New annotation Dec2020

STEP 5

working directory:
/data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

in the working directory there must be:

- softmasked genome as “$species”_softmasked.fa
- mikado.loci.gff3
- Augustus output as “$species”.aug.out
- alignAssembly.config and annotCompare.config with the right paths specified

agat_convert_sp_gxf2gxf.pl and fgrep

```
module load anaconda3
conda activate agat_env
agat_convert_sp_gxf2gxf.pl -g mikado.loci.gff3 -o mikado.loci.AGAT.gff3
#grep "ncRNA" mikado.loci.AGAT.gff3 | awk '{print $9}' | uniq > mikado.ncRNA.IDs
grep "ncRNA" mikado.loci.AGAT.gff3 | awk '{print $9}' | uniq | sed "s/=/\t/" | sed "s/;/\t/" | awk '{print $2}' > mikado.ncRNA.IDs
fgrep -v -w -f mikado.ncRNA.IDs mikado.loci.AGAT.gff3 > mikado.loci.AGAT.NOncRNA.gff3
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

gffread_universal_5.1_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
```

```
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 4
#$ -l h_vmem=5G
#$ -l h_rt=24:0:0

species=$1
species_softmasked="$species"_softmasked.fa
augustus_output="$species".aug.out
output_mikado_transcripts="$species"_mikado_transcripts_NoncRNA.fa
output_augustus_genes="$species".Augustus.genes.fa

echo "Working on "$species

module load anaconda3
source activate augustus

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

gffread -C -x $output_mikado_transcripts -g $species_softmasked mikado.loci.AGAT.NOncRNA.gff3

gffread $augustus_output -V -w $output_augustus_genes -g $species_softmasked
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

seqclean_universal_5.2_v1.sh

```
#!/bin/bash
```

```
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 4
#$ -l h_vmem=4G
#$ -l h_rt=24:0:0

species=$1
mikado_transcripts="$species"_mikado_transcripts_NoncRNA.fa

echo "Working on "$species

module load anaconda3
source activate pasa

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

/data/home/btx654/.conda/envs/pasa/opt/pasa-2.4.1/bin/seqclean $mikado_transcripts
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

pasa_universal_5.3_v1.highmem.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 24
#$ -l h_vmem=25G
```

```
#$ -l h_rt=240:0:0
#$ -l highmem

species=$1
species_softmasked="$species"_softmasked.fa
mikado_transcripts="$species"_mikado_transcripts_NoncRNA.fa
mikado_transcripts_clean="$species"_mikado_transcripts_NoncRNA.fa.clean

echo "Working on "$species

module load anaconda3
source activate pasa
module load samtools

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

/data/home/btx654/.conda/envs/pasa/opt/pasa-2.4.1/Launch_PASA_pipeline.pl -c alignAssembly.config -C -R -g $species_softmasked -t $mikado_transcripts_clean -T -u $mikado_transcripts --ALIGNERS blat --CPU 24
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

pasa_universal_5.4_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 2
```

```
#$ -l h_vmem=10G
#$ -l h_rt=48:0:0

species=$1
species_softmasked="$species"_softmasked.fa
augustus_output="$species".aug.out
augustus_gtf="$species".aug.gtf

echo "Working on "$species

module load anaconda3
source activate augustus

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

gffread $augustus_output -E -T -o $augustus_gtf

conda deactivate
source activate pasa
module load samtools/1.9

/data/home/btx654/.conda/envs/pasa/opt/pasa-2.4.1/scripts/Load_Current_Gene_Annotations.dbi -c alignAssembly.config -g $species_softmasked -P $augustus_gtf
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

pasa_universal_5.5_v1.highmem.sh

```
#!/bin/bash
```

```
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 30
#$ -l h_vmem=3G
#$ -l h_rt=48:0:0
#$ -l highmem

species=$1
species_softmasked="$species"_softmasked.fa
mikado_transcripts_clean="$species"_mikado_transcripts_NoncRNA.fa.clean

echo "Working on "$species

module load anaconda3
source activate pasa
module load samtools

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

/data/home/btx654/.conda/envs/pasa/opt/pasa-2.4.1/Launch_PASA_pipeline.pl -c annotCompare.config -A -g $species_softmasked -t $mikado_transcripts_clean --CPU 30
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

change name pasa gff3 output

```
mv sqlite_db.gene_structures_post_PASA_updates.16689.gff3 oasisia_pasa_FirstStep.gff3
```

```
mv sqlite_db.gene_structures_post_PASA_updates.20574.gff3 riftia_pasa_FirstStep.gff3
mv sqlite_db.gene_structures_post_PASA_updates.47524.gff3 osedax_pasa_FirstStep.gff3
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

pasa_universal_5.6_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 2
#$ -l h_vmem=10G
#$ -l h_rt=48:0:0

species=$1
species_softmasked="$species"_softmasked.fa
pasa_first_step_gff3="$species"_pasa_FirstStep.gff3
pasa_first_step_gtf="$species"_pasa_FirstStep.gtf


module load anaconda3
source activate augustus


cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5

gffread $pasa_first_step_gff3 -E -T -o $pasa_first_step_gtf

```

```
conda deactivate
source activate pasa
module load samtools/1.9

/data/home/btx654/.conda/envs/pasa/opt/pasa-2.4.1/scripts/Load_Current_Gene_Annotations.dbi -c alignAssembly.config -g $species_softmasked -P $pasa_first_step_gtf
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

