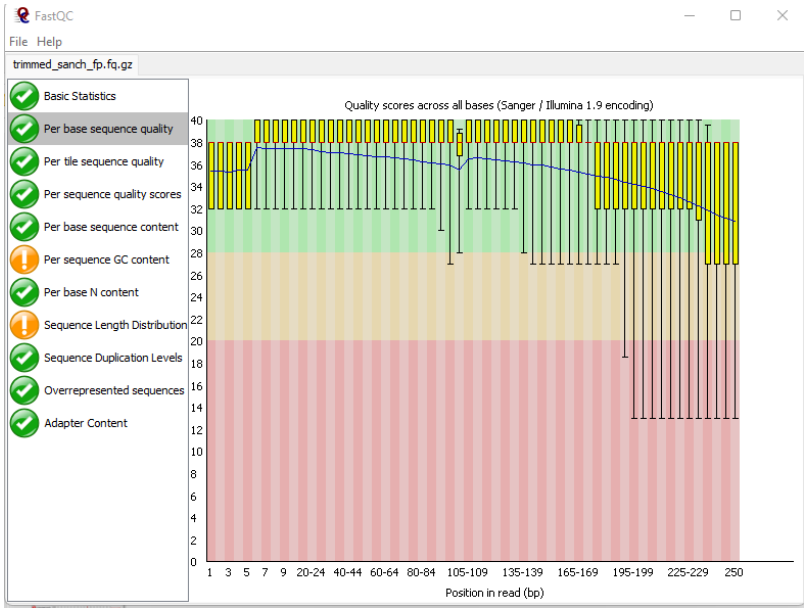
The FastQC results for the trimmed forward and reverse paired sequences are:

Forward Paired:

