

# GS scripts

#Research #fbbox #scripts

## Genome assembly

```
# trim
zcat fastq_pass/*.gz > raw.fq
porechop-runner.py -i raw.fq -o raw.t.fq &> trim.log &

# filter reads length > 5000
cat raw.t.fq|seqkit fx2tab|awk -F '\t' 'length($2)>5000'|
seqkit tab2fx > raw.t.5k.fq

# correct with canu
# results ${i}.5k.correctedReads.fasta.gz
canu -correct genomeSize=130m -d cor_5k -p canu_cor.5k
-nanopore raw.t.5k.fq

# flye assemble
python flye --nano-raw ${i}.5k.correctedReads.fasta.gz --
out-dir flye_out --genome-size 130m --threads 96 &>flye.log
&

# ragtag correct
ragtag.py correct ${i}.fa assembly.fasta --aligner nucmer
-o cor_out -f 5000 --nucmer-params '--mum --mincluster 100
--maxgap 300 -t 96'

# ragtag scaffold
ragtag.py scaffold ${i}.fa cor_out/ragtag.correct.fasta --
aligner nucmer -o scaf_out -f 5000 --nucmer-params '--mum --
mincluster 100 --maxgap 300 -t 96'
```

## Remove bacteria

```
cat ${i}.fa|seqkit fx2tab|grep -v Chr|seqkit tab2fx > $
{i}.unplaced.fa

blastn -db nt -num_threads 48 -query ${i}.unplaced.fa
-outfmt "6 qseqid sseqid pident length mismatch gapopen
qstart qend sstart send eval evalue bitscore stitle" > $
{i}.unplaced.blast.txt

cat ${i}.unplaced.blast.txt|grep 'Escherichia coli'|cut -f1|
sort|uniq|seqkit grep -v -f - ${i}.fa > clean/${i}.fa
```

## Polishing

```
bwa-mem2 index ${i}.fa
bwa-mem2 mem -t 48 -M ${i}.genome.fa *fq.gz|samtools sort -@
48 -o ${i}_R1.bam
```

```
java -Xmx24G -jar pilon-1.24.jar --genome ${i}.fa --frags ${i}_R1.bam
```

## **RepeatMasking**

```
# RepeatModeler
```

```
BuildDatabase -name ${i}DB ${i}.fa
```

```
RepeatModeler -database ${i}DB -pa 16 &> ${i}.RepeatModeler.log.txt &
```

```
getorf -minsize 90 ${i}DB-families.fa -outseq ${i}-families.orf.fa
```

```
#cel_pro = path to celegans protein
```

```
blastp -query ${i}-families.orf.fa -db $cel_pro -evaluate 1e-09 -outfmt 6 -seg yes -num_threads 12 -out ${i}.orf.blastp.txt
```

```
# run interpro to filter out those with protein domain  
interpro/interproscan:5.65-97.0 --input ${i}-families.orf.fa --disable-precvalc
```

```
# combine blastp and interpro results
```

```
cat ${i}.orf.blastp.txt |cut -d "#" -f1|sort|uniq|sed 's/$/#/g' > ${i}.orf.blastp.families.txt
```

```
cat ${i}-families.orf.fa.tsv|cut -d "#" -f1|sort|uniq|sed 's/$/#/g' > ${i}.orf.interpro.families.txt
```

```
cat ${i}.*.families.txt|sort|uniq > ${i}.filter.txt
```

```
# filter and make the final repeat library
```

```
seqkit fx2tab ${i}DB-families.fa|grep -vf ${i}.filter.txt|  
seqkit tab2fx > ${i}-families.filtered.fa
```

```
# RepeatMasker
```

```
RepeatMasker -lib ${i}-families.filtered.fa -xsmall -pa 6 -gff ${i}.fa -e ncbi -gccalc &>${i}.RepeatMasker.log &
```

## **Prediction - mapping**

```
trim_galore --cores 4 --paired *.fastq
```

```
STAR --runThreadN 24 --runMode genomeGenerate --genomeDir ${i}_index --genomeFastaFiles ${i}.fa
```

```
# mapping
```

```
STAR --runThreadN 24 --genomeDir ${i}_index --outFileNamePrefix ${i} --readFilesIn *.fq
```

```
# samtools sort results
```

```
samtools sort ${i}Aligned.out.sam -@ 24 -o ${i}.bam  
samtools index ${i}.bam
```

```
# we can use samtools flagstat to check the mapping quality
samtools flagstat ${i}.bam
```

### **Prediction - AUGUSTUS**

```
bam2wig ${i}.bam > ${i}.wig
cat ${i}.wig |wig2hints.pl --width=10 --margin=10 --
minthresh=2 --minscore=4 --src=W --type=ep --radius=4.5 > $
{i}.hints.exon.gff
```

```
bam2hints --in AF.bam --out=${i}.hints.intron.gff
```

```
cat *.gff > ${i}.hints.gff
```

```
augustus --extrinsicCfgFile=extrinsic.M.RM.E.W.cfg --
softmasking=True --hintsfile=${i}.hints.gff --
uniqueGeneId=true --protein=on --introns=on --start=on --
stop=on --cds=on --codingseq=on --gff3=on --progress=true --
species=caenorhabditis ${i}.fa.masked > ${i}.gff 2>$
{i}.gff.log
```

### **OrthoFinder**

```
# OrthoFinder
orthofinder -f . -M msa -A muscle -T iqtree -X &>
orthofinder.log &
```