Repeating “pasa_universal_5.5_v1.highmem.sh”

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

change name pasa gff3 and bed outputs

```
mv sqlite_db.gene_structures_post_PASA_updates.45674.gff3 riftia_pasa_SecondStep.gff3
mv sqlite_db.gene_structures_post_PASA_updates.45674.bed riftia_pasa_SecondStep.bed
cp riftia_pasa_SecondStep.* /data/SBCS-MartinDuranLab/03-Giacomo/data/riftia/annotation/New_annotation_Dec2020/step5/
mv sqlite_db.gene_structures_post_PASA_updates.37064.gff3 oasisia_pasa_SecondStep.gff3
mv sqlite_db.gene_structures_post_PASA_updates.37064.bed oasisia_pasa_SecondStep.bed
cp oasisia_pasa_SecondStep.* /data/SBCS-MartinDuranLab/03-Giacomo/data/oasisia/annotation/New_annotation_Dec2020/step5/
mv sqlite_db.gene_structures_post_PASA_updates.11480.gff3 osedax_pasa_SecondStep.gff3
mv sqlite_db.gene_structures_post_PASA_updates.11480.bed osedax_pasa_SecondStep.bed
cp osedax_pasa_SecondStep.* /data/SBCS-MartinDuranLab/03-Giacomo
```

```
/data/osedax/annotation/New_annotation_Dec2020/step5/
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

STEP 6

create a folder which can be deleted after this step6:
/data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation
/repeatmasker_delete_after_step6_Dec2020
containing the output of repeatmasker named as:

- "$species"_repeatmasker.fa.out

filtering_universal_6.1_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 2
#$ -l h_vmem=3G
#$ -l h_rt=24:0:0

species=$1
species_softmasked="$species"_softmasked.fa
species_softmasked_path=/data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5/$species_softmasked
pasa_second_step_gff3="$species"_pasa_SecondStep.gff3
pasa_second_step_path=/data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step5/$pasa_second_step_gff3
pasa_second_step_AGAT="$species"_pasa_SecondStep.AGAT.gff3
output_gff3="$species".AGAT.noSTOP.gff3
```

```

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/

mkdir -p step6
cd step6

cp $species_softmasked_path ./
cp $pasa_second_step_path ./

module load anaconda3
conda activate agat_env

agat_convert_sp_gxf2gxf.pl -g $pasa_second_step_gff3 -o $pasa_second_step_AGAT

conda deactivate
source activate augustus

gffread -E $pasa_second_step_AGAT -g $species_softmasked -V -H -o $output_gff3
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

```
grep -c "^Warning: In-frame STOP found for" filtering_universal_6.1_v1.sh.o1397518 #291 riftia
grep -c "^Warning: In-frame STOP found for" filtering_universal_6.1_v1.sh.o1397519 #451 oasisia
grep -c "^Warning: In-frame STOP found for" filtering_universal_
```

```
6.1_v1.sh.o1399077 #232 osedax
```

filtering_universal_6.2_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 2
#$ -l h_vmem=5G
#$ -l h_rt=24:0:0

species=$1
pasa_gff3="$species".AGAT.noSTOP.gff3
pasa_gtf="$species".AGAT.noSTOP.gtf
repeatmasker="$species"_repeatmasker.fa.out
repeatmasker_path=/data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/repeatmasker_delete_after_step6_Dec2020/$repeatmasker
output_filt_gtf="$species".AGAT.noSTOP.filt.gtf
output_filt_gff3="$species".AGAT.noSTOP.filt.AGAT.gff3
species_softmasked="$species"_softmasked.fa

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step6/

module load anaconda3
source activate augustus

```

```
gffread -E $pasa_gff3 -T -o $pasa_gtf

conda deactivate
module unload anaconda3

module load python/2.7.15
module load bedtools

cp $repeatmasker_path ./
sed -e 's/.arrow.arrow.pilon.pilon//' $repeatmasker > repeatmasker_sed.fa.out

python2 /data/SBCS-MartinDuranLab/03-Giacomo/src/various/rep2bed.py repeatmasker_sed.fa.out > RepeatMasker.bed
python2 /data/SBCS-MartinDuranLab/03-Giacomo/src/various/filt-rep-gtf.py $pasa_gtf RepeatMasker.bed

module unload python/2.7.15
module unload bedtools
module load anaconda3
conda activate agat_env

agat_convert_sp_gxf2gxf.pl -g $output_filt_gtf -o $output_filt_gff3
agat_sq_stat_basic.pl -i $output_filt_gff3 -g $species_softmasked
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

Results:

- Riftia

```
238207 exons...
initial number of genes: 43443
number of genes after filtering 37455