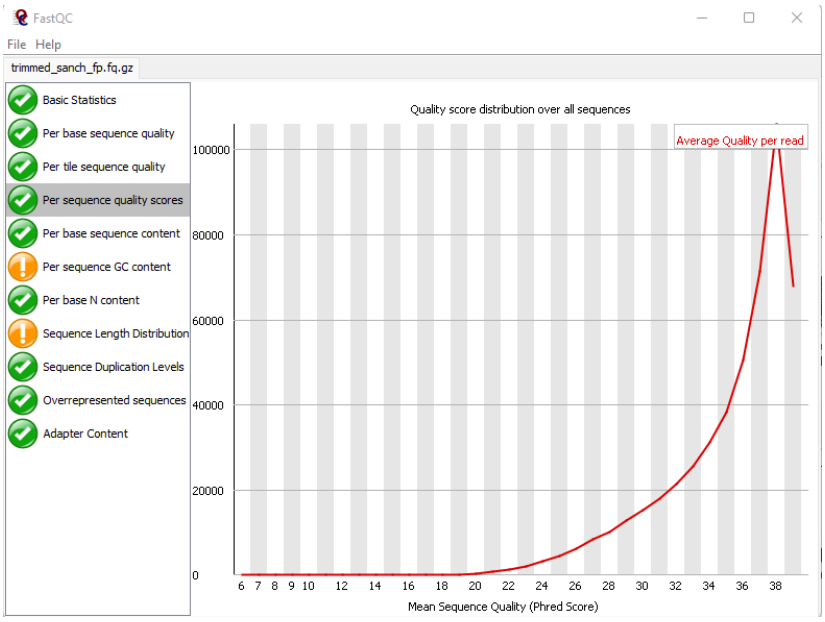
a. Adapter Content



b. Per Base Sequence Quality

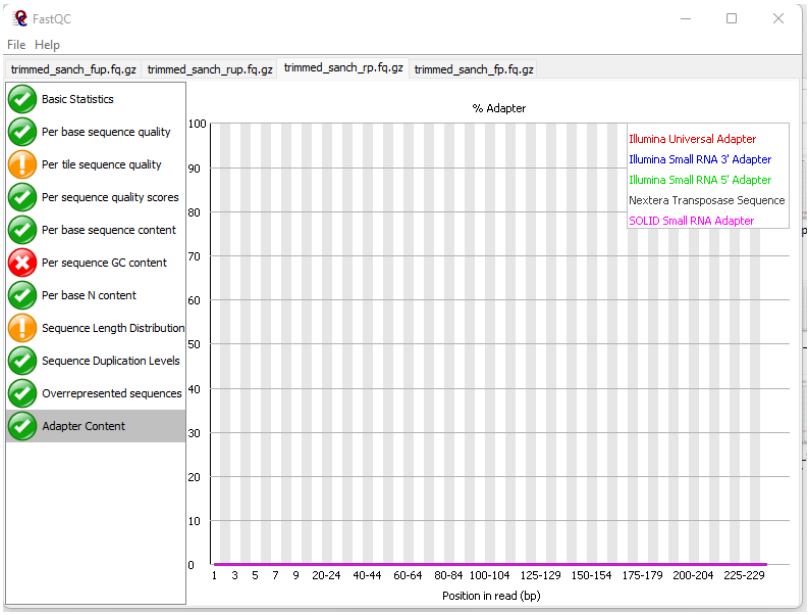


c. Per Sequence Quality Score

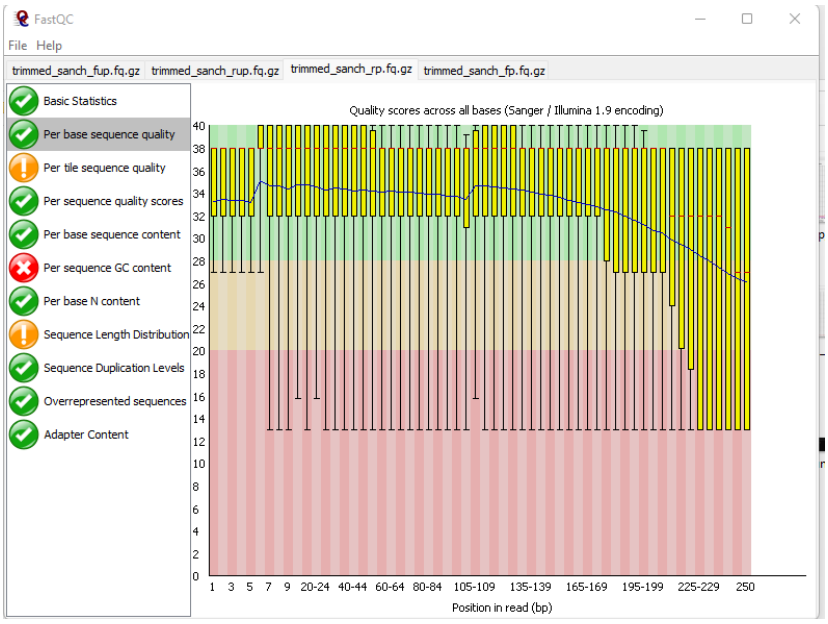


Reverse Paired

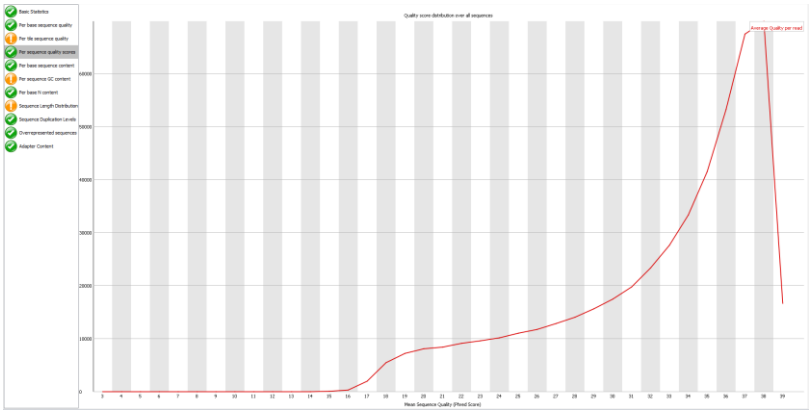
a. Adapter Content



b. Per Base Sequence Quality



c. Per Sequence Quality Score



We saw that the adapter content in both the Forward and Reverse Paired Sequences is 0. The Per Base Quality for the Forward Paired was a little better, as some low quality reads had been removed, increasing the average per base quality score. Thus, the bases with previously bad quality have a better quality. The Average Sequence Quality score remained high for both the forward and reverse paired sequences. However, more sequences in the forward paired sequences had this high quality score - over 1,00,000 sequences had high scores above 38.

iv) Step 4: Mitogenome Assembly:

The circularized Mitochondrial genome was assembled using the NOVOPlasty software.
In order to be able to assemble the genome, a seed sequence is required. This sequence is the area from where assembly begins. We used a single read(606 bp long) of a portion of cytochrome b mitochondrial gene of *Cynopterus sphinx* as the seed.
The details of the seed file, as well as the Forward and Reverse Paired sequences were entered into a Configuration text file.

config_ngs - Notepad

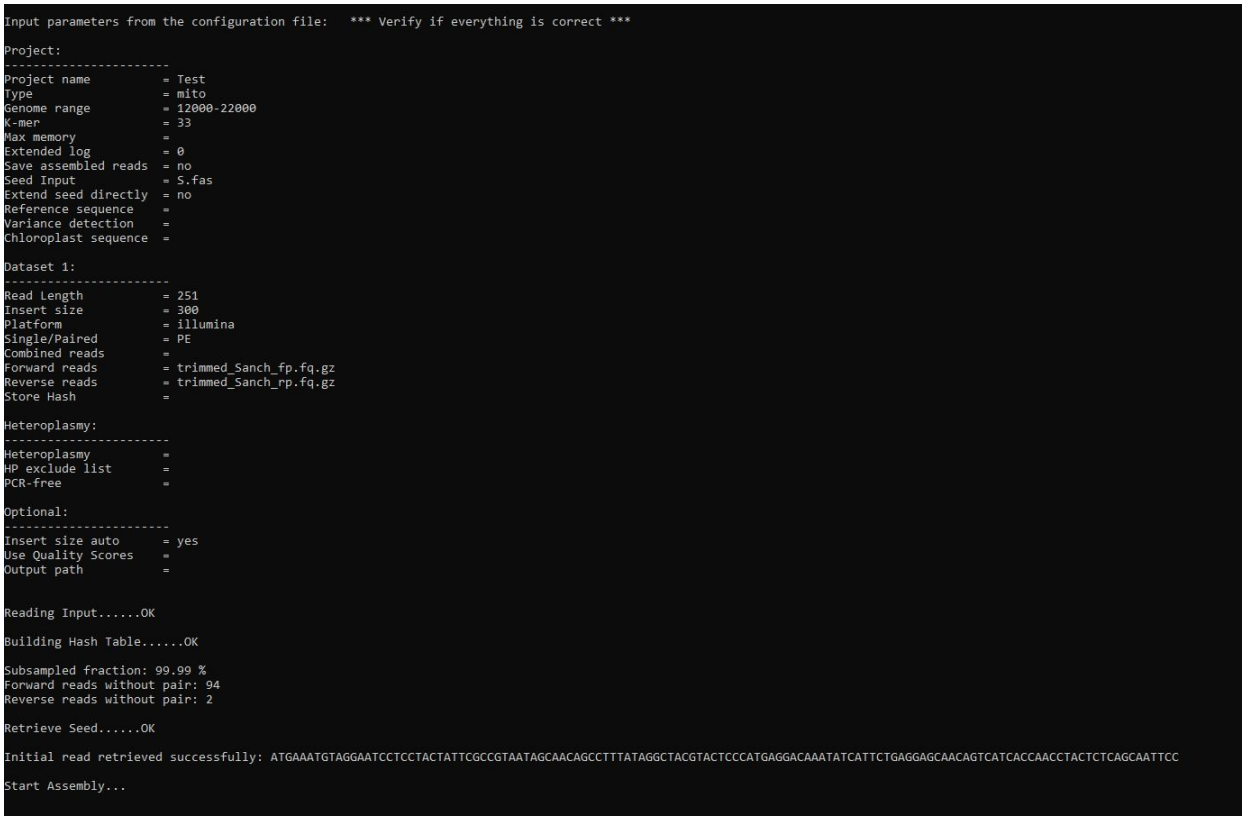
File Edit View

```
Project:
-----------------------
Project name          = Test
Type                  = mito
Genome Range          = 12000-22000
K-mer                 = 33
Max memory            =
Extended log          = 0
Save assembled reads  = no
Seed Input            = S.fas
Extend seed directly  = no
Reference sequence    =
Variance detection    =
Chloroplast sequence  =

Dataset 1:
-----------------------
Read Length           = 251
Insert size           = 300
Platform              = illumina
Single/Paired         = PE
Combined reads        =
Forward reads         = trimmed_Sanch_fp.fq.gz
Reverse reads         = trimmed_Sanch_rp.fq.gz
Store Hash            =

Heteroplasmy:
-----------------------
MAF                   =
HP exclude list       =
PCR-free              =

Optional:
-----------------------
```