Type (3rd column)        Number        Size total (kb)        Size mean (bp)        % of the genome        /!\Results are rounding to two decimal places
cds        226079        55718.80        246.46        10.07
exon        226211        55876.82        247.01        10.09
five_prime_utr        1531        91.68        59.89        0.02
gene        37455        310501.42        8289.99        56.09
three_prime_utr        236        66.33        281.06        0.01
transcript        38594        342050.72        8862.80        61.79
Total        530106        764305.77        1441.80        138.07
```

- Oasisia

```
337078 exons...
initial number of genes: 62270
number of genes after filtering 37929

Type (3rd column)        Number        Size total (kb)        Size mean (bp)        % of the genome        /!\Results are rounding to two decimal places
cds        279272        62067.95        222.25        7.68
exon        279538        62212.03        222.55        7.70
five_prime_utr        1993        102.59        51.47        0.01
gene        37929        419731.45        11066.24        51.95
three_prime_utr        125        41.49        331.94        0.01
```

```
transcript        39850        482789.42        12115.17
 59.76
Total        638707        1026944.94        1607.85        127.
11
```

- Osedax

```
185496 exons...
initial number of genes: 21969
number of genes after filtering 18176

Type (3rd column)        Number        Size total (kb)        Si
ze mean (bp)        % of the genome        /!\Results are roundi
ng to two decimal places
cds        174271        29205.60        167.59        10.26
exon        174426        29269.47        167.80        10.28
five_prime_utr        1278        47.57        37.22        0.02
gene        18176        173417.61        9541.02        60.92
three_prime_utr        32        16.30        509.44        0.01
transcript        18808        183673.13        9765.69        6
4.52
Total        386991        415629.70        1074.00        146.0
0
```

filtering_universal_6.3_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 8
#$ -l h_vmem=5G
#$ -l h_rt=24:0:0

species=$1
```

```
pasa_gtf="$species".AGAT.noSTOP.filt.gtf
species_softmasked="$species"_softmasked.fa
pasa_prot_fasta="$species".AGAT.noSTOP.filt.prot.fasta
final_pasa_gtf="$species".AGAT.noSTOP.filt.noTE.gtf
final_pasa_prot_fasta="$species".AGAT.noSTOP.filt.noTE.prot.fasta
final_pasa_gff3="$species".AGAT.noSTOP.filt.noTE.AGAT.gff3

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step6/

module load anaconda3
source activate augustus

gffread -E $pasa_gtf -S -g $species_softmasked -y $pasa_prot_fasta

conda deactivate
module unload anaconda3

cp /data/SBCS-MartinDuranLab/03-Giacomo/src/RepeatMasker/Libraries/RepeatPeps.lib ./
module load diamond/0.9.22
diamond makedb --in RepeatPeps.lib -d RepeatPeps
diamond blastp -d RepeatPeps -q $pasa_prot_fasta -o pasa.vs.RepeatPeps.1e5.blastp -f 6 qseqid bitscore evalue stitle -k 25 -e 1e-5 -p 8

cat pasa.vs.RepeatPeps.1e5.blastp | cut -f 1 | sort | uniq > TEsIDs.txt
```

```
fgrep -w -v -f TEsIDs.txt $pasa_gtf > $final_pasa_gtf

#check if we still have TEs in our proteins
module load anaconda3
source activate augustus

gffread -E $final_pasa_gtf -S -g $species_softmasked -y $final_pasa_prot_fasta

conda deactivate
module unload anaconda3

module load diamond/0.9.22
diamond blastp -d RepeatPeps -q $final_pasa_prot_fasta -o pasa_noTE.vs.RepeatPeps.1e5.blastp -f 6 qseqid bitscore evalue stitle -k 25 -e 1e-5 -p 8
#check end

module load anaconda3
conda activate agat_env

agat_convert_sp_gxf2gxf.pl -g $final_pasa_gtf -o $final_pasa_gff3
agat_sq_stat_basic.pl -i $final_pasa_gff3 -g $species_softmasked
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

Results:

- Riftia

```
415 queries aligned.

```

```
Type (3rd column)        Number        Size total (kb)        Si
ze mean (bp)        % of the genome        /!\Results are roundi
ng to two decimal places
cds        224328        55173.67        245.95        9.97
exon        224460        55330.47        246.50        10.00
five_prime_utr        1520        90.92        59.81        0.02
gene        37043        307881.72        8311.47        55.62
three_prime_utr        233        65.88        282.76        0.0
1
transcript        38179        339383.55        8889.27        6
1.31
Total        525763        757926.20        1441.57        136.9
2
```

- Oasisia

```
2072 queries aligned.


Type (3rd column)        Number        Size total (kb)        Si
ze mean (bp)        % of the genome        /!\Results are roundi
ng to two decimal places
cds        270655        58746.82        217.05        7.27
exon        270916        58885.90        217.36        7.29
five_prime_utr        1970        100.30        50.91        0.0
1
gene        35869        405225.34        11297.37        50.16
three_prime_utr        116        38.78        334.31        0.0
0
transcript        37777        467927.45        12386.57
 57.92
Total        617303        990924.59        1605.25        122.6
5
```

- Osedax

```
151 queries aligned.

```

```
Type (3rd column)        Number        Size total (kb)        Size mean (bp)        % of the genome        /!\Results are rounding to two decimal places
cds        173041        28963.63        167.38        10.17
exon        173196        29026.81        167.60        10.20
five_prime_utr        1276        47.55        37.27        0.02
gene        18025        172119.65        9548.94        60.46
three_prime_utr        30        15.63        520.97        0.01
transcript        18657        182375.17        9775.16        64.07
Total        384225        412548.44        1073.72        144.92
```

rename_and_longest_isoform.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 1
#$ -l h_vmem=5G
#$ -l h_rt=2:0:0

species=$1
prefix=$2
final_pasa_gff3="$species".AGAT.noSTOP.filt.noTE.AGAT.gff3
final_pasa_gtf="$species".AGAT.noSTOP.filt.noTE.gtf
output_annotation="$species"_annotation_v101220.gff3
loci_merged="$species"_lociMerged.gff
longest_isoform="$species"_lociMerged_longestIsoform.gff

echo "Working on "$species" using prefix: "$prefix
```

```

mkdir /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step6
cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step6/

cp $final_pasa_gff3 /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step6/
cp $final_pasa_gtf /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step6/

cd /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step6/

module load anaconda3
conda activate agat_env

agat_sp_manage_IDs.pl -f $final_pasa_gff3 --ensembl --prefix $prefix --type_dependent --tair -o $output_annotation

agat_convert_sp_gxf2gxf.pl --gff $output_annotation --merge_loci -o $loci_merged
agat_sp_keep_longest_isoform.pl --gff $loci_merged -o $longest_isoform
```

- [x] ~~oasisia_OALV~~
- [x] ~~osedax_OFRA~~
- [x] ~~riftia_RPAC~~

STEP 7

busco_universal_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
```

```
#$ -o /data/scratch/btx654/
#$ -pe smp 4
#$ -l h_vmem=20G
#$ -l h_rt=48:0:0
#$ -j y
#$ -l highmem

species=$1
annotation_gtf="$species".AGAT.noSTOP.filt.noTE.gtf
annotation_fa="$species"_annotation.prot.fa
species_softmasked="$species"_softmasked.fa
output_busco="$species"_busco_annotation

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step6/

module load anaconda3
source activate augustus

gffread -E $annotation_gtf -g $species_softmasked -y $annotation_fa

conda deactivate
source activate busco_env
#export BUSCO_CONFIG_FILE="/data/home/btx654/.conda/envs/busco_env/busco/config/myconfig.ini"
#export AUGUSTUS_CONFIG_PATH=/data/SBCS-MartinDuranLab/02-Chema/src/Augustus/config/

```

```
busco -i $annotation_fa -m proteins -o $output_busco -c 4 -l metazoa_odb10

cd $output_busco
cd run_*
mkdir /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step7
cp full_table.tsv /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step7
cp missing_busco_list.tsv /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step7
cp short_summary.txt /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step7
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

STEP 8

gffread_universal_8.1_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 4
#$ -l h_vmem=5G
#$ -l h_rt=24:0:0
#$ -l highmem

species=$1
final_annotation="$species"_annotation_*.gff3
```

```
final_annotation_path=/data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step6/$final_annotation
species_softmasked="$species"_softmasked.fa
species_softmasked_path=/data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step6/$species_softmasked
output_mRNA="$species"_mRNA.fa
output_CDS="$species"_CDS.fa
output_proteins="$species"_proteins.fa

echo "Working on "$species

module load anaconda3
source activate augustus

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/

mkdir -p step8
cd step8

cp $species_softmasked_path ./
cp $final_annotation_path ./

gffread -w $output_mRNA -g $species_softmasked $final_annotation
gffread -x $output_CDS -g $species_softmasked $final_annotation
gffread -y $output_proteins -g $species_softmasked $final_annotation
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

obtain_trinotate_inputs_8.1_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 1
#$ -l h_vmem=20G
#$ -l h_rt=72:0:0
#$ -l highmem

species=$1
fasta_mRNA="$species"_mRNA.fa
fasta_CDS="$species"_CDS.fa
fasta_proteins="$species"_proteins.fa
gene_trans_map="$species".gene_trans_map

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/

grep ">" $fasta_mRNA | awk '{ print $2"(+)"}' > positions.txt
sed -i 's/CDS=/:/' positions.txt

awk '/^>/ {if (seqlen){print "len:"seqlen}; print ;seqlen=0;next; } { seqlen += length($0)}END{print "len:"seqlen}' $fasta_proteins | grep -v ">" > lenghts.txt

grep ">" $fasta_proteins > names.txt
sed 's/>//' names.txt > names_clean.txt

```

```
sed '/^>/s/ .*//' $fasta_mRNA > input_trinotate_mRNA.fa

cp $fasta_proteins input_trinotate_proteins.fa
INDEX=1
while read -r line
do
original_name=$(echo $line)
lenght=$(head -$INDEX lenghts.txt | tail -1)
name_clean=$(head -$INDEX names_clean.txt | tail -1)
position=$(head -$INDEX positions.txt | tail -1)

sed -i "s/$original_name/$original_name $lenght $name_clean$posi
tion/" input_trinotate_proteins.fa