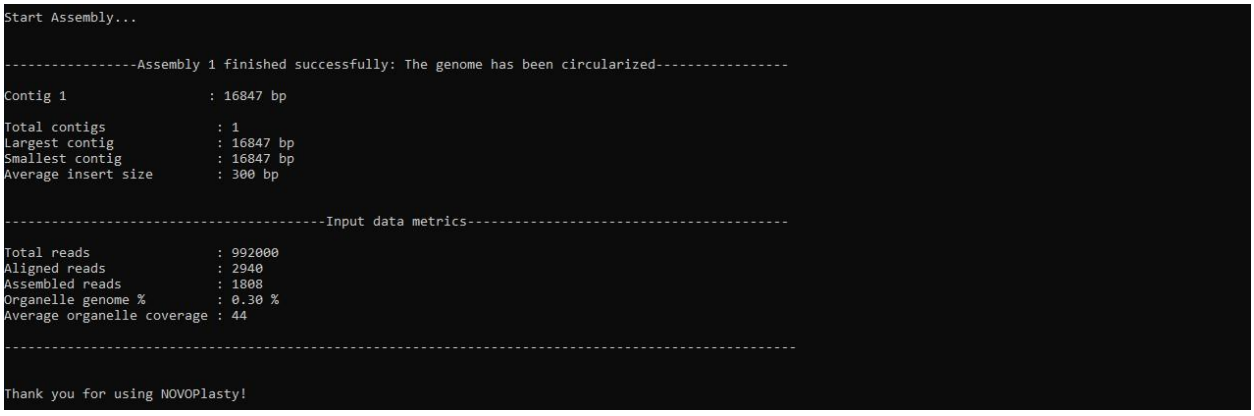
NOVOPlasty software is run using this file to assemble the mitochondrial genome, using the code: perl NOVOPlasty4.3.1.pl -c config_ngs.txt

```
Usage: perl NOVOPlasty4.3.1.pl -c config.txt
sanchitha@AU-CSLAB23:/mnt/c/Users/CSLAB/Desktop/NGS_Practical$ perl NOVOPlasty4.3.1.pl -c config_ngs.txt
```

The software assembled the given reads into a single contig (16847 bp long). A contig is a series of overlapping DNA sequences that forms a region of consensus of DNA. NOVOPlasty also generated a FASTA file, containing the circular DNA which had just been assembled.



```
Input parameters from the configuration file:   *** Verify if everything is correct ***

Project:
-----------------------
Project name          = Test
Type                  = mito
Genome range          = 12000-22000
K-mer                 = 33
Max memory            =
Extended log          = 0
Save assembled reads  = no
Seed Input            = S.fas
Extend seed directly  = no
Reference sequence    =
Variance detection    =
Chloroplast sequence  =

Dataset 1:
-----------------------
Read Length           = 251
Insert size           = 300
Platform              = illumina
Single/Paired         = PE
Combined reads        =
Forward reads         = trimmed_Sanch_fp.fq.gz
Reverse reads         = trimmed_Sanch_rp.fq.gz
Store Hash            =

Heteroplasmy:
-----------------------
Heteroplasmy          =
HP exclude list       =
PCR-free              =

Optional:
-----------------------
Insert size auto      = yes
Use Quality Scores    =
Output path           =


Reading Input......OK

Building Hash Table......OK

Subsampled fraction: 99.99 %
Forward reads without pair: 94
Reverse reads without pair: 2

Retrieve Seed......OK

Initial read retrieved successfully: ATGAAATGTAGGAATCCTCCTACTATTCGCCGTAATAGCAACAGCCTTTATAGGCTACGTACTCCCATGAGGACAAATATCATTCTGAGGAGCAACAGTCATCACCAACCTACTCTCAGCAATTCC

Start Assembly...
```



```
Start Assembly...


-----------------Assembly 1 finished successfully: The genome has been circularized----------------

Contig 1                  : 16847 bp

Total contigs              : 1
Largest contig             : 16847 bp
Smallest contig            : 16847 bp
Average insert size        : 300 bp


----------------------------------------Input data metrics----------------------------------------

Total reads                : 992000
Aligned reads              : 2940
Assembled reads            : 1808
Organelle genome %         : 0.30 %
Average organelle coverage : 44

--------------------------------------------------------------------------------------------------


Thank you for using NOVOPlasty!
```