INDEX=$((INDEX+1))
done < names.txt


grep ">" input_trinotate_proteins.fa > full_names.txt
sed -i 's/>//' full_names.txt

INDEX=1
while read -r line
do
full_name=$(echo $line)
name_clean=$(head -$INDEX names_clean.txt | tail -1)

echo -e $full_name'\t'$name_clean >> $gene_trans_map

INDEX=$((INDEX+1))
```

```
done < full_names.txt
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

BLASTp_universal.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 8
#$ -l h_vmem=1G
#$ -l h_rt=36:0:0

species=$1

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/trinotate_env

blastp -query input_trinotate_proteins.fa -db /data/SBCS-MartinDuranLab/03-Giacomo/db/trinotate/uniprot_sprot.pep -num_threads 8 -max_target_seqs 1 -outfmt 6 -evalue 1e-3 > blastp.outfmt6
```

BLASTx_universal.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
```

```
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 8
#$ -l h_vmem=1G
#$ -l h_rt=36:0:0

species=$1

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/trinotate_env

blastx -query input_trinotate_mRNA.fa -db /data/SBCS-MartinDuranLab/03-Giacomo/db/trinotate/uniprot_sprot.pep -num_threads 8 -max_target_seqs 1 -outfmt 6 -evalue 1e-3 > blastx.outfmt6
```

HMMER_universal.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -l highmem
#$ -pe smp 12
#$ -l h_vmem=40G
#$ -l h_rt=36:0:0
#$ -l highmem
```

```
10
11 species=$1
12
13 echo "Working on "$species
14
15 cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_D
   ec2020/step8/
16
17 module load anaconda3
18 conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda
   3/trinotate_env
19
20 hmmscan --cpu 12 --domtblout PFAM.out /data/SBCS-MartinDuranLab/
   03-Giacomo/db/trinotate/Pfam-A.hmm input_trinotate_proteins.fa >
   pfam.log
```

- i got “Segmentation fault” for oasisia and osedax so I will send this job with more ram (40G) for them

---

This section is intended to fix the “Segmentation Fault” error of the previous script. Basically we split the input_trinotate_proteins.fa in 100 parts in order to not overload HMMER

obtain_rename_chunks_universal_v1.sh

```
1 #!/bin/bash
2 #$ -cwd
3 #$ -pe smp 1
4 #$ -l h_vmem=1G
5 #$ -j y
6 #$ -l h_rt=01:00:00
7 species=$1
8 trinotate_folder=input_trinotate_proteins_chunks
9 trinotate_chunk=input_trinotate_proteins.fa_chunk_
```

```
cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/

mkdir $trinotate_folder
module load exonerate/2.4.0
fastasplit -f input_trinotate_proteins.fa -o ./input_trinotate_proteins_chunks/ -c 100
cd $trinotate_folder

readlink -f "$trinotate_chunk"* > list
for i in $(seq 1 100);
do
  original_file=$(head -"$i" list | tail -1)
  renamed_file="$trinotate_chunk""$i"
  mv -- "$original_file" "$renamed_file"
done
```

Run HMMER on the single chunks:
HMMER_universal_chunks.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 12
#$ -l h_vmem=40G
#$ -l h_rt=36:0:0
#$ -t 1-100
#$ -l highmem

species=$1
```

```
target_chunk=input_trinotate_proteins.fa_chunk_"${SGE_TASK_ID}"
PFAM_out=PFAM_"${SGE_TASK_ID}".out
pfam_log=pfam_"${SGE_TASK_ID}".log

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/trinotate_env

hmmscan --cpu 12 --domtblout $PFAM_out /data/SBCS-MartinDuranLab/03-Giacomo/db/trinotate/Pfam-A.hmm $target_chunk > $pfam_log
```

Some jobs will crash because of the same error so we can divide in 100 subchunks the chunks that failed the previous jobs. With these steps we will remove those problematic subchunks from the analysis. Every subchunk removed in this way will be just the 0.01% of the total so it won't impact too much the consistency of our analyses

obtain_rename_chunks_universal_fix_v1.sh

```
#!/bin/bash
#$ -cwd
#$ -pe smp 1
#$ -l h_vmem=1G
#$ -j y
#$ -l h_rt=01:00:00

species=$1
chunk=$2
```

```
trinotate_folder_chunk=input_trinotate_proteins_chunk_"$chunk"
chunk_fix=input_trinotate_proteins.fa_chunk_"$chunk"
trinotate_chunk=input_trinotate_proteins.fa_chunk_

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks/

mkdir $trinotate_folder_chunk
module load exonerate/2.4.0
fastasplit -f $chunk_fix -o ./$trinotate_folder_chunk/ -c 100
cd $trinotate_folder_chunk

readlink -f "$trinotate_chunk"* > list
for i in $(seq 1 100);
do
  original_file=$(head -"$i" list | tail -1)
  renamed_file="$trinotate_chunk""$i"
  mv -- "$original_file" "$renamed_file"
done
```

Run HMMER on the subchunks
HMMER_universal_chunks_fix_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 12
#$ -l h_vmem=1G
#$ -l h_rt=36:0:0
#$ -t 1-100
#$ -l highmem
```

```

species=$1
chunk=$2
trinotate_folder_chunk=input_trinotate_proteins_chunk_"$chunk"
target_chunk=input_trinotate_proteins.fa_chunk_"${SGE_TASK_ID}"
PFAM_out=PFAM_"${SGE_TASK_ID}".out
pfam_log=pfam_"${SGE_TASK_ID}".log

echo "Working on "$species" on chunk: "$chunk

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks/$trinotate_folder_chunk

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/trinotate_env

hmmscan --cpu 12 --domtblout $PFAM_out /data/SBCS-MartinDuranLab/03-Giacomo/db/trinotate/Pfam-A.hmm $target_chunk > $pfam_log
```

to be removed:

- oasisia 38 37 rm PFAM_37.out
- oasisia 46 25 rm PFAM_25.out
- oasisia 75 32 rm PFAM_32.out
- osedax 75 81 rm PFAM_81.out
- osedax 52 70 rm PFAM_70.out

Now we need to put together the PFAM.out file from the subchunks to basically rebuild the PFAM.out file of the chunk that failed before
merge_chunks_universal_fix_v1.sh

```
#!/bin/bash
#$ -cwd
#$ -pe smp 1
```

```
#$ -l h_vmem=1G
#$ -j y
#$ -l h_rt=01:00:00

species=$1
chunk=$2
trinotate_folder_chunk=input_trinotate_proteins_chunk_"$chunk"
PFAM_chunk_out=PFAM_"$chunk".out

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks/$trinotate_folder_chunk

for i in $(seq 1 100);
do
  PFAM_out=PFAM_"$i".out
  PFAM_out_ok=PFAM_"$i"_ok.out
  head -n -10 $PFAM_out | tail -n +4 > $PFAM_out_ok
done
cat PFAM_*_ok.out > ../$PFAM_chunk_out
```

And finally we need to put together the PFAM.out file of all the chunks to reconstruct the PFAM.out file that we would have obtained if we didn't encounter the Segmentation fault error

merge_chunks_universal_v1.sh

```
#!/bin/bash
#$ -cwd
#$ -pe smp 1
#$ -l h_vmem=1G
#$ -j y
#$ -l h_rt=01:00:00
```

```

species=$1

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks

for i in $(seq 1 100);
do
  PFAM_out=PFAM_"$i".out
  PFAM_out_ok=PFAM_"$i"_ok.out
  head -n -10 $PFAM_out | tail -n +4 > $PFAM_out_ok
done
cat PFAM_*_ok.out > ../PFAM.out
```

End of fix

Signalp_universal.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 1
#$ -l h_vmem=8G
#$ -l h_rt=36:0:0

species=$1

echo "Working on "$species

```

```
cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/

module load perl

/data/SBCS-MartinDuranLab/03-Giacomo/src/signalp-4.1/signalp -f short -n signalp.out input_trinotate_proteins.fa
```

wrapper_trinotate_8.2_v1.sh

```
#!/bin/bash
#$ -wd /data/home/btx654/scripts/annotation/New_annotation_Dec2020/step8/
#$ -j y
#$ -pe smp 1
#$ -l h_vmem=1G
#$ -l h_rt=1:0:0

species=$1

echo "Working on "$species

qsub BLASTp_universal.sh $species
qsub BLASTx_universal.sh $species
qsub HMMER_universal.sh $species
qsub Signalp_universal.sh $species
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

trinotate_SQLite_database_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
```

```
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 1
#$ -l h_vmem=8G
#$ -l h_rt=2:0:0

species=$1
gene_trans_map="$species".gene_trans_map
sqlite_db="$species".sqlite

echo "Working on "$species

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/trinotate_env

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/

Build_Trinotate_Boilerplate_SQLite_db.pl $species

Trinotate $sqlite_db init --gene_trans_map $gene_trans_map --transcript_fasta input_trinotate_mRNA.fa --transdecoder_pep input_trinotate_proteins.fa
#LOAD annotations, below are examples from Trinotate manual, change accordingly to your species ouput from blast+:
Trinotate $sqlite_db LOAD_swissprot_blastp blastp.outfmt6
Trinotate $sqlite_db LOAD_swissprot_blastx blastx.outfmt6
Trinotate $sqlite_db LOAD_signalp signalp.out
Trinotate $sqlite_db LOAD_pfam PFAM.out

```

```
Trinotate $sqlite_db report --incl_pep --incl_trans > annotation_report.xls
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

STEP 9

pantherScore_universal_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 30
#$ -l h_vmem=5G
#$ -l h_rt=240:00:0
#$ -l highmem

species=$1
gffread_proteins="$species"_proteins.fa #generated by gffread_universal_8.1_v1.sh
gffread_proteins_path=/data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/$gffread_proteins
panther_output="$species"_Panther

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/