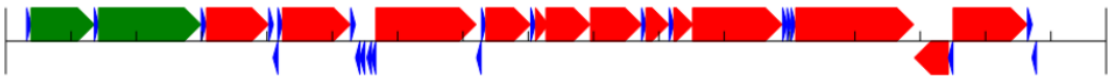
v) Step 5: Annotate Mitogenome Assembly:

MITOS server was used to perform a structural annotation of the mitochondrial genome. The newly generated FASTA file containing the circularized genome was entered into this server in order to identify the elements of the genome, such as the genes, non-coding regions, etc.

The MITOS results are as follows:

Name	Start	Stop	Strand	Length	Structure
trnF(ttc)	322	389	+	68	svg ps
rrnS	390	1356	+	967	svg ps
trnV(gta)	1357	1424	+	68	svg ps
rrnL	1423	2989	+	1567	svg ps
trnL2(tta)	2996	3070	+	75	svg ps
nad1	3073	4023	+	951	
trnI(atc)	4029	4097	+	69	svg ps
trnQ(caa)	4095	4168	-	74	svg ps
trnM(atg)	4169	4237	+	69	svg ps
nad2	4238	5275	+	1038	
trnW(tga)	5283	5352	+	70	svg ps
trnA(gca)	5356	5424	-	69	svg ps
trnN(aac)	5426	5498	-	73	svg ps
trnC(tgc)	5531	5599	-	69	svg ps
trnY(tac)	5600	5666	-	67	svg ps
cox1	5668	7200	+	1533	
trnS2(tca)	7210	7278	-	69	svg ps
trnD(gac)	7286	7352	+	67	svg ps
cox2	7353	8033	+	681	
trnK(aaa)	8040	8110	+	71	svg ps
atp8	8112	8306	+	195	
atp6	8273	8947	+	675	
cox3	8953	9735	+	783	
trnG(gga)	9737	9804	+	68	svg ps
nad3	9805	10149	+	345	
trnR(cga)	10160	10227	+	68	svg ps
nad4l	10228	10521	+	294	
nad4	10518	11885	+	1368	
trnH(cac)	11896	11963	+	68	svg ps
trnS1(agc)	11964	12022	+	59	svg ps
trnL1(cta)	12024	12093	+	70	svg ps
nad5	12094	13899	+	1806	
nad6	13914	14432	-	519	
trnE(gaa)	14433	14501	-	69	svg ps
cob	14506	15639	+	1134	
trnT(aca)	15645	15714	+	70	svg ps
trnP(cca)	15714	15779	-	66	svg ps

tRNA gene rRNA gene protein coding gene



MITOS helps to identify which parts of the genome are protein coding and which code for rRNA and tRNA. The above data in the table shows the different genes, where their start and stop codons are located, which strand they are located on, and the length of the gene. It also has links to view the secondary structure of the rRNA or tRNA coded for by genome.

The image at the bottom helps to visualize the annotation. Genes present on the plus strand are present above, while genes present on the minus strand are present below. The protein coding, tRNA and rRNA regions are all colored differently.
We can see that most of the genome we assembled consists of protein coding genes on the plus strand. There is a small portion in the beginning which codes for rRNA and regions throughout which code for tRNA.
We can also see that one of the genes(third last row) annotated is cob, i.e., cytochrome b, which corresponds to the seed which we used to initiate assembly. This sequence is 1134 bp long, according to MITOS, which is in line with previously known data that this gene is around 1140 bp long.

Works Cited

1. Bioinformatics, ecSeq. *Trimming Adapter Sequences - Is It Necessary?* www.ecseq.com/support/ngs/trimming-adapter-sequences-is-it-necessary.

2. Bioinformatics, ecSeq. *Trimming Adapter Sequences - Is It Necessary?* www.ecseq.com/support/ngs/trimming-adapter-sequences-is-it-necessary.

3. *MITOS Web Server Help. mitos.bioinf.uni-leipzig.de/help.py.*