```

```
mkdir -p step9
cd step9

cp $gffread_proteins_path ./

module load perl
module load hmmer/

export PERL5LIB=/data/SBCS-MartinDuranLab/03-Giacomo/src/hmmscoring/lib/

perl /data/SBCS-MartinDuranLab/03-Giacomo/src/hmmscoring/pantherScore2.2.pl -l /data/SBCS-MartinDuranLab/03-Giacomo/src/hmmscoring/PANTHER15.0/ -D B -n -o $panther_output -i $gffread_proteins -c 30 -V -s

mkdir /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step9
cp "$species"_Panthe* /data/SBCS-MartinDuranLab/03-Giacomo/data/$species/annotation/New_annotation_Dec2020/step9
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

combining Panther and Trinotate:

```
cd /data/scratch/btx654/btx604-scratch/riftia/annotation_step9
cp /data/SBCS-MartinDuranLab/03-Giacomo/data/riftia/annotation/step8/riftia_annotation_report.xls ./
sort oasisia_Panther > panther_sorted
cut -f 1 panther_sorted > IDs_panther
cut -f 2 annotation_report.xls | tail -n +2 > IDs_all
fgrep -v -f IDs_panther IDs_all > IDs_absentPanther ### There are oasisia:8632 osedax:4052 riftia:8182 genes without Panther ann
```

```
otation
awk '{print $0"\t""NO PTHR""\t""NO HIT"}' IDs_absentPanther > PANTHER_nohits
cat panther_sorted PANTHER_nohits | sort -k 1,1 > Panther_sorted_allgenes
# now we need to remove duplicated lines from panther all genes
awk '!a[$1]++' Panther_sorted_allgenes > Panther_sorted_allgenes_noduplicates
## use vim to add a header in Owenia_Panther_sorted_allgenes so that it matches Trinotate file
# #gene_id        transcript_id        sprot_Top_BLASTX_hit        RNAMMER        prot_id        prot_coords        sprot_Top_BLASTP_hit        Pfam        SignalP        TmHMM        eggnog        Kegg        gene_ontology_BLASTX        gene_ontology_BLASTP        gene_ontology_Pfam        transcript        peptide
paste annotation_report.xls Panther_sorted_allgenes_noduplicates > oasisia_annotation_Dec2020_TrinoPanther.xls
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

KofamKOALA

```
module load anaconda3
conda create --prefix /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/KofamKOALA_env
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/KofamKOALA_env
conda install -c bioconda kofamscan


cd /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/KofamKOALA_env
wget -r ftp://ftp.genome.jp/pub/db/kofam/profiles.tar.gz
cd ftp.genome.jp/pub/db/kofam
tar -zxvf profiles.tar.gz
```

```
wget ftp://ftp.genome.jp/pub/db/kofam/ko_list.gz
gzip -d ko_list.gz
```

KofamKoala_universal_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 12
#$ -l h_vmem=40G
#$ -l h_rt=72:0:0
#$ -l highmem

species=$1
gffread_proteins="$species"_proteins.fa #generated by gffread_universal_8.1_v1 .sh
gffread_proteins_path=/data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/$gffread_proteins
output_kofam="$species"_kofam_result.txt

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step9
mkdir -p kofamKoala
cd kofamKoala

cp $gffread_proteins_path ./
awk '/^>/ {printf("\n%s\n",$0);next; } { printf("%s",$0);}  END {printf("\n");}' < $gffread_proteins | tail -n +2 > single_fasta_proteins.fa
```

```
23
24 module load anaconda3
25 conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda
   3/KofamKOALA_env
26
27 exec_annotation \
28  --profile=/data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/Ko
   famKOALA_env/ftp.genome.jp/pub/db/kofam/profiles/ \
29  --ko-list=/data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/Ko
   famKOALA_env/ftp.genome.jp/pub/db/kofam/ko_list \
30  --cpu=12 \
31  --format=mapper \
32  --report-unannotated \
33  -o $output_kofam \
34  single_fasta_proteins.fa
```

- [x] ~~oasisia~~
- [x] ~~osedax~~
- [x] ~~riftia~~

I think I have to divide in chunks
obtain_rename_chunks_universal_v1.sh

```
1  #!/bin/bash
2  #$ -wd /data/scratch/btx654/
3  #$ -o /data/scratch/btx654/
4  #$ -pe smp 1
5  #$ -l h_vmem=1G
6  #$ -j y
7  #$ -l h_rt=01:00:00
8  species=$1
9  gffread_folder=proteins_chunks
10 gffread_chunk="$species"_proteins.fa_chunk_
11
```

```
cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step9/kofamKoala

mkdir $gffread_folder
module load exonerate/2.4.0
fastasplit -f single_fasta_proteins.fa -o ./$gffread_folder/ -c 100
cd $gffread_folder

readlink -f single_fasta_proteins.fa_* > list
for i in $(seq 1 100);
do
  original_file=$(head -"$i" list | tail -1)
  renamed_file="$gffread_chunk""$i"
  mv -- "$original_file" "$renamed_file"
done
```

Run kofamkoala on the single chunks:
KofamKoala_universal_chunks.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
#$ -j y
#$ -pe smp 6
#$ -l h_vmem=40G
#$ -l h_rt=36:0:0
#$ -t 1-100
#$ -l highmem

species=$1
target_chunk="$species"_proteins.fa_chunk_"${SGE_TASK_ID}"
```

```
output_kofam="$species"_kofam_result_chunk_"${SGE_TASK_ID}".txt

echo "Working on "$species

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step9/kofamKoala/proteins_chunks/

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/KofamKOALA_env

exec_annotation \
 --profile=/data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/KofamKOALA_env/ftp.genome.jp/pub/db/kofam/profiles/ \
 --ko-list=/data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/KofamKOALA_env/ftp.genome.jp/pub/db/kofam/ko_list \
 --cpu=6 \
 --format=mapper \
 --report-unannotated \
 -o $output_kofam \
 $target_chunk
```

Some jobs will crash because of the same error so we can divide in 100 subchunks the chunks that failed the previous jobs. With these steps we will remove those problematic subchunks from the analysis. Every subchunk removed in this way will be just the 0.01% of the total so it won't impact too much the consistency of our analyses

obtain_rename_chunks_universal_fix_v1.sh

```
#!/bin/bash
#$ -cwd
#$ -pe smp 1
```

```
#$ -l h_vmem=1G
#$ -j y
#$ -l h_rt=01:00:00

species=$1
chunk=$2
trinotate_folder_chunk=input_trinotate_proteins_chunk_"$chunk"
chunk_fix=input_trinotate_proteins.fa_chunk_"$chunk"
trinotate_chunk=input_trinotate_proteins.fa_chunk_

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks/

mkdir $trinotate_folder_chunk
module load exonerate/2.4.0
fastasplit -f $chunk_fix -o ./$trinotate_folder_chunk/ -c 100
cd $trinotate_folder_chunk

readlink -f "$trinotate_chunk"* > list
for i in $(seq 1 100);
do
  original_file=$(head -"$i" list | tail -1)
  renamed_file="$trinotate_chunk""$i"
  mv -- "$original_file" "$renamed_file"
done
```

Run HMMER on the subchunks
HMMER_universal_chunks_fix_v1.sh

```
#!/bin/bash
#$ -wd /data/scratch/btx654/
#$ -o /data/scratch/btx654/
```

```
#$ -j y
#$ -pe smp 12
#$ -l h_vmem=1G
#$ -l h_rt=36:0:0
#$ -t 1-100
#$ -l highmem

species=$1
chunk=$2
trinotate_folder_chunk=input_trinotate_proteins_chunk_"$chunk"
target_chunk=input_trinotate_proteins.fa_chunk_"${SGE_TASK_ID}"
PFAM_out=PFAM_"${SGE_TASK_ID}".out
pfam_log=pfam_"${SGE_TASK_ID}".log

echo "Working on "$species" on chunk: "$chunk

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks/$trinotate_folder_chunk

module load anaconda3
conda activate /data/SBCS-MartinDuranLab/03-Giacomo/src/anaconda3/trinotate_env

hmmscan --cpu 12 --domtblout $PFAM_out /data/SBCS-MartinDuranLab/03-Giacomo/db/trinotate/Pfam-A.hmm $target_chunk > $pfam_log
```

to be removed:

- oasisia 38 37 rm PFAM_37.out
- oasisia 46 25 rm PFAM_25.out
- oasisia 75 32 rm PFAM_32.out
- osedax 75 81 rm PFAM_81.out
- osedax 52 70 rm PFAM_70.out

Now we need to put together the PFAM.out file from the subchunks to basically rebuild the PFAM.out file of the chunk that failed before merge_chunks_universal_fix_v1.sh

```
#!/bin/bash
#$ -cwd
#$ -pe smp 1
#$ -l h_vmem=1G
#$ -j y
#$ -l h_rt=01:00:00

species=$1
chunk=$2
trinotate_folder_chunk=input_trinotate_proteins_chunk_"$chunk"
PFAM_chunk_out=PFAM_"$chunk".out

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks/$trinotate_folder_chunk

for i in $(seq 1 100);
do
  PFAM_out=PFAM_"$i".out
  PFAM_out_ok=PFAM_"$i"_ok.out
  head -n -10 $PFAM_out | tail -n +4 > $PFAM_out_ok
done
cat PFAM_*_ok.out > ../$PFAM_chunk_out
```

And finally we need to put together the PFAM.out file of all the chunks to reconstruct the PFAM.out file that we would have obtained if we didn't encounter the Segmentation fault error

merge_chunks_universal_v1.sh

```
#!/bin/bash
#$ -cwd
#$ -pe smp 1
#$ -l h_vmem=1G
#$ -j y
#$ -l h_rt=01:00:00

species=$1

cd /data/scratch/btx654/btx604-scratch/$species/New_annotation_Dec2020/step8/input_trinotate_proteins_chunks

for i in $(seq 1 100);
do
  PFAM_out=PFAM_"$i".out
  PFAM_out_ok=PFAM_"$i"_ok.out
  head -n -10 $PFAM_out | tail -n +4 > $PFAM_out_ok
done
cat PFAM_*_ok.out > ../PFAM.out
```

End of